DIAPL
Difference Image Analysis Package

P. R. Wozniak[1] & W. Pych[2]
1) wozniak@lanl.gov
2) pych@camk.edu.pl
2006.10.09

Table of Contents

Introduction....2
Installation....2
Typical Run....2
Description of the scripts....4
Description of the programs....5
Parameter files....8
List of distributed files....16

Introduction

A Method for Optimal Image Subtraction was described by ***Christophe Alard and Robert H. Lupton in 1998, Astrophysical Journal v.503, p.325*** and by ***Christophe Alard in 2000, Astronomy and Astrophysics Supplement, v.144, p.363***. The **DIA** package is an implementation of this method written by Przemek Woźniak and described in ***P.R. Woźniak, 2000, Acta Astronomica, v. 50, p. 421***. The **DIAPL** package is a new, more versatile version of **DIA** modified by Wojtek Pych. In this implementation the input images can be automatically divided into smaller sub-images. This helps in a proper treatment of the rapidly variable PSF (Point Spread Function) in the processed images. This software was originally written, compiled and tested running Red Hat and Fedora Core Linux distributions.

Installation

The package may be downloaded via http://ftp.camk.edu.pl/camk/pych/DIAPL
`diapl_src.tgz` file is a gzip-compressed tar file containing the sources. It may be unpacked using a command:

```
tar -xvzf diapl_src.tgz
```

Several subdirectories containing sources of the programs and default parameter files are created.

Compilation

The software has been prepared to be compiled using GNU C and C++ compiler on a PC running Linux. The compilation procedure is started using a command:

```
install.csh
```

If everything is correct there should be no error nor warning messages during this process. All the resulting binary executable files are moved into `DIAPL_BIN` subdirectory.

Typical Run

Prepare the working directory

Working directory is a place where everything happens. In the beginning user has to copy into this directory the following files:

- the script files from `SCRIPTS` subdirectory of the distribution,
- the parameter files from `PARS` subdirectory of the distribution.

After the whole processing is finished the results are stored in the working directory.

Edit instrument parameters file

An example file: `instrument.par` is located in the `PARS` subdirectory of this distribution.

Keyword	*Description*
GAIN	Camera gain [electrons/ADU]
MIN_LEVEL	Minimum valid counts
SAT_LEVEL	Maximum valid counts
NX	Length of the X-axis side of the sub-images in pixels.
NY	Length of the Y-axis side of the sub-images in pixels.
MARG	Size of the overlapping strip between the sub-images.

Edit `diapl_setup.csh`

The user has to supply the following information:

Keyword	*Description*
DATA_DIR	Path to the directory where the input FITS files containing bias-subtracted and flat-fielded images are stored.
WORK_DIR	Path to the working directory, where the scripts, parameter files, lists of files should be copied/created and where the resulting files are to be stored.
BIN	Path to the directory where the binary executable files from the DIAPL package are stored.
FITS	Filename extension of the input/output file names in the FITS format. (Default: `fits`)
INSTRUMENT	Name of the file containing parameters for a particular instrument.
IMAGES	The name of the input text file containing the list of the input FITS files with images to be processed. (Default: `images.txt`)
TPLIMAGES	The name of the input text file containing the list of the input images to be averaged in order to create a template image for the subtraction. (Default: `tplimages.txt`)
SHIFTS	The name of the text file containing the information on shifts between the images. This file is to be created by the `shifts.csh` script.(Default: `shifts.txt`)
REFIM	The name of reference image for the geometric transformation of the analyzed images.
EDGE	Stars lying closer to the edge of a frame than this number of pixels are not taken into account for PSF modeling and photometry.

Prepare a list of images to be processed

The name of the list file is set in `diapl_setup.csh`. Since the path is set by the DATA_DIR parameter in `diapl_setup.csh` the list has to contain the file names only without the path. The names of the images should be inserted one name in a single line.

Prepare a list of template images

The list has to contain the names of the best images: small FWHM, low background level, low air-mass etc. Since the path is set by the DATA_DIR parameter in `diapl_setup.csh` the list has to contain the file names only without the path. This list may be prepared from the list of all images using the script `mktpllist.csh`
It is strongly suggested to check the quality of the selected images (presence of traces of satellites, large relative shifts etc.).

