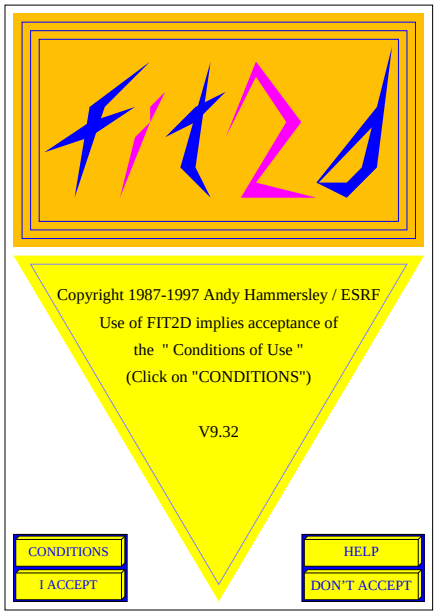
FIT2D: An Introduction and Overview

A P Hammersley

ESRF, BP 220, 38043 Grenoble, France

February 5, 1997



Abstract

FIT2D is both a general purpose and specialist 1 and 2 dimensional data analysis program, used on most of the European Synchrotron Research Facility beam-lines and by many other crystallography groups throughout the world. **FIT2D** is used for both interactive and "batch" data processing, and is used by many different user communities for different purposes.

Calibration and correction of detector distortions is one of the important uses of **FIT2D**. Difficult data analysis problems may be tackled using fitting of user specified models. To enable model fitting to be performed on a wide variety of input data, many other more basic data analysis operations are also available. A wide variety of performant graphical display methods are available.

The program is available free to academic users and is available by Internet for a wide range of computer systems. This paper provides an overview to **FIT2D** and is intended to serve as an "Introductory Guide", to complement the full "Reference Manual".

Contents

1 Introduction **4**

2 The Start-Up of FIT2D **5**

3 General Principles And Common Features **6**

3.1 The "MEMORY" 6

3.2 "Active Data Region" (*ADR*) 6

3.3 The Graphics Window 7

3.4 Information, Warnings, and User Input Areas 7

3.5 Parameter Value Input 8

3.5.1 Integer, Real Value, and Character String Input 8

3.5.2 Boolean Value Input 9

3.5.3 Graphical Menu Choices 10

3.5.4 Graphical Coordinate Input 11

3.6 Graphical forms 12

3.7 File Selection and input 14

3.8 False Colour Image Display and Intensity Scaling 17

3.9 Graphical Display Possibilities 19

3.10 Display Style Options 21

4 FIT2D for General Image Processing Operations **22**

5 FIT2D for 1-D "Peak" Fitting **22**

6 FIT2D for On-line Display **23**

7 FIT2D for Powder Diffraction with Area Detectors **24**

8 The "Keyboard" Interface **26**

8.1 The `CALIBRATION` Sub-Menu 26

8.2 The `FIT` Sub-Menu 27

9 Command line Options **27**

9.1 Program Dimensions 27
9.2 Variance Arrays 28
9.3 "Landscape" Output 28
9.4 "Keyboard" Interface 28
9.5 No Graphics 29
9.6 Other Options 29

10 Trouble-Shooting **29**

10.1 Command not found 29
10.2 Failure in **FIT2D** Start-up 29
10.3 Dynamic Memory Allocation Failure 31
10.4 No Image Output 31

11 Availability **31**

12 Future Developments **32**

13 Acknowledgements **32**

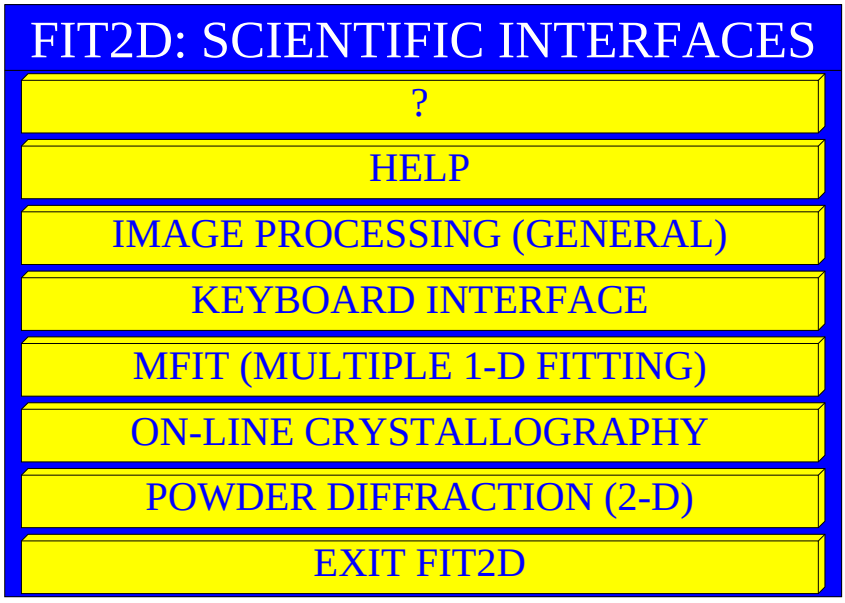
14 Appendix A: Colour Usage **32**

1 Introduction

This paper is an introduction to the "Graphical User Interface" (GUI) of **FIT2D**, and an overview of the main functionality. More detailed descriptions of particular commands are available in the **FIT2D** "Reference Manual" (Hammersley, 1995). Basic knowledge of modern computer systems is assumed, but no detailed knowledge of a particular operating system is needed.

The manner in which the user works with the GUI is described. The GUI is designed to make easily accessible the most used functionality which is available within **FIT2D**. To achieve this, functionality has been arranged into different "interfaces" for different scientific areas. Figure 1 shows the menu for the present choice of "scientific interfaces". (Further interfaces are likely to be added in future versions.) It should be emphasized that data can be moved from one interface to another without restriction e.g. 2-D Powder diffraction data could be input in the **IMAGE PROCESSING (GENERAL)** menu, and manipulated, then Debye-Scherrer rings could be integrated within the **POWDER DIFFRACTION (2-D)** menu.

Figure 1: The **FIT2D** Scientific Interfaces Main Menu



By clicking on the different "buttons" the user enters the different "interfaces". The buttons are:

? The question mark in this menu, and other menus gives a brief description of each of the available choices.

HELP The "HELP" button enters the "context" related help associated with the menu. In this case there is information describing the different available interfaces. Clicking on "HELP" enters a "pager" which allows the user to scroll down and up the help information.

IMAGE PROCESSING (GENERAL) Is an interface for general purpose image processing and display operations on data input from a file.

KEYBOARD INTERFACE This enters the "keyboard" command-line interface to **FIT2D**. Certain functionality is presently only available from this interface.

MFIT (MULTIPLE 1-D FITTING) This interface allows 1-D "peak" functions such as Gaussian, Lorentzian, and Voigt functions to be fitted to data along with other functions such as a polynomial "background" function, and exponential decay or trigonometric functions. The fit model may be fitted repeatedly to a whole series of 1-D datasets. (This provides the same functionality as was previously available through the program **MFIT** (Hammersley, 1989).)

ON-LINE CRYSTALLOGRAPHY This interface allows on-line display and examination of crystallographic data, with the option of automatic updating of new image data as they are obtained. Simple statistics to evaluate data quality are produced.

POWDER DIFFRACTION (2-D) This interface specialises in the integration of Debye-Scherrer rings from 2-D detectors, to 1-D "2-theta" scans, and to other scans. This is useful for standard powder refinement, but is also useful for texture analysis and other types of integrations and "scans", and is used for small angle scattering data.

EXIT FIT2D Allows **FIT2D** to be exited, after confirmation.

2 The Start-Up of FIT2D

Computer systems should be set-up so that a user can simply type `fit2d` (in lower case) on the command-line to start it working: e.g.

```
> fit2d
```

Additionally the commands `fit2d_new`, and `fit2d_old` may be available to start-up a new "development" version, or a previous version respectively. (The presence of these versions will depend on the particular site, but sites are recommended to keep the previous version when updating **FIT2D** in-case the new version leads to problems.) A new window should appear for the **FIT2D** graphics, and the **FIT2D** "banner page" should appear within the window. (The start-up may fail for a number of different reasons. These are explained in Section 10 together with advice on ways to circumvent problems.)

The "banner page" and the terminal window contain some general information, including the version number of **FIT2D** which is being used. The "banner page" also contains four buttons:

HELP Help text on the general use of **FIT2D** and in particular on the banner page.

CONDITIONS Text explaining the "conditions of use" for **FIT2D**.

I ACCEPT Accept the "conditions of use", and entering the main part of the program.

DON'T ACCEPT Refuse the "conditions of use" and exit **FIT2D**.

Having accepted the "conditions of use", a "graphical form" appears. This form, allows the size of internal program arrays to be set. **FIT2D** uses "dynamic allocation" of program arrays to allow enormous flexibility in the size of data-sets which may be treated. Normally only the size of the internal arrays in the "X-direction" (horizontal) and "Y-direction" (vertical) need to be set. These dimensions should normally be set as large, or larger than the size of the data to be input, but smaller arrays may be used, with the consequence that not all the data may be input at full resolution.

A "Memory" array, to allow a second image to be stored within **FIT2D**, is created by default, but doubles the computer memory demands for the internal program arrays so may be disabled. By default "memory" arrays are created since most commands depend on the existence of the "memory".

It is also possible to create "variance" arrays which allow error propagation to be carried out, and weighted fitting to be performed. However, this doubles again the requirements for computer memory, and operations will take twice as long or longer. By default variance arrays are not created.

(It is possible to avoid this form completely, by specifying this information using "command-line options" (see Section 9).)

3 General Principles And Common Features

A number of general principles persist throughout **FIT2D**. These should be largely intuitive and obvious, but some points may be missed by a new user, so are explained here.

