Astronomical Data Analysis Software and Systems VI
ASP Conference Series, Vol. 125, 1997
Gareth Hunt and H. E. Payne, eds.

IDL Library Developed in the Institute of Solar-Terrestrial Physics (Irkutsk, Russia)

S. K. Konovalov, A. T. Altyntsev, V. V. Grechnev, E. G. Lisysian, and G. V. Rudenko

Institute of Solar-Terrestrial Physics, Lermontov St. 126, Irkutsk, Russia 664033, E-mail: root@sitmis.irkutsk.su

A. Magun

Institute of Applied Physics, University of Bern, Switzerland, 3012 Bern Sidlerstrasse 5, E-mail: magun@sun.iap.unibe.ch

Abstract. We process and analyze data provided by the SSRT interferometer using IDL. Thanks to the well-known capabilities of IDL, this has expedited our research. Special requirements, convenience, and efficiency led us to the creation of an expanding library of IDL routines and programs, described herein.

1. Introduction

We have been processing and analyzing data provided by the Siberian Solar Radio Telescope (SSRT) (Smolkov et al. 1986). Much of these data required development of special techniques for processing and previewing. Using the well-known capabilities of IDL, we were able to speed up our research. However, a variety of routines had to be developed due to the specific needs of our research and the features of our instrument, as well as our desire to make processing still more convenient and efficient. Having started this work in collaboration with the Institute of Applied Physics (Bern), we continued it in Irkutsk. Coming up with our own IDL installation allowed us to expedite our work. As a result, a whole set of routines and programs supporting our programming has emerged (Konovalov et al. 1997). This set contains special routines for our use, as well as routines which could be of common interest. This library is being currently expanded.

2. Overview

In response to inherently specific features of our instrument, the SSRT, and to our research needs and preferences, we have developed, supplemented, and corrected some routines from the standard IDL library. Since our IDL library contains more than 100 routines and functions, we can present here only an overview of these routines, classifying them by category, and briefly describing their capabilities. The routines can be used on UNIX and MS Windows platforms, and all can be run under IDL 3.0.1 or later versions.

3. General Routines and Functions

These routines and functions were written for specific needs, but could be generally useful:

String-type variables manipulation: We have added functions which allow (*i*) finding a given substring in a string, (*ii*) replacing it with a given model of arbitrary length, and (*iii*) splitting a string containing any delimiter (e.g., whitespace) into substrings.

Files and file names manipulation: These routines provide: (*i*) conversion of text files between UNIX and MS Windows; (*ii*) adding an End-Of-File marker; (*iii*) extracting a short file name, extension, file name without extension, and directory name from the full path name; (*iv*) automatically recognizing the type of some graphics files (GIF, TIFF, BMP, or FITS); (*v*) creating a new file name, given a wildcarded pattern; (*vi*) an interactive tool for deleting a file, etc.

Mathematical functions: We have developed routines for rapidly evaluating some often-required functions: (*i*) SIGN and SINC ($\sin x/x$); (*ii*) recognizing even numbers; and (*iii*) calculating the Gaussian function. Furthermore, some operations for array manipulation have been implemented: (*i*) conversion of 1-D subscripts into 2-D; (*ii*) extracting subarrays; and (*iii*) searching for coinciding elements in different arrays.

Graphics windows manipulation: We have developed routines expanding the standard IDL tools with the following capabilities: (*i*) bringing all existing graphics windows to the front; (*ii*) deleting all existing graphics windows; (*iii*) displaying current cursor coordinates within a selected graphics window; (*iv*) saving and restoring scaling of the axes in a given window; and (*v*) converting a cursor into another shape.

Plotting routines: We have developed the routines which: (*i*) plot a graph versus time expressed in hours, minutes, and seconds; (*ii*) plot a graph of a fragmented array; (*iii*) plot a straight line of a given slope crossing a given point; (*iv*) plot a given arc or circle; (*v*) overplot a triangle-shaped marker onto a plot or image; (*vi*) display an array in the brightness representation enclosed by axes; and (*vii*) provide a box-shaped cursor for widget-based programs, etc.