Sequence of the executed scripts

1. **mktpllist.csh**
   Relevant parameter files to be edited: `fwhm.par`shifts.csh
2. **shifts.csh**
   Relevant parameter files to be edited: `cross.par`
3. **template.csh**
   Relevant parameter files to be edited: `sfind.par, xymatch.par, xygrid.par, resample2.par, mstack.par, getpsf.par`
4. **pipe.csh**
   Relevant parameter files to be edited: `sfind.par, xymatch.par, xygrid.par, resample2.par,`

`aga.par, phot.par`

Description of the scripts

mktpllist.csh

This script prepares a list of FITS files containing the images with low background level and smallest FWHM among the images from the list. User can edit this script to change the following parameters:

FWHM_FILE	name of the output file containing the information on the FWHM and background level values for the images in the list
BKG_DENOM	(integer) the number of low background images will the total number of images divided by this parameter
FWHM_DENOM	(integer) the number of smallest FWHM images will the total number of images divided by this parameter

NOTE: FWHM_DENOM should be larger than BKG_DENOM

This script calls the program: `fwhm`

NOTE: The script works properly only when the VERBOSE parameter in the `fwhm.par` is set to 0.

shifts.csh

This script is used to estimate linear shifts between the processed frames and the reference image. The results are written into the file with the name set in `diapl_setup.csh`.

This script calls the program: `cross`

template.csh

This script is used to create a template image. The template image is then scaled and subtracted from all processed images in the `pipe.csh` script.

This script calls the following programs: `cutfitsim, sfind, xymatch, xygrid, resample2, mstack, getpsf, template`

pipe.csh

This script is used to subtract the template image from all images in the input list. Then the variable stars are searched for and finally the photometry of the stars found to be variable is performed. User can edit this script to change the parameter FIELD (default= default) which is the prefix of the names for the output files.

This script calls the following programs: `cutfitsim, sfind, im2float, xymatch, xygrid, resample2, aga, getvar, phot`

Description of the programs

(In alphabetic order)

aga

This program performs subtraction of a template image from the image being processed for the photometry purposes.

usage: **aga** parameters instrument setup mask template list_of_images

parameters:	name of the input text file containing parameters (default `aga.par`)
instrument:	name of the text file containing instrument related parameters (default `instrument.par`)
setup:	name of the DIAPL setup file (default `diapl_setup.csh`)
mask:	name of the FITS file containing bad pixels mask image (current version of script `pipe.csh` automatically generates a file `blank_ushort.fits` with no pixels marked)
template:	name of the FITS file containing the template image
list_of_images:	name of the text file containing the list of FITS file names with images to be processed

cross

This program is used to estimate preliminary shifts between a pair of input images or a reference frame and frames from the input list. The process is performed in two phases. In the first phase the images are binned (default 16 pixel bins) and the rough shift is found. The images are corrected for this shift. In the second phase a sub-frame of the central part of the images is extracted (default 512x512 pixels sub-frame). The shifts between these sub-frames are estimated with 1 pixel accuracy.

usage: **cross** parameters instrument output reference {image or -f list_of_images}

parameters:	name of the input text file containing parameters (default `cross.par`)
instrument:	name of the text file containing instrument related parameters (default `instrument.par`)
output:	name of the output text file with the relative shift(s)
reference:	name of the input FITS file with the reference image
image:	name of the input FITS file with the image to be cross-correlated
list_of_images:	name of the text file with the list of FITS file names of the images to be cross-correlated

cutfitsim

This program cuts a sub-frame of an input image. The resulting image is created even if the coordinates are outside the input image. In such case the additional pixels are filled with zero values.

usage: **cutfitsim** input_image output_image nx1 nx2 ny1 ny2

input_image:	name of the FITS file with the input image
output_image:	name of the output FITS file with the desired section of the input image
nx1:	starting X coordinate of the section to be cut (original image has 1)
nx2:	ending X coordinate of the section to be cut (original image has NAXIS1)
ny1:	starting Y coordinate of the section to be cut (original image has 1)
ny2:	ending Y coordinate of the section to be cut (original image has NAXIS2)

fitshedit

This program enables the user to view or modify the contents of the headers in the FITS files.

usage: **fitshedit** fits_file [keyword [value]]

fwhm

This program calculates average FWHM and background level for the input image. Parameters are read from the file: `fwhm.par`