3.1 The "MEMORY"

FIT2D uses two main internal arrays to store image data and to perform almost all operations. The default array will be referred to as the "current data array"; this is where data is stored when input, and where data to be displayed graphically must be situated. The second array will be referred to as the "memory". The "memory" is like the memory facility on many calculators except that the single memory may take up 40 Mbytes or more, and may store a full image.

To move data from the current data array to the memory and vice versa, most menus contain the **EXCHANGE** button. The output of all operations within the GUI is in the current data array. In many cases the original data is transferred to the memory, and thus may be recovered using the **EXCHANGE** button. In other cases the original data may be recovered using the inverse operation e.g. **SUBTRACT** after **ADD**.

3.2 "Active Data Region" (*ADR*)

All commands which operate on data elements may be limited as to the range of data elements upon which they act. The region or "window" over which operations take place

is referred to as the "Active Data Region" (*ADR*). In other programs and contexts this concept is sometimes referred to as a "Data Window" or as a "Region Of Interest" (ROI).

When data is input the *ADR* is initially set to include all the data. The menu commands **ZOOM IN**, **UN-ZOOM**, and **FULL** buttons allow the *ADR* to be changed (see Section 3.5.3). The **MOVEMENT** graphics sub-menu also contains many commands which change the *ADR* e.g. all the commands which move around the image; **UP, DOWN, LEFT,** etc.

It should be noted that the *ADR* concept does not just affect graphical display, but also affects arithmetic and other operations, and the output of data to file (**OUTPUT**). **FIT2D** follows the WYSIWYG ("what you see is what you get") principle in data operations, so only the area of the data which is currently displayed will be operated upon. (The exception is the **EXCHANGE** command which swaps the entire data-sets including the definitions of the *ADR*.)

3.3 The Graphics Window

The graphics window may be resized, to suit user needs, by dragging on one of the corners of the window border. When the graphics window is resized, it will change size, but will keep the same "aspect ratio" i.e. the relative lengths will be kept the same. To do this the correctly proportioned window is fitted within the region defined by the dragged cursor. Thus, to define a larger window, it is important to drag the cursor in both directions. The window can also be made smaller, but there is a minimum size.

By default the graphics window in created in the "portrait" format, in the same ratio as A4 paper. However other formats may be used and in particular "landscape" A4 format is available (see Section 9).

3.4 Information, Warnings, and User Input Areas

Different types of prompt text, information text, and warning and error messages will appear in the graphics window and also extra information text will appear in the terminal window. The main "dialogue" is in always in the graphics window, but sometimes more detailed explanation is given in the terminal window.

Throughout the GUI a simple colour coding is used to indicate: user input, information, or warning messages. "Active" regions such as buttons and region for inputting text have a yellow background with blue writing. (Note: the image of the data and the whole graphics window may be "active" for certain operations, but these are left in their normal colours.) General information text has a white background and black text, and warning or error messages have a red border and within black writing on a white background.

Information and warning text in the terminal window is identified by one of four initial "keywords":

`INFO`: The text is of a purely informative nature e.g. values calculated as the result of some operation, or information on an input data-set.

`NOTE`: The text is also informative, but highlights a potential for misunderstanding by an inexperienced user. Thus, such messages should be carefully noted.

`WARNING`: Something has not worked as it should. This may be of greater or lesser importance depending on the circumstances. e.g. An input file was not found, or was of the wrong format.

`ERROR`: Something "serious" has gone wrong. This may require exiting **FIT2D** or may be "recoverable" depending on the circumstances. e.g. The system has failed to allocate dynamic memory as requested.

It should be noted that these categories are not strict i.e. in some circumstances a `WARNING` message may be more serious than an `ERROR` message, nevertheless, this show a generally increasing importance and need for user comprehension and maybe action.

3.5 Parameter Value Input

FIT2D has been developed in a very modular fashion and the user interface exhibits this modularity. This means efficient development and easy use. The same "components" are re-used many times and the user will soon become familiar with the input style.

At the most basic level of user input is parameter input. This may be a number e.g. a value for the number of pixels for the internal arrays in one direction, a character string e.g. a title for a data-set, or a choice between a limited number of values.

The following different types of input exist:

Integer Value

Real Value

Boolean Value

Character String

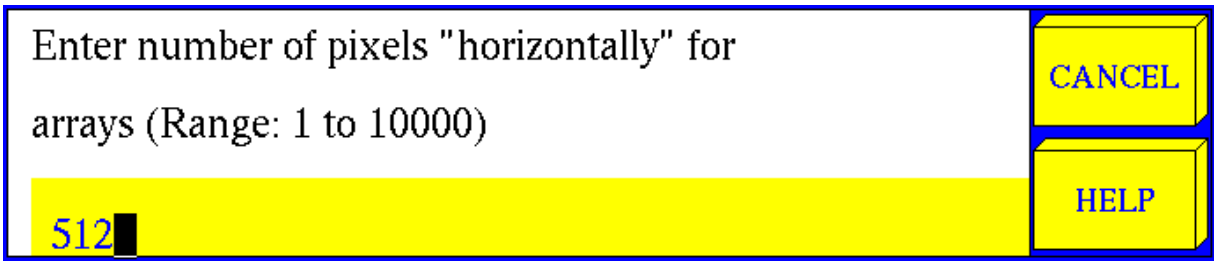
Graphical Menu Choice

Graphical Coordinate

3.5.1 Integer, Real Value, and Character String Input

Figure 2 shows the typical display box for the input of an integer value. A very similar style is used for the input of real and character string values.

Figure 2: Example of the User Input of an Integer Value



The following "components" of the dialogue box are apparent:

Descriptive Text: At the top (with the white background) is a short text describing the required input. This includes the text `(Range: 1 to 10000)` which tells that the value must by between 1 and 10000. For parameters where any value may be input this text is not present.

CANCEL The cancel button cancels the parameter input without changing its value. This may also cancel the operation for which the parameter value was being requested.

HELP The help button may be clicked to obtain more detailed information concerning the parameter value being input and its context in relation to the operation being undertaken.

Text Input Box: The yellow lower rectangle is the area for the display of input text. The box already contains the text `512`, which is the default value for the parameter, and the cursor to the right of the number. Text entered from the keyboard will appear in this box at the cursor position.

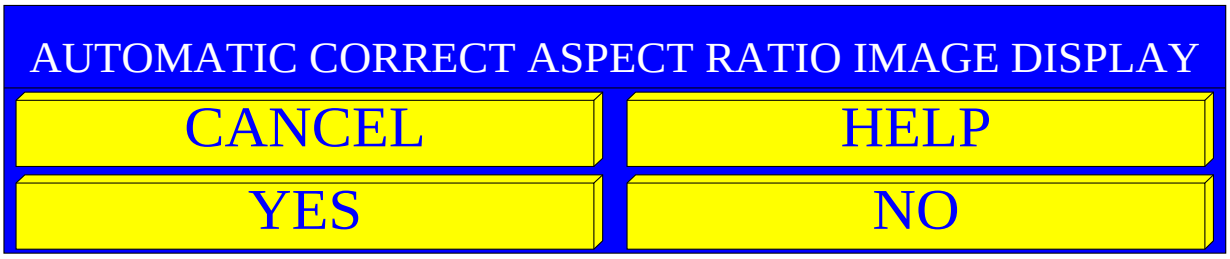
The text may be "edited" using the left and right arrow keys to move the cursor, the "backspace" or "delete" key to remove text and other numerical or alphabetic keys to enter characters. Additionally the up arrow and down arrow keys allow previous inputs to be scrolled. The up-arrow key allows previous input to be "re-called". Each time the up-arrow key is pressed the previous input replaces any entered text. The down-arrow key can be used after the up-arrow key to scan through the previous inputs. If used by itself, or when the current input is being displayed, it removes all characters. This can be useful for quickly removing all the characters of a presented default response.

When the text is as required, the <ENTER> (or <RETURN>) key is used to enter the value. If the value is correct the parameter input is completed, otherwise an error message will presented, followed by the full information "help" message, and the user will be required to re-enter the parameter value.

3.5.2 Boolean Value Input

Many times a simple "yes/no" choice is offered to the user. This sort of "boolean" parameter value input is controlled with a simple menu of four buttons. An example is shown in Figure 3.

Figure 3: Example of the User Input of a Boolean Value



The buttons are:

? Help information on the parameter value being requested

CANCEL Cancel input without changing the value of the parameter. This will also in many cases cancel the operation for which the parameter was requested.

YES Set the value to "Yes".

NO Set the value to "No".

The "CANCEL" button will cancel the operation and the parameter will keep its previous value.

The "?" button will display help text describing the purpose of the parameter and the type of value required.

3.5.3 Graphical Menu Choices

Graphical Menus are the normal manner in which data analysis operations may be chosen by the user. The choice of "scientific interfaces" is a graphical menu, and each of the interfaces is a menu. Various menu commands may themselves produce sub-menus e.g. the **Z-SCALING** command produces a sub-menu.

When a button is pressed, it will be re-drawn pressed-in, to show that it has indeed been pressed. (Note: afterwards it may be re-drawn normally, and if this happens quickly enough the pressed-in button may not be noticed.)

Different interfaces have different commands, but the following commands are common to several or all menus:

EXIT Every menu has an exit button which is used to exit the menu and return to the previous menu, or in the case of the "scientific interfaces" menu to exit **FIT2D**, provided confirmation is given.

? The question mark button provides a message with the available button commands and a short explanation of the commands.

HELP The help button provides a "context" related help message on the menu and the overall functionality which it provides.

INPUT Input data from a file (see Section 3.7 for more details).

OUTPUT The output button is used to save the current *ADR* of the data to an output file.