Curve analysis: The developed routines allow (*i*) finding (and, if needed, marking on the plot) local extremes; (*ii*) selecting a local peak; (*iii*) calculating its width; (*iv*) finding the circle passing through three given points; and (*v*) finding a line bounding a given fast-oscillating curve (similar to the detection of a radio signal).

Graphics file manipulation and image processing: We use FITS files, so we have routines to handle the headers (searching for a keyword and returning the corresponding value, extracting of time and date from the header, etc.). We can automatically recognize the type of some graphics files (GIF, TIFF, BMP, or FITS) and display them on the screen. Routines were developed to interactively measure the coordinates of the solar disk's center and radius, and to save the image contained in a graphics file into a FITS file. There is also a routine converting a plot obtained from an image into a digital array, which can be saved for further processing.

Viewers: Some viewers have been developed for the SSRT data, as well as for the standard format graphics or digital files. There is an interactive array viewer which allows looking at data in various representation (wire-mesh surface, shaded surface, halftone image, contour, and their combinations), measuring profiles in two directions, and reading pixel values. Contour levels can be selected interactively. There is also a viewer for text files or string-type arrays entered on the command line (similar to XDISPLAY-FILE).

Date and time formats and conversion: A few routines and functions have been developed which are used in reading data records, in calculations, and in displaying date and time in a suitable form.

Astronomical calculations: Our routines: (*i*) compute an hour angle, a declination, and a radius of the Sun; (*ii*) calculate the heliographic, Carrington's, and plane coordinates, and perform transformation between them; (*iii*) compute the time interval between two events given in different calendar formats; (*iv*) transform the coordinates on the Sun according to differential rotation; and (*v*) rotate a flat image of the Sun around the polar axis.

Service widgets: They are used for inputting a date and time, a brief text, or coordinates. There are widgets for issuing a message, for obtaining an answer to a question, and for selection of one item from the list, etc.

Other: The following routines we have not classified: (*i*) conversion of a bit-serial array into a byte array; (*ii*) creation of a new system variable !UC containing universal constants; (*iii*) selection and preparing of the graphics device required for the next output (including setting of sizes, thickness, color, etc.), and closing them; and (*iv*) deallocation of all possible file units (convenient for debugging.)

4. Routines and Functions for SSRT Data Users

These routines are oriented to SSRT data processing.

SSRT: instrumentation, data processing: A range of routines is used to calculate the parameters of the SSRT beam and response as well as astrometric values; to read and convert the information contained in the original records; to execute some actions with original files; to perform data processing; and to simulate instrumental characteristics of the SSRT and its response.

Special routines: We have also some routines completely for our own convenience, such as converting ASCII codes from Roman into Russian, etc.

5. Supplemented and Corrected Routines

We have modified the routine CONGRID to transform small arrays correctly. We have also supplemented the routine IMAGE_CONT with various keywords, and edited it to display small arrays correctly. The routine READ_BMP has been enhanced to handle monochrome images.

We welcome any interest and cooperation.

Acknowledgments. We are grateful to Drs. B. Kliem (AIP, Germany), Yu. M. Rosenraukh, T. A. Treskov, S. V. Lesovoi, V. G. Miller, A. V. Bulatov, and B. I. Lubyshev (ISTP, Russia) for the helpful discussion and the assistance in providing us with the data.

This work was supported by grants International Science Foundation (ISF) RLD000 and RLD300, ESO C&EE Programme A-03-049 and A-05-013, Swiss National Science Foundation 20 29871.90, and INTAS (INTAS-94-4625). Our special thanks to ISF for supporting our participation at this conference.

References

Smolkov, G. Ya., Pistolkors, A. A., Treskov, T. A., Krissinel, B. B., & Putilov, V. A. 1986, Ap&SS, 119, 1

Konovalov, S. K., Altyntsev, A. T., Grechnev, V. V., Lisysian, E. G., & Magun, A. 1997, this volume, 100