The minimum output consists of:

the input file name, mean FWHM, background level, median FWHM, number of star taken for the statistics.

usage: **fwhm** input_image

input_image: name of the input FITS file with the image to analyzed

getpsf

This program finds stars in the input image and then calculates a model of the PSF. This model is a sum of Gaussian functions multiplied by polynomials depending on the (X, Y) coordinates in the image.

usage: **getpsf** parameters input_image output_file

parameters: name of the input text file containing parameters (default `getpsf.par`)

input_image: name of the FITS file with the input image

output_file: name of the output binary file containing parameters of the PSF model

getvar

This program finds variable object candidates in a series of subtracted images.

usage: **getvar** parameters instrument template psf images field x0_off y0_off

parameters: name of the input text file containing parameters (default `getvar.par`)

instrument: name of the text file containing instrument related parameters (default `instrument.par`)

template: name of the input FITS file with the template image

psf: name of the input binary file containing the parameters of the PSF model

images: list of the names of FITS files with subtracted images to be tested

field: name of the observed field; output files are:

{field}`.coo`: ASCII table of coordinates of variable object candidates

{field}`.cat`: binary catalog of variable object candidates

x0_off: X axis offset of the sub-frame in respect to the whole image

y0_off: Y axis offset of the sub-frame in respect to the whole image

im2float

This program converts image from a FITS file to FITS file in FLOAT format.

usage: **im2float** input_image output_image

input_image: name of the input FITS file conforming any format of the FITS standard (BITPIX=8, 16, 32, 32, -64)

output_image: name of the output FITS file with the image in FLOAT format (BITPIX = -32)

mstack

This program averages several images to create a template image.

usage: **mstack** parameters instrument setup list_of_images output_image

parameters: name of the input text file containing parameters (default mst`ack.par`)

instrument: name of the text file containing instrument related parameters (default `instrument.par`)

setup: name of the input DIAPL setup file (default `diapl_setup.csh`)

list_of_images: name of the input text file containing the list of the names of FITS files to be stacked

output_image: name of the output FITS file with the averaged image

phot

This program calculates the photometry (AC signal) for each object from the input coordinate list.

usage: **phot** parameters instrument template psf list_of_images field

parameters: name of the input text file containing parameters (default `phot.par`)
instrument: name of the text file containing instrument related parameters (default `instrument.par`)
template: name of the input FITS file with the template image
psf: name of the input binary file with the parameters of the PSF model
list_of_images: name of the input text file containing the list of the names of FITS files to be processed
field: name of the observed field;
input file:
{field}`.coo`: ASCII table of coordinates of variable object candidates
output file:
{field}`.db`: ASCII/binary database of the photometry of variable objects

resample2

This program geometrically transforms an input image to the pixels grid of the reference image. The bi-cubic spline method is used.

usage: **resample2** parameters instrument coefficients input_image output_image

parameters: name of the input text file containing parameters (default `resample2.par`)
instrument: name of the text file containing instrument related parameters (default `instrument.par`)
coefficients: name of the input binary file with the transformation coefficients
input_image: name of the input FITS file with the image to be transformed
output_image: name of the output FITS file with the transformed image

sfind

This program performs a search for stars in the input image. Coordinates and a rough estimate of the magnitude are written into the output file.

usage: **sfind** parameters instrument input_image output_file

parameters: name of the input text file containing parameters (default `sfind.par`)
instrument: name of the text file containing instrument related parameters (default `instrument.par`)
input_image: name of the FITS file with the input image
output_file: name of the output text file with the coordinates and rough estimates of brightness of the stars found

template

This program glues several sub-images into one big image.

usage: **template** list_of_images output_image x_nim y_nim

list_of_images: name of the input text file containing a list of FITS files with the image sections to be glued
output_image: name of the output FITS file with the whole image
x_nim: number of section cuts along the X axis
y_nim: number of section cuts along the Y axis

xymatch

This program cross-identifies two lists of stellar positions using the triangles method. A list of corresponding coordinates in created as an output. Relative shift between the images is written to the standard output.