EXCHANGE Exchange the current data with the memory data. If the memory was previously undefined, then the current data array becomes un-defined (see Section 3.1).

ZOOM IN Define a smaller *ADR* using graphical cursor input (see Section 3.2).

FULL Set the *ADR* to be the whole of the currently defined data (see Section 3.2).

UN-ZOOM Increase the size of the *ADR*. This is done symmetrically if possible (see Section 3.2).

PRINT Save the currently displayed graphical image to a PostScript file. The name for the PostScript file is requested.

DISPLAY Sub-menu with variety of alternative data display possibilities to the false colour image e.g. contour or 3-D surface views (see Section 3.9).

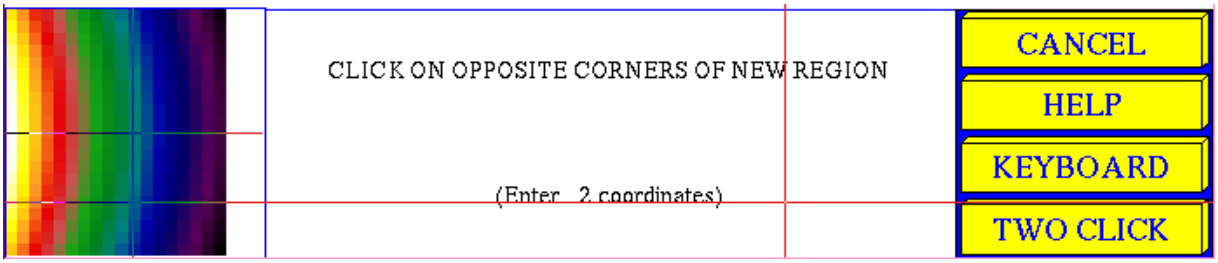
OPTIONS Sub-menu offering control of graphical display style options (see Section 3.10).

3.5.4 Graphical Coordinate Input

FIT2D has a true **graphics user interface** in that the user interface includes the graphics display, as opposed to only buttons and dialogue boxes, of most programs with "GUI's". For many operations the most convenient form of user interaction is for the user to indicate some feature in the data. This is performed by clicking on a false colour image of the data (or an X/Y graph for 1-D data).

An example of the dialogue box associated with graphical coordinate input is shown in Figure 4.

Figure 4: Example of Graphical Coordinate Input



The various components of this dialogue region are:

"Spy-Glass": On the left is the "spy-glass" or zoom window which shows the region around the cursor at full pixel resolution. As the cursor is moved within the data image, so the image within the spy-glass changes. The cross-hairs in the centre of the spy-glass show the position of the cursor. Unlike the main image display, the spy-glass is always full range automatic intensity scaling.

User Prompt: The white background rectangle contains "prompt" information describing the purpose of the required coordinate input. Note: For some coordinate input, this region can be replaced by a yellow button, to be pressed to end input e.g. when the user can enter a variable number of coordinates.

Cross-Hair: Drawn on top of all the graphics is the cursor cross-hair. These vertical and horizontal lines follow the cursor, and may be useful when the input coordinate needs to be aligned with features in the data.

HELP The help button will provide a message describing in greater detail than the "prompt" the required input and its purpose.

CANCEL The cancel button allows the input to be cancelled. This will usually also cancel the operation for which the coordinate was required.

KEYBOARD The keyboard button may be used to enter the X and Y coordinate position, by entering two numbers from the keyboard instead of the graphical click. This can be useful when some precise value is required.

TWO CLICK Enter "two click mode". In this mode for graphical coordinate input the first click within the image stops the spy-glass, and then a second within the spy-glass defines the input coordinate. This allows more accurate input, particularly for very large images which will be displayed at much lower than single pixel resolution. When in "two click mode" this button will change to **ONE CLICK**, and may be used to return to normal "one click mode".

When a coordinate is selected, a double cross is drawn on the image at the appropriate position. The double cross is made from a black horizontal/vertical cross, and a white diagonal cross. Thus, regardless of the colour of the data the cross will be visible.

Often when two or more coordinates are input, the later coordinates are related in some way to the first, or previous coordinate. In such a case, a "rubber-band" line or rectangle is drawn and re-drawn as the cursor moves. e.g. The **ZOOM IN** command requires two coordinates to define the new *ADR*; and a rectangular box will be drawn defined by the first coordinate and the cursor position, whilst the second coordinate is being input.

3.6 Graphical forms

Often a number of control parameters for a data analysis operation may be varied by the user. Since for many of the parameters the default values will be suitable "graphical forms" are presented. The same basic "form" is used for many different purposes. The form for defining the size of program arrays when **FIT2D** is first started-up is an example. All forms have the same basic components and functionality, although the number of user variable parameters may change. The form presented to the user for modification when the **INTEGRATE** command of the **POWDER DIFFRACTION (2-D)** interface is shown as an example in Figure 5.

Figure 5: Example of a Graphical Form

Each user controllable parameter has a button which may be clicked to change the value of the parameter. This button will contain some short text to describe the parameter. To the far left will be a short phrase giving more detail as to the purpose of the parameter, and immediately to the left is the current value.

Clicking on the buttons of parameters which require a "YES/NO" value will simply toggle the value, whilst clicking on other types of parameters will produce a "dialogue" region at the bottom of the graphics window. Here the parameter value can be entered from the keyboard (see Section 3.5.1), or a choice or inputs will be displayed as a graphical menu.

In addition to the parameter buttons are the following general buttons:

O.K. Clicking on the O.K. button will exit the form, saving any changes to parameter values which may have been made.

CANCEL Clicking on the cancel button will exit the form without making any changes. This may also cancel the operation for which the parameters were being input and return to a menu.

INFO This gives general information explaining the use of the graphical form.

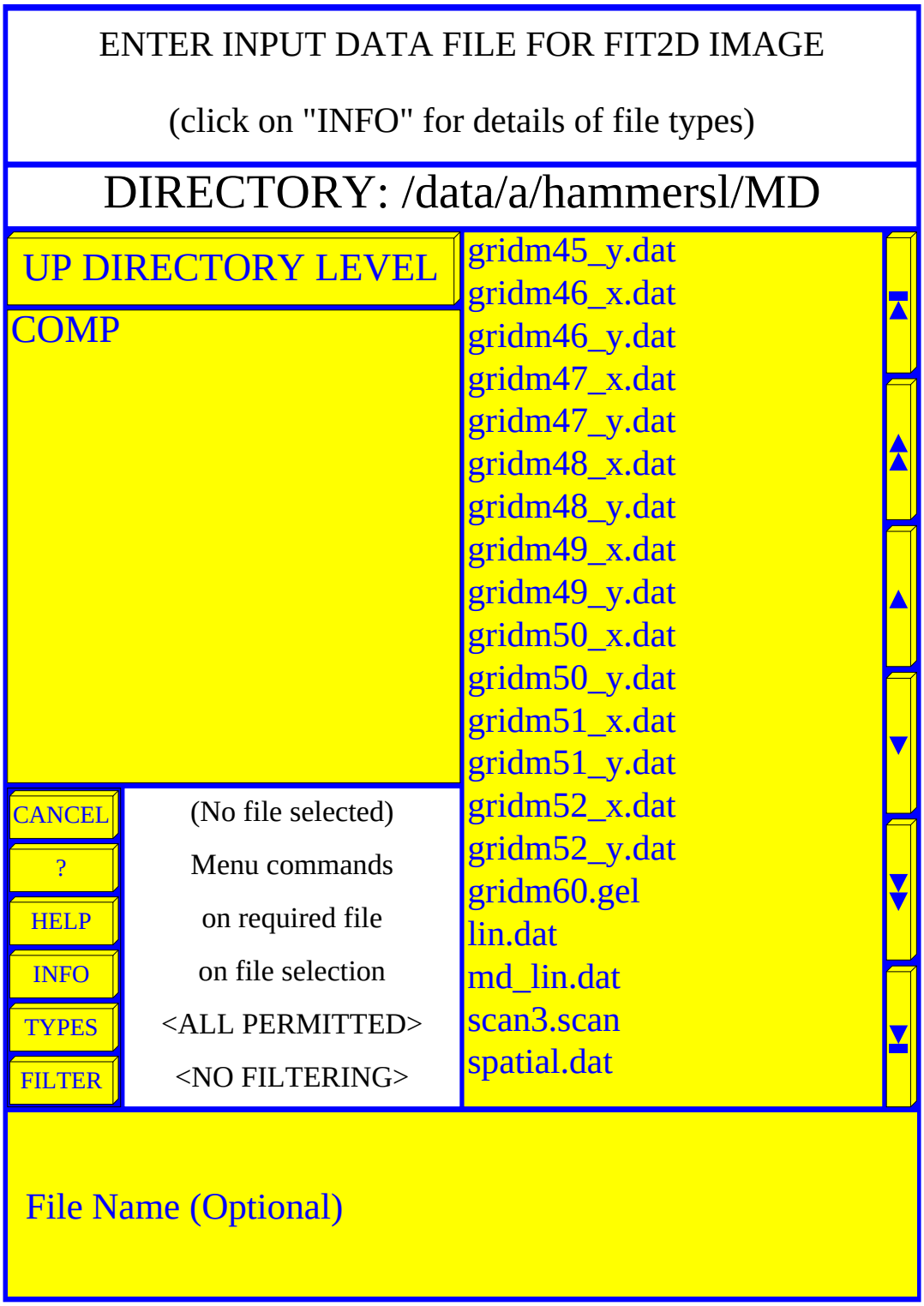
? This gives a short phrase explaining the use of each of the buttons in the form

HELP This gives help information specific to the particular graphical form and the parameters being controlled. This gives an overview of how the parameters control or vary the operation.

3.7 File Selection and input

Another often repeated operation is the selection of a file for input of data (or for other purposes). To facilitate this operation a "file selection tool" is used. This allows the directory tree to be searched as well as displaying available files. An example is shown in Figure 6.

Figure 6: The File Selection Tool



At the top of the display is the "prompt" describing the type of file to be entered. The line underneath shows the current directory and this text will change if the directory is changed. Underneath, on the left hand side is the "UP DIRECTORY" button which allows the directory tree to be mounted. The sub-directories of the current directory, if any, are displayed in a region on the left of the window and any files are displayed on the right. Both, when present, are displayed in blue writing on a yellow background (i.e. active),

and if more directories of files are present than can be displayed "scroll" buttons appear to allow the list to be scrolled. Clicking on a directory name will change the current directory, and a new list of sub-directories and files will be displayed.

As well as the directory and file selection regions, there is a region for text input. The text is displayed at the bottom of the window. Thus, rather than clicking on a displayed file name it is possible to enter file names from the keyboard. The <ENTER> (or <RETURN>) key is used to complete entry of the name. If a valid file name is entered then this will be the selected file. If a valid directory name is entered, then the directory will be changed and a new set of files and sub-directories will be displayed. The <TAB> key may be used for automatic file name completion. Pressing the <TAB> key will search for possible completions of the entered text. If only one file or directory is possible, then this will be automatically selected. In case that more than one completion is possible then common characters will be automatically completed as far as possible. Immediately using the <TAB> key again will update the lists of files and directories which are possible completions of the file name. (Note: when this file completion facility is used, any filters or type matching (see below) is ignored.)

Additionally the following buttons follow the directory list:

CANCEL To cancel the selection of a file. This will cancel the input and return to a menu.

? List of button commands and short explanatory text.

HELP Help information on the particular type of file required.

INFO Help information on the use of the file selection tool.

TYPES A sub-set of files may be selected and displayed by specifying only files of a certain file extension to be "selectable". By clicking on the **TYPES** button the require extension of the files may be entered. e.g. `image` may be entered to select only MarReARCH image plate scanner files. When a file type has been defined it is displayed in the information box, to the right of the **TYPES** button. In the example no file types have been defined so the information box contains the text `<ALL PERMITTED>`.

FILTER A sub-set of files may be selected and displayed by defining a file "filter". The asterisk is used as a "wild-card" so that a pattern may be defined and only files matching the pattern will be displayed in the file window. e.g. If "f*.dat" is entered, then only files which start with lower case "f" and are of file extension "dat" will be selectable. Any active filter is displayed to the right of the "FILTER" button. In the example no file filter has been defined so the information box contains the text `<NO FILTERING>`. (It should be noted that both "FILTER" and "TYPES" will act together if they are both activated.)

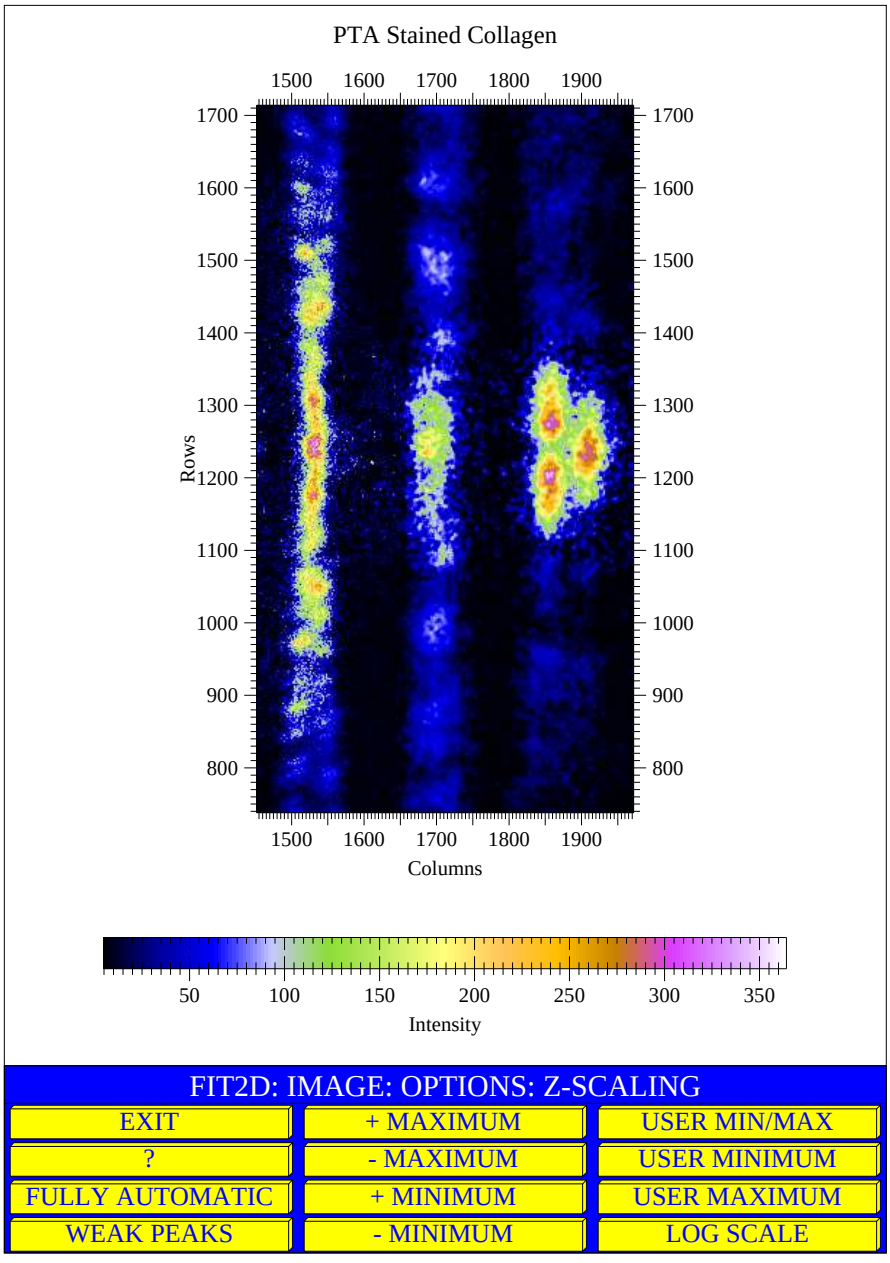
Having selected a file **FIT2D** will try to use the file extension to determine the data format. If the file extension is of a number of "known" types the appropriate input code will automatically be tried. e.g. The Molecular Dynamics image plate scanners save files with the extension `gel`, so all such files are read in automatically. If a file type is unknown,

then a menu of available input formats is displayed, and the user can click on the correct format, or the **CANCEL** button if none are suitable. (The **BINARY** input type allows a very wide range of file types to be input, so long as the user knows the basic details of the format.)

3.8 False Colour Image Display and Intensity Scaling

The basis of all interactive data analysis in **FIT2D** is the false colour image display for 2-D data, or an X/Y Graph plot for 1-D data. Figure 7 is an example of the false colour display of 2-D data, together with the **Z-SCALING** menu for changing the manner in which pixel intensity values are displayed as different colours or grey levels.

Figure 7: Example of False Colour Image Data Display



The whole of the *ADR* is displayed as an image even if there are less screen pixels than data elements, which is often the case. Figure 7 shows the default style, but the style of the graphics may be varied by using **OPTIONS** menu (see Section 3.10). For general data analysis the style options are generally not important, with the exception of the intensity or "z-scaling". It is necessary to understand the manner in which the data element values are changed into displayed pixel colours, to interpret the display images.

By default **FIT2D** uses automatic full range linear scaling of the *ADR* data values. This means that the minimum value and maximum of the *ADR* are found and all values in-between are converted to a colour (black, white and grey's are colours) in the current colour table, using linear interpolation.

With automatic range intensity scaling the whole of the range of data values will always be visible, although some images with a very few very extreme intensity values may not appear very interesting. Full range automatic scaling may be selected using the **FULLY AUTOMATIC** button. As the *ADR* changes then so do the current minimum and maximum values, and same intensity values may be displayed using different colours. This can be confusing and undesirable in some circumstances.

"Fixed" range intensity scaling is also available. With completely fixed scaling the minimum and and maximum data values which correspond to the extremes of the colour table are fixed to some values. Thus, as the *ADR* changes the intensities displayed with particular colours stay fixed. However, if data values outside the defined range are encountered then they are just displayed using the colour corresponding to the minimum or maximum of the range. Fixed range intensity scaling may be selected using the **USER MIN/MAX** button, which will prompt for the minimum and maximum values of the displayed range of intensity values. It is also used when the **+ MAXIMUM**, **- MAXIMUM**, **+ MINIMUM**, or **- MINIMUM** buttons are used. These increase or reduce the minimum or maximum values of the range by 10%.

In addition to these two scaling modes **FIT2D** allows the following additional modes:

USER MINIMUM This uses an entered fixed minimum value for the displayed intensity range, but the maximum is automatically defined by the *ADR* data values.

USER MAXIMUM This uses an entered fixed maximum value for the displayed intensity range, but the minimum is automatically defined by the *ADR* data values.

WEAK PEAKS This selects reduced range automatic scaling. This is designed to display the weak features of diffraction data. Often full range scaling is not very informative since the intensity range is dominated by a very few strong reflections, and little detail is seen in the majority of the image. In this mode, the average value and standard deviation of intensity values are calculated in several regions of the image. The minimum of the display intensity range is then calculated from the minimum of the mean minus three times the standard deviation, and the maximum is set to the maximum of the mean plus five times the standard deviation from each of the regions.