usage: **xymatch** parameters reference_list matched_list output_file

parameters: name of the input text file containing parameters (default `xymatch.par`)
reference_list: name of the input text file with the list of coordinates of stars in the reference image
matched_list: name of the input text file with the list of coordinates of stars in the analyzed image

output_file: name of the output text file with the list of matched coordinates in the format: X1, Y1, X2, Y2

xygrid

This program calculates coefficients for the geometric transformation of an image to the pixels grid of the reference image.

usage: **xygrid** parameters input_list coeff_file

parameters: name of the input text file containing parameters (default `xygrid.par`)

input_list: name of the input text file with the list of coordinates (as returned by program **xymatch**)

coeff_file: name of the output binary file containing the transformation coefficients

Parameter files

(In alphabetic order)

`aga.par`

Keyword	Default value	Description
KER_HW	7	convolution kernel half width [pixels]
N_COMP	3	number of components of kernel convolution (Gaussian functions modified by polynomials)
DEG_1	4	first polynomial degree
SIG_GAUSS_1	0,78	sigma for first Gaussian
DEG_INC	-1	degree increment between subsequent polynomials
SIG_GAUSS_INC	1,73	sigma increment of subsequent Gaussian functions
BKG_DEG	1	degree of the polynomial that describes background dependence on (X,Y) (when sky has gradients set larger values)
BAD_VALUE	65000	value assigned to bad pixels
BAD_GROWRAD1	2	radius around bad pixel where BAD_VALUE is assigned (for the analyzed frame)
BAD_GROWRAD2	2	the same as above but for the subtracted frame
MIN_AREA	0,4	minimum area with good pixels required to perform subtraction
MAX_NITER	1	maximum number of iteration for cleaning pixel distribution inside a domain
N_SIG	3	sigma multiplicity for pixels distribution cleaning inside domain
MAX_CHI2	8	maximum χ^2 allowed for the correct subtraction
WDEG_SPATIAL	2	degree of the space dependence polynomial
DOM_HW	15	half-width for the star in the domain
NDOM_X	20	density of domain grid in X
NDOM_Y	20	density of domain grid in Y
DOM_THRESH	500	threshold above background for the stars used as domain centers (central pixel of

Keyword	Default value	Description
		the star)
N_SIG_DOM	2	sigma value for domains distribution cleaning
DOMAIN_MODE	1	mode of domain search: 0=uniformly distributed 1=centered on bright stars
MOHW	4	half-width for searching of domains around bright stars
N_ITER_DOM	2	number of iterations to find domain distribution
MIN_AREA_DOM	0,75	percentage of the domain to accept it as good
MIN_NKEEP	50	minimum number of domains left to calculate the final solution
VERBOSE	1	verbosity level (0=quiet, 1, 2, 3, ...)

`cross.par`

Keyword	Default value	Description
X_NBIN	16	binning rate in the X direction for the preliminary shift estimate
Y_NBIN	16	binning rate in the Y direction for the preliminary shift estimate
NX_CUT	512	X size of the central sub-frame for the final shift estimate
NY_CUT	512	Y size of the central sub-frame for the final shift estimate
BKG_FRAC	0,5	moment for the background level estimate (0.5: median)
VERBOSE	1	verbose level (0=quiet, 1, 2, 3)

`fwhm.par`

Keyword	Default value	Description
SKY_MOD	mode	algorithm for the calculation of the background level: mode, median or mean
DATA_SIGN	signed	in some images with BITPIX=16 or 32 not conforming the FITS standard unsigned may happen
APERTURE	8	initial aperture size [pixels]
IN_SKY_RAD	2	inner background annulus radius; this number multiplied by the MAX_FWHM is used for calculations
OUT_SKY_RAD	5	outer background annulus radius; this number multiplied by the MAX_FWHM is used for calculations

MARGIN	5	stars closer to edge of the image will not be taken into account
MIN_FWHM	2,5	minimum valid FWHM of a star (to avoid cosmic rays)
MAX_FWHM	10	maximum valid FWHM of a star (to avoid galaxies)
MIN_PEAK	5000	minimum peak counts of the stars taken for statistics
MAX_PEAK	20000	maximum peak counts of the stars taken for statistics
MIN_SKY	0	minimum valid background level
MAX_SKY	5000	maximum valid background level
VERBOSE	0	verbosity level (0: for `mktpllist.csh`)