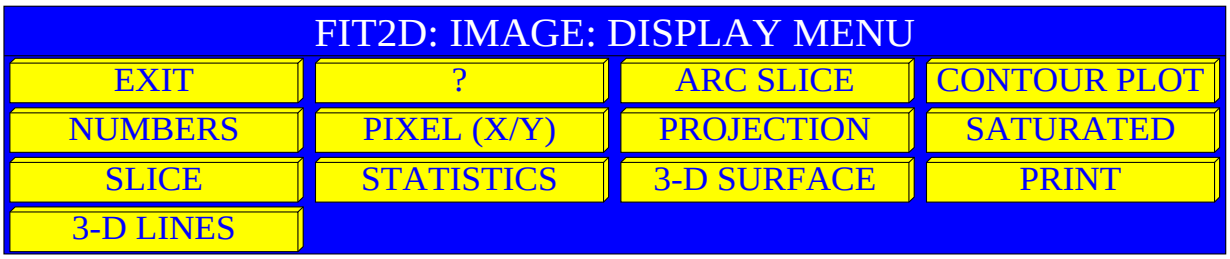
As well as linear scaling, logarithmic scaling is available through the **LOG SCALE** button. When logarithmic scaling is enabled this button is replaced by the **LINEAR SCALE**

button. If zero or negative range limits are encountered when logarithmic scaling is being used, a small positive value will be used instead.

3.9 Graphical Display Possibilities

The **DISPLAY** menu allows other graphical display possibilities in addition to the standard false colour image display. Figure 8 shows the menu button choices.

Figure 8: The Display Menu



The buttons have the following functions:

EXIT Exit display menu

? Explanation on each of the menu options

ARC SLICE Define an arc, by entering the end coordinate and a coordinate on the arc, and calculate the 1-D "slice" along this arc. This is displayed as an X/Y graph and is saved in the memory.

CONTOUR PLOT Display the *ADR* as a contour plot

NUMBERS Allow the user to click on coordinates and output the 11 by 11 square of pixel values around the coordinate. (If a log file is open this will be saved in the log file.)

PIXEL (X/Y) Allow the user to click on pixels and display information on the coordinate and pixel values. If the geometry of a diffraction experiment has been defined, then angles and equivalent D-spacings will be calculated and displayed.

PROJECTION Allow the user to define an arbitrary line, and a region either side of the line, and "project" the pixel values in the defined region onto the line. This is displayed as an X/Y graph and is saved in the memory. (This is similar to **SLICE** but allows averaging over several or many pixels.)

SATURATED Count the number of pixels with an intensity greater or equal to an input value.

SLICE Allow the user to click on two coordinates to define an arbitrary "slice" through the data. The nearest pixels to the line define the output data values. This is displayed as an X/Y graph and is saved in the memory.

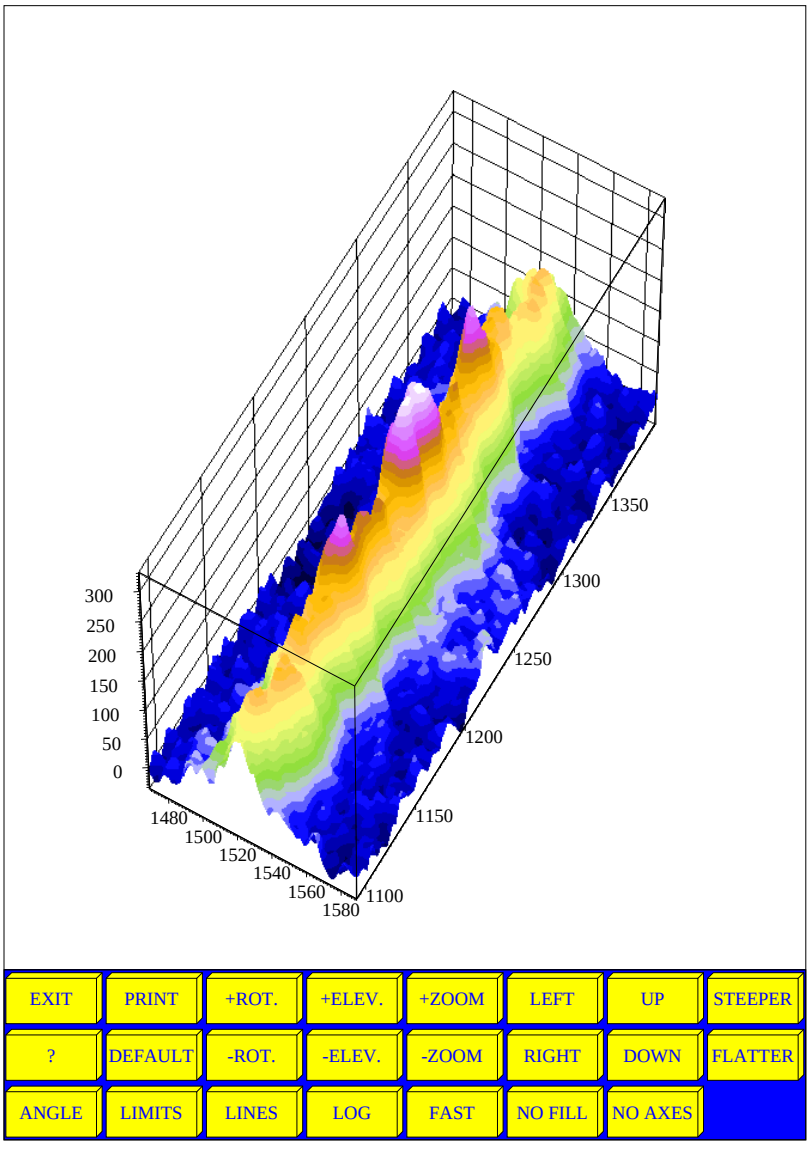
STATISTICS Allow the user to define a arbitrary polygon region, and calculate a number of statistics within this region. e.g. number of pixels, total intensity, mean and standard deviation values. By first defining a background region, followed by a region around a diffraction peak, a primitive form of peak integration is available.

3-D SURFACE The *ADR* is displayed a 3-D surface perspective view. The surface may additionally be coloured by the current colour table. The viewing angle and many aspects of the display are controlled by a menu. An example of the display and menu are shown in Figure 9.

PRINT Save the current graphics display to a PostScript file.

3-D LINES Draw the *ADR* as a 3-D line plot, with hidden line removal.

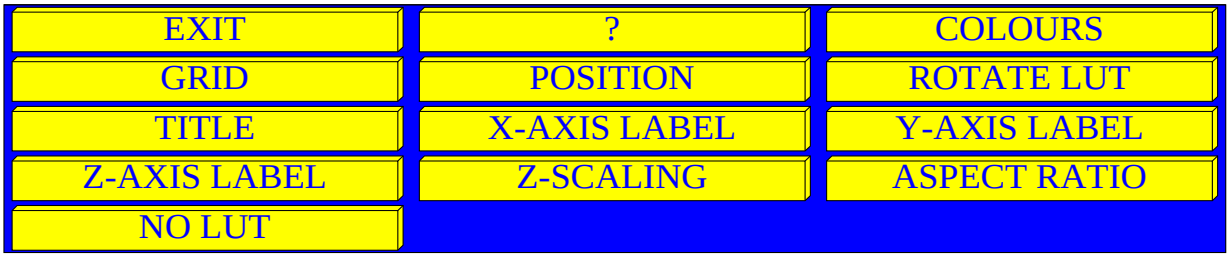
Figure 9: Example of the 3-D Surface "Viewer"



3.10 Display Style Options

The **OPTIONS** menu is shown in Figure 10.

Figure 10: The Graphics Display Style Options Menu



The following buttons are available:

EXIT Exit display menu

? Explanation on each of the menu options

COLOURS Choice of different colour tables e.g. grey scale, inverse grey scale (black is more intense), "temperature". Several of the different colour tables are shown throughout this paper.

GRID Add or remove horizontal and vertical grid lines to the image display

POSITION Change the position for the image display in the graphics window

ROTATE LUT Change cyclically the colours used to display different intensity levels. This can emphasise different features in the data.

TITLE Change the title of the image.

X-AXIS LABEL Change the text used to label the X-axis of the image.

Y-AXIS LABEL Change the text used to label the Y-axis of the image.

Z-AXIS LABEL Change the text used to label the intensity scale (Z-scale) of the image. This is drawn with the look-up table intensity scale

Z-SCALING Change the mapping of data values to pixel colours (see Section 3.8 for further details).

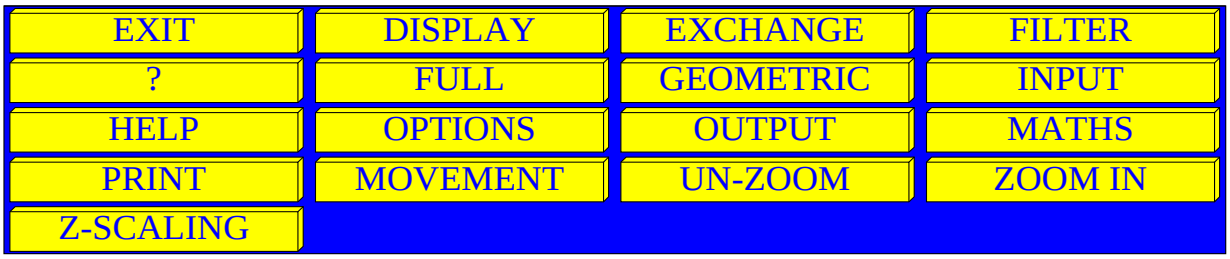
ASPECT RATIO Change the "aspect ratio" for image display. By default pixels are drawn square, so if the *ADR* is rectangular the image is rectangular. When many more pixels are defined in one direction than the other, it can be more useful to allow the pixels to be rectangular and allow the image to fill the full region available. These modes can be toggled.

NO LUT Don't draw the "look-up table". When clicked, this is replaced by the button **DRAW LUT**.

4 FIT2D for General Image Processing Operations

The **IMAGE PROCESSING (GENERAL)** interface contains commands which allow normal mathematical operations and geometrical transformations to be applied to image data. Filtering operations are also available. The **IMAGE PROCESSING (GENERAL)** interface main menu is shown in Figure 11.

Figure 11: IMAGE PROCESSING (GENERAL) Interface Menu



As an example of using this menu the basic operation of subtracting one image from another is detailed. e.g. Subtracting a dark current image from a diffraction image. The following commands are used:

INPUT Input dark current image

EXCHANGE Place the dark current image in the memory

INPUT Input diffraction image

MATHS Enter **MATHS** sub-menu

SUBTRACT Subtract the memory image from the current data array image

EXIT Exit **MATHS** sub-menu

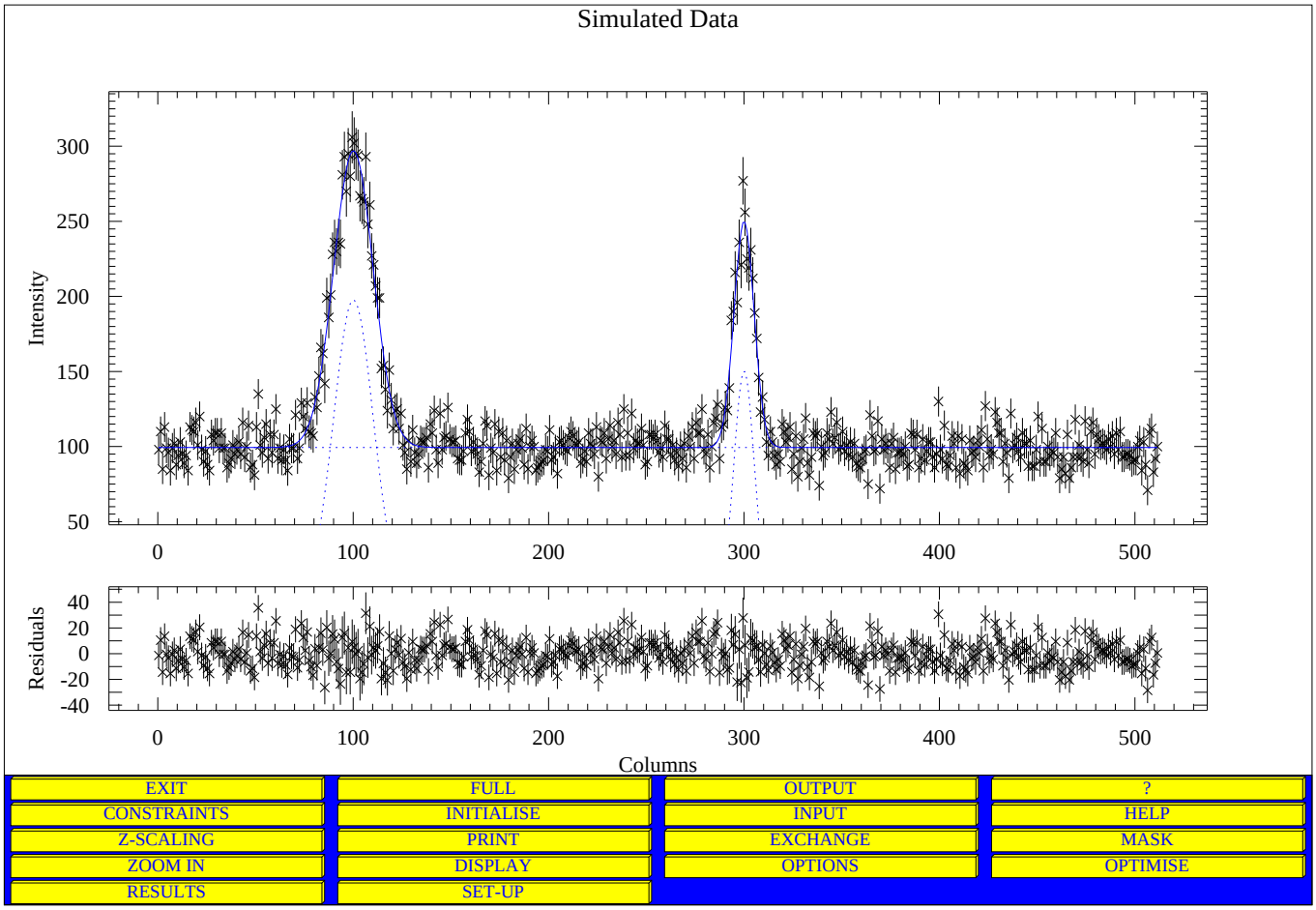
OUTPUT Save dark current subtracted data in a file

(It may be noted that the memory still contains the dark current image, so another diffraction image could be input without having to re-input the dark current.)

5 FIT2D for 1-D "Peak" Fitting

Both 1-D and 2-D models may be fitted to experimental data using a variety of minimisation methods. 1-D model fitting is available within the **MFIT (MULTIPLE 1-D FITTING)** interface. The main menu is shown in Figure 12.

Figure 12: A Fitted 1-D Peak Model



Fit models may be defined using the **INITIALISE** sub-menu. A fit model may be constructed from an arbitrary number of peak functions (Gaussian, Lorentzian, and Voigt) and trigonometric or exponential decay functions, and a polynomial function of specified order. The **SET-UP** form allows the control parameters to be varied e.g. Weighted fitting or not (provided variance arrays exist). The **CONSTRAINTS** form allows individual model parameters to be constrained, or set to freely vary. The **OPTIMISE** button optimises the model to the data, according to the current control parameters.

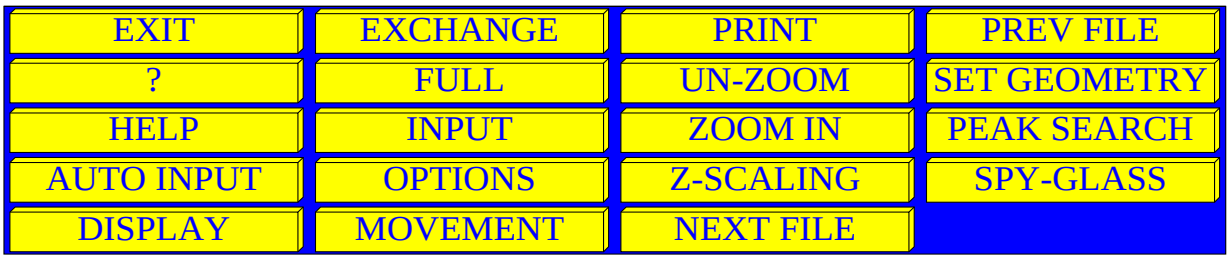
The display of the fitted model and the data is shown in the upper X/Y graph in Figure 12, and beneath is a display of the residuals. This graphical display of the results may be saved as a PostScript file using the **PRINT** button (while this is displayed in the graphics window).

6 FIT2D for On-line Display

As well as an "off-line" data analysis program, **FIT2D** may be used as an "on-line" data display and manipulation program. Figure 13 shows the **ON-LINE CRYSTALLOGRAPHY** interface. Data may be input and displayed as with the other menus, and additionally the **AUTO INPUT** button allows files in a sequence to be automatically input and displayed as they are saved by the data acquisition system. An optional "alarm"

may be set which will ring if the time for an image to appear is greater than a specified length.

Figure 13: ON-LINE CRYSTALLOGRAPHY Interface Menu

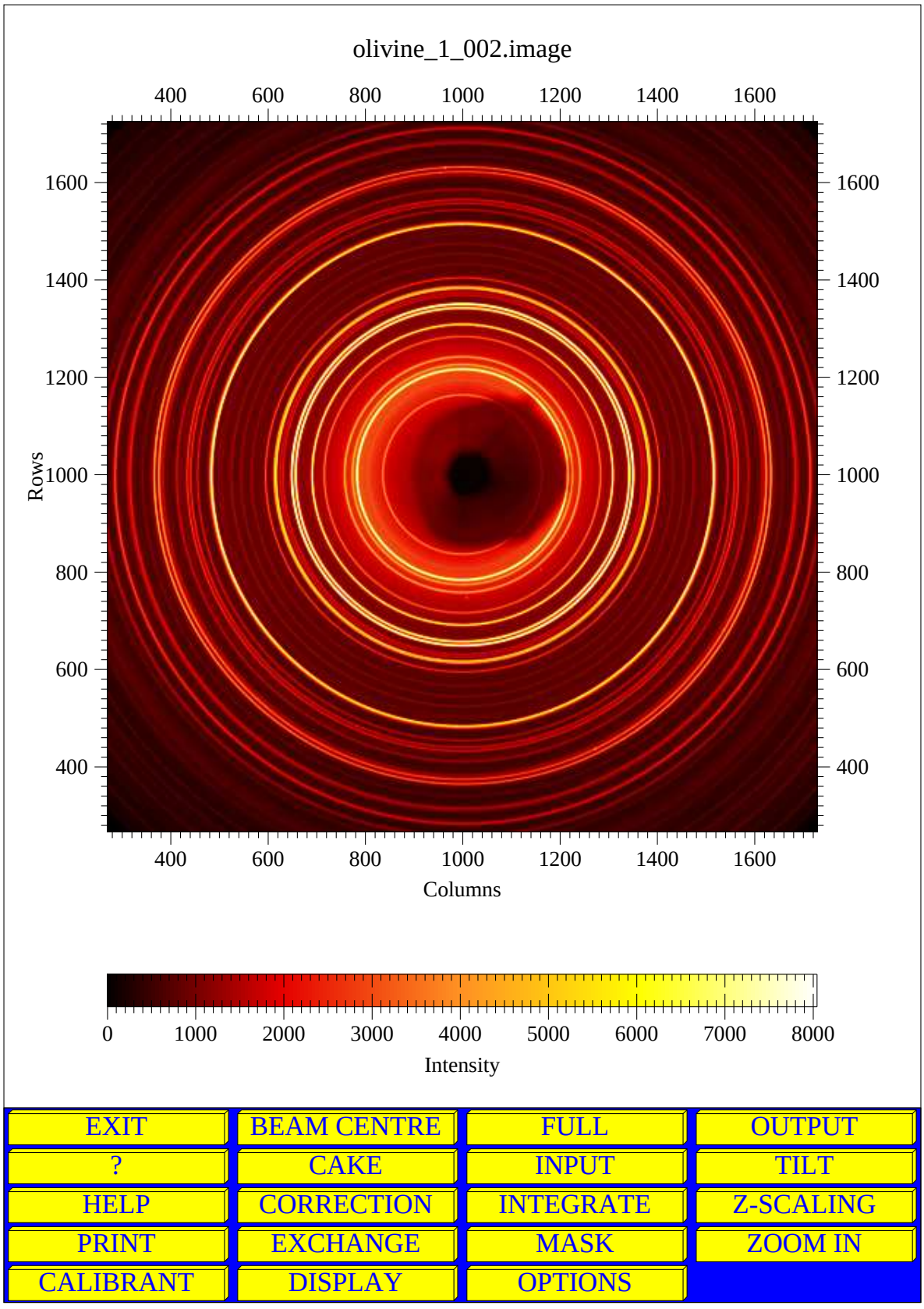


7 FIT2D for Powder Diffraction with Area Detectors

The **POWDER DIFFRACTION (2-D)** interface allows Debye-Scherrer rings which have been recorded of a 2-D detector to be integrated to the equivalent of a 2-theta scan with a zero-dimensional detector (Hammersley *et al*, 1996). An example of an input image and the main menu are shown in Figure 14. To aid with this the **MASK** sub-menu allows interactive definition of masked areas i.e. pixels which will be ignored during further operations. The **TILT** button allows the beam centre and any non-orthogonality of the detector to the main beam to be refined, and this tilt is used by the **INTEGRATE** command. Other types of output are also available. The **CAKE** sub-menu allows different regions of the image to be defined with graphical coordinate input for the starting azimuth, end azimuth and inner and outer radii of the integrated region. The number of output bins in both the azimuthal and the 2-theta or radial direction are user specified. This allows a variety of different output possibilities e.g. A number of different 2-theta scans, for different azimuth ranges; a 1-D profile of intensity of a ring as a function of azimuth (e.g. for texture studies); or a polar transform of the data. The output formats include one of the GSAS formats (Larson & Von Dreele, 1994), one of the Cerius formats[1], and a general purpose ASCII two column format, which may easily be edited.

[1]Crystallographic software package from Molecular Simultations

Figure 14: The Powder Diffraction Menu



The **CALIBRANT** button allows the fitting of the powder pattern from a standard calibrant sample. The known D-spacings of the diffracting planes allow the beam centre

and tilt to be defined with greater accuracy than with an unknown sample, and if reasonably high angle data is present allow the sample to detector distance, and the wavelength to be refined independently. This has been verified to agree to within the accuracy that may be obtained from scanning an absorption edge using a Ge detector on ID-30 at the ESRF.

8 The "Keyboard" Interface

The "Keyboard" Interface was the original user interface for **FIT2D** and is still preferred by some experienced users. The GUI is almost certainly the easier way for newcomers to learn to use **FIT2D**, but an experienced user can often enter commands more quickly at the keyboard than by moving a mouse around. Thus, most of the functionality of **FIT2D** is available through the command interface. Some functionality such as the calibration menu is still only available through the keyboard menu at present, but gradually all commonly used functionality will be made available within the GUI.

Contrary to the GUI, the output of many commands is often in the memory. Where an operation can work on a simple pixel by pixel basis, the output is in the current data array, but for operations where one pixel may affect several pixels in the result, the output is in the memory. The Reference Manual (Hammersley, 1995) contains tables of the commands showing where any output appears.

About 120 commands are available in the main keyboard menu allowing more data analysis operations and further graphical output style control than presently available in the GUI. Additionally, the `CALIBRATION` command enters a sub-menu of commands specialising in detector distortion calibration, and the `FIT` command enters a sub-menu of commands for 2-D model fitting.

8.1 The `CALIBRATION` Sub-Menu

The `CALIBRATION` sub-menu contains commands to characterise and calibrate detector distortions such as intensity linearity, spatial distortion, and non-uniformity of intensity response (Hammersley, Svensson, & Thompson, 1994, Hammersley *et al*, 1995, Moy *et al*, 1996). Raw images of calibration grid exposures may be treated to define spatial distortion correcting functions, and thus image data may be re-binned from a distorted grid to an ideal regularly spaced grid. Once defined the function coefficients may be saved to a file for later re-use, and within the GUI.

The re-binning operation involves calculating the covered area in the ideal output pixel grid of each of the pixels from the raw distorted image. This involves much calculation and is a relatively slow procedure. To make correction of multiple images much faster it is possible to calculate a look-up table of pixel shifts and the relative contributions of each input pixel to the target and surrounding pixels. Once calculated the correction is 10 times faster than without the look-up table (2.5 seconds for an image of 1242×1152 on an HP-755 125MHz system).

A typical use of the `CALIBRATION` menu is for the correction of diffraction images from detectors with spatial distortion and non-uniformity of intensity response, for macromolec-

ular and small molecule crystallography. Up to hunderds of images are input, corrected for the distortions, and output in an appropriate file format, for integration using a standard integration program e.g. **DENZO** (Otwinowski 1993). This correction is performed automatically using macro files (which are described in the Reference Manual).

8.2 The `FIT` Sub-Menu

The `FIT` sub-menu allows similar model fitting to that available within the **MFIT (MULTIPLE 1-D FITTING)** interface, but with 2-D functions rather than 1-D functions. Commands also exist which allow the same functionality which is available through the **POWDER DIFFRACTION (2-D)** interface.

9 Command line Options

To make **FIT2D** more versatile and for easier use, and for "batch" mode applications a number of command line options are available. **FIT2D** may be started with "landscape" graphics instead of the default "portrait" graphics, the "keyboard" interface may be used instead of the GUI, and the graphics system may be "turned-off", (this is only valid when the "keyboard" interface is being used). (Complicated command line calls may be made simpler and shorter through the use of "aliases".) The manner in which the command line options work is best illustrated by an example. The following is an example command line which may be used:

```
> fit2d -dim1152x1482 -land -key
```

The command line would start **FIT2D** in the following manner:

`-dim1152x1482` Starts **FIT2D** with programs arrays of 1152 pixels in the X-direction and 1482 pixels in the Y-direction

`-land` Starts the graphics in "landscape" mode i.e. the graphics window appears wider than its height and PostScript output is printed in this orientation.

`-key` Starts-up **FIT2D** with the "keyboard" interface instead of the default GUI.

The different options are described in greater detail below:

9.1 Program Dimensions

The size of the internal program arrays may be specified on the command line by using the `-dim` or `-dimensions` option. (`-DIM`, `-DIMENSIONS`, `+dim`, `+dimensions`, `+DIM`, and `+DIMENSIONS` are equivalent.) This avoids having to edit the **DIMENSION OF PROGRAM ARRAYS** form, and unless other options are used starts-up **FIT2D** with the

memory array, but without variance arrays. The `-dim` option should be immediately followed by the size of the program arrays to create in the X-direction, and the size of the arrays to create in the Y-direction separated by an `x` or an `X`[2].

e.g. The following operating system command line would start **FIT2D** with current data and memory arrays of 500 elements in the X-direction and 700 elements in the Y-direction, but no variance arrays:

```
> fit2d -dim500x700
```

(The `DIMENSIONS` command may be used within a macro as an alternative method of changing the size of program arrays from the default values.)

9.2 Variance Arrays

By default variance arrays will **not** be created if the `-dim` option is used to define the program array sizes. If this is not suitable then their creation may be specified on the command line by using the `-var` or `-variances` option. (`-VAR`, `-VARIANCES`, `+var`, `+variances`, `+VAR`, and `+VARIANCES` are equivalent.)

9.3 "Landscape" Output

By default **FIT2D** always produces vertical A4 ("portrait") output on the screen and on paper. The page format can be changed to horizontal ("landscape") output by using the option `-landscape` or `-land` on the command line when running the program (`-LANDSCAPE`, `-LAND`, `+landscape`, `+land`, `+LANDSCAPE`, and `+LAND` are equivalent.) i.e. To start up **FIT2D** in "landscape" mode enter:

```
fit2d -landscape
```

Since the page layout has been designed for "portrait" mode, some re-arrangement of the page layout may be necessary.

9.4 "Keyboard" Interface

By default **FIT2D** starts up with the GUI, but the command-line option `-key` or `-keyboard` may be used to obtain directly the "keyboard" interface (see Section 8). (`-KEY`, `-KEYBOARD`, `+key`, `+keyboard` `+KEY`, and `+KEYBOARD` are equivalent.)

[2]The multiplication sign cannot be used as this has a special purpose for the Unix operating system.

9.5 No Graphics

Sometimes it is required to run **FIT2D** without a graphics output window. This may be required because the output window cannot work properly across the network. It may also be for batch processing, which doesn't require any display. The `-nographics` command line option allows **FIT2D** to start-up and stop without the graphics display window being opened. Clearly, this makes many commands invalid, and is only valid when the "keyboard" interface is being used. (The `-nographics` option can be shortened to `-nogr`, and alternative forms are equally permissible: `-NOGRAPHICS, +nographics, +NOGRAPHICS`).

9.6 Other Options

Other options exist which are not relevant to the GUI, but which are briefly described here for completeness. The `-mac` option allows a macro file to be run to be specified from the command, and the `-sym` option allows "program symbol" values to be specified. These two options allow **FIT2D** to be used in a very flexible manner from an operating system command file (a "script"), for batch processing. (The saving of data analysis command sequences into macro files and running these macro files on sequences of image and/or files, from both within **FIT2D**, and externally, are described in the Reference Manual (Hammersley, 1995).)

10 Trouble-Shooting

10.1 Command not found

If when you enter **FIT2D** the system responds:

```
fit2d: Command not found.
```

this means either that **FIT2D** has not been installed (properly), or that your `PATH` variable is not properly defined. Normally, **FIT2D** should be found in the `/usr/local/bin` directory. Thus, your environment variable `PATH` should include this directory.

See your system administrator, if you cannot rectify the problem.

10.2 Failure in FIT2D Start-up

If **FIT2D** fails to start-up properly, this can be due to one of a number of reasons, which are outside the control of **FIT2D**.

If the welcome text appears in the terminal window, but the graphics window does not appear, and a message similar to the following appears, it is necessary to authorise another workstation (client) to communicate with your workstation (you have a remote login), or perhaps the "`DISPLAY`" environment variable is not defined correctly.

```
Xlib: connection to ``expgf:0.0'' refused by server
Xlib: Client is not authorized to connect to Server
Failed to open X-11 display
ERROR: X-11 workstation not opened successfully (EXPG_LG_OPEN_WK)
```

To add the remote workstation to the list of allowed communicating workstations use `xhost` e.g. if the remote workstation is called `expga`, it is necessary to type the following command on **your** workstation:

```
xhost +expga
```

If the problem is not easily cured, contact your system administrator for further advice and assistance.

In addition to the above problems, the graphics window can also fail to be created if the X-terminal window is of the wrong type. The **FIT2D** graphics are presently only written for "8-bit plane pseudo-colour X-windows" (X-11 terminology). If this is not the case, the following error message will be output to the terminal window:

```
ERROR: At present the LG system is only written to cope with 8-bit
       plane pseudo-colour X-displays. Your display default window is
       not presently supported. If you can change the default behaviour
       to 8-bit plane pseudo-colour, LG should be able to work.
       Otherwise extra code will need to be added to either create a
       pseudo-colour window (a medium sized task), or add functionality
       to cope with true colour or monochrome displays (a very large task).
```

In almost all cases it should be possible to change the set-up of the X-display, or X-emulation, such that the **FIT2D** graphics can work (local documentation will need to be consulted).

If the following message appears, or one similar, it means that the **present** available system virtual memory size is too small for the program to start.

```
Not enough memory
```

This could be because you have accidently demanded enormous internal arrays to be created. If this is the case, then re-starting **FIT2D** and specifying smaller arrays may solve the problem. If not, then some system parameter may need to be changed; see your system administrator (see the next section).

(This problem may appear and go away owing to other users starting and finishing processes, and therefore variable amounts of memory being available for **FIT2D**.)

FIT2D may also produce a message explaining that it is "protected by a software key, which is presently not defined.". After exiting the text viewer, a message will be left in the terminal window with information needed to calculate the "key value". To use **FIT2D** users must register and are given a personnel code or "key", and simple instructions on how to set the key value. If this message occurs it probably means that this procedure has not been carried out, and a key needs to be obtained.

10.3 Dynamic Memory Allocation Failure

FIT2D uses dynamic memory (or virtual memory) allocation to allow as much flexibility as possible in the size of data that can be treated. This has obvious advantages, but also a disadvantage. 2-D dimensional data (images) from modern detector systems can be very large, so the demands of **FIT2D** for memory allocation can be similarly large. Users are recommended to try to use the minimum of memory necessary for their data. At start-up the initial size of program arrays is set.

If at any stage during the running of FIT2D the user gets a message that memory allocated has failed, it must be assumed that the previous command has failed.

The amount of memory available on a computer system varies as the demands of other uses and processes changes. Thus, a program which works one day, may fail another. The available memory is determined by system parameters, so see your system administrator if there are persistent problems.

It should be noted that the memory does not have to be RAM, as modern computer systems allow hard-disk files to used as "virtual memory". However, if sufficient RAM is not available, a performance degradation will be noticed.

10.4 No Image Output

If the image display appears to be wrong, or nothing appears to be displayed it is likely that you have previously used the `Z-SCALE` command to set the scaling mode to semi-automatic, or fixed scale. If the present image is outside the range then the image will appear totally uniform (black or white for most of the colour tables).

To overcome this problem, use the `Z-SCALING` button to set the scaling mode back to fully automatic, or set a reasonable range.

11 Availability

FIT2D is available free of charge to academic research in return of an undertaking for appropriate acknowledgement of use of the program and the methods. Commercial users will be requested to pay a yearly support fee, which is presently 5000 Francs per year.

FIT2D executables for various machines and operating system types can be obtained by anonymous ftp from www.esrf.fr. DEC-ALPHA OSF1, DEC-MIPS Ultrix 4.4, HP-700 HP-UX V9, IBM RS6000 AIX 3.2.1, Silicon Graphics MIPS R4000 Irix 5.3, and SunOS V4 and V5 (Solaris) versions are presently available. (Silicon Graphics Irix 6.2 and Linux versions should soon be available.)

Users of **FIT2D** are required to sign a "Conditions of use" form and return it by mail, or initially by Fax. The "Conditions of Use" are intended to allow freedom of use for personal academic research provided suitable acknowledgement of the program and methods is given.

The protection allows the usage of **FIT2D** to be monitored and means that users will be informed of improvements, important bug fixes, etc.

12 Future Developments

FIT2D will continue to be developed to enhance usability and new data analysis methods will be added. Further interfaces are likely to be added, to extend the GUI to access more of the existing functionality the the "keyboard interface" e.g. the functionality of the `CALIBRATION` sub-meu.

Nevertheless, the same basic user interaction of the GUI as here-in described will continue unchanged, although extra buttons may well appear in some of the menus !

13 Acknowledgements

The author would like to thank Tim and Linda Wess for permission to show the collagen data, which was collected at the SRS, Daresbury, England; Andy Fitch and Phil Pattison for permission to show the olivine data, which was collected on the Swiss-Norwegian Beamline at the ESRF, France; Olof Svensson and Martin Kunz for proof reading and testing the paper and many helpful suggestions.

14 Appendix A: Colour Usage

FIT2D is presently optimised for use on colour displays ("X-displays") with a maximum of 256 available displayable colours at any one time ("8-bit plane pseudo-color"). These are still the vast majority of available displays, although "24-bit plane" "true colour" displays are starting to become more common.

An individual window e.g. the **FIT2D** graphics window, is allowed to define up to 256 individual colours, which may be different to those which are defined by another window. When the user clicks in a new window[3], then the colours defined by the new window replace the colours previously used. This is termed "colour flash", and previously readable text may become unreadable.

"Colour flash" may be avoided if all windows share the same colours, but this not suitable for efficient false colour image display e.g. grey scale, of large images. **FIT2D** defines its own colours. If there are enough remaining undefined colours in the default colour table, then this will be used and "colour flash" will not occur. Unfortunately, window managers and other programs are using more and more colours simply for button display and shading, so often very few colours levels are undefined. In this case, a new individual colour table is created and used by **FIT2D**. To minimise the problem cased by "colour flash" the first 40 colours in the existing colour table are copied and are left unchanged.

[3]Or in some cases by simply moving the cursor into a new window, depending on how the "window manager" has been defined.

On some X-displays this keeps text readable, but unfortunately other X-displays use other colour levels for displaying text so still suffer from "colour flash".

It is worth noting that some programs, have not been written to create their own colour tables, so will not work after **FIT2D** has been started. However, if such programs are started first, and **FIT2D** afterwards, the two can be used simultaneously.

Within the colours defined by **FIT2D** eight colours are always fixed, and are used for menu display and user prompts (white, black, red, green, blue, yellow, cyan, and magenta). The remaining colour levels are filled with the required false colour table e.g. geographical or inverse grey-scale. Different colour tables contain different colours so graphical objects which are not coloured in one of the eight constant colours will change colour when the colour table is changed. (This is why the "FIT2D" logo background sometimes changes colour just after **FIT2D** is started. The background is orange, which is defined in the default colour table, but not for example in a grey-scale.)

References

- Hammersley, A. P. (1989). *Syn. Rad. News*, **2**, 24-26.
- Hammersley, A. P., Svensson, S. O. & Thompson, A. (1994). *Nucl. Instr. Meth.*, **A346**, 312-321.
- Hammersley, A. P. (1995) *ESRF Internal Report*, **EXP/AH/95-01**, FIT2D V5.18 Reference Manual V1.6.
- Hammersley, A. P., Svensson, S. O., Thompson, A., Graafsma, H., Kvick, Å. & Moy, J-P. (1995). *Rev. Sci. Instr.*, **66**, 2729-2733.
- Hammersley, A. P., Svensson, S. O., Hanfland, M. Fitch, A. N., & Häusermann, D. (1996). *High Pressure Research*, **14**, 235-248.
- Larson, A. C. & Von Dreele, R. B. (1994). General Structure Analysis System (GSAS) Manual, **LANSCE, MS-H805**, Los Alamos National Laboratory, Los Alamos, NM 87545
- Moy, J-P., Hammersley, A. P., Svensson, S. O., Thompson, A., Brown, K., Claustre, L., Gonzalez, A. & McSweeney, S. (1996). *J. Syn. Rad.*, **3** 1-5.
- Otwinowski, Z. (1993). *in Proceedings of the CCP4 Study Weekend: "data Collection and Processing", 29-30 January 1993*, compiled by L Sawyer, N Isaacs and S Bailey, SERC Daresbury Laboratory, England, 56-62.