`getpsf.par`

Keyword	Default value	Description
NBOX_X	5	X-density of the BOXes
NBOX_Y	5	Y-density of the BOXes
NDEG_SPAT	2	degree of the polynomial describing dependence of the PSF shape on the position on a frame (X, Y)
NDEG_LOCAL	3	degree of the polynomial describing the PSF shape (U, V)
NGAUSS	2	number of Gaussian functions used in the PSF model
NPSF_MAX	100	maximum number of stars taken for the PSF modeling
MIN_NBOX	2	minimum number of stars in a BOX
MIN_FLUX	3000	minimum flux of a star
MAX_THRESH	500	maximum counts of the brightest pixel is brighter than the PSF model
RAT_THRESH	0,2	maximum ratio of the brightest pixel to the median of surrounding pixel (cosmic ray shield)
NSIG_DETECT	2	sigma threshold for the stars detection
CONTRAST	2	coefficient for the maxima selection
NSIG_RAT	3	
PSFHW	10	raster size for the PSF model
MAXHW	1	maximum radius of the circle around PSF peak within which the brightest pixel must be found
PEAKHW	5	radius of the circle within which the brightest pixel must be a local maximum
ISOHW	3	size of the region within which a star must be isolated
ISO_OFF	0	flux offset for the isolation test
ISO_SLO	0,15	slope of the offset dependence for the isolation test:

Keyword	Default value	Description
		neighbor_flux[%of_star_flux]<distance[pixels]*ISO_SLO+ISO_OFF (If ISO_OFF=0.0 and ISO_SLO=0.0 then star must not have any neighbors)
FITRAD	5	radius of the circle around centroid within which all pixels are taken to calculate the PSF model
APRAD	10	aperture radius
ANRAD1	20	inner radius of the background annulus
ANRAD2	25	outer radius of the background annulus
BKG_FRAC	0,3	moment for the background estimate within the background annulus
NSIG_CLIP	2	PSF sigma clipping factor
NITER_INIT	3	number of iteration steps with constant PSF
NITER	4	number of iteration steps with variable PSF
RECENTER	1	recenter the stars: 0=off 1=on (0.1 [pixels] step)
PSF_COS	1	starting value of the cosine coefficient for the PSF rotation angle
PSF_SIN	0	starting value of the sine coefficient for the PSF rotation angle
PSF_AX	-0,8	PSF X-width scale
PSF_AY	-0,8	PSF Y-width scale
SIGMA_INC	0,55	ratio of the widths of the subsequent Gaussian functions in the PSF model
SIGMA_MSCALE	1,58	coefficient for sigma scaling for the distribution moments estimation
VERBOSE	1	verbose level (0=quiet, 1, 2, ...)

`getvar.par`

Keyword	Default value	Description
APRAD	10	aperture radius
FITRAD	10	profile photometry fitting radius
NORMRAD	15	PSF normalization radius
ANRAD1	20	inner annulus radius for background estimation
ANRAD2	25	outer annulus radius for background estimation
BKG_MODE	1	mode for the background calculation: 0=the background set to 0.0 1=the background is calculated on the subtracted image in the annulus defined by

Keyword	Default value	Description
		ANRAD parameters
NSIG_BKG	2	sigma clipping factor for background estimate
BAD_VALUE	-99,9	value assigned when program is unable to compute photometry
ERR_CODE	65000	value assigned to bad pixels
SMHW	1	radius of the circle within which the local maximum is search for
MOHW	2	model of the PSF used for variability check
C_MIN	0,7	minimum correlation factor between the star's profile and the PSF model
NOBJ_INIT	64	initial length of variable table (obsolete but still in use)
ID_RAD	2	identification radius for stars classified as *transient* and *sinusoidal*
NCONS_VAR1	3	minimum number of subsequent counts standing out to classify the pixel as variable
NPTS_VAR2	10	minimum number of subsequent counts standing out in one direction to classify the pixel as *transient*
NSIG_VAR1	3	sigma threshold for variable selection
NSIG_VAR2	4	sigma threshold for *transient* variable selection
LIM_RATIO	2	ratio of the number of pixels classified as *transient* and *sinusoidal* to classify a star
ISOHW	4	distance in pixels for searching of isolated stars
ISO_OFF	0	flux offset for the isolation test
ISO_SLO	0,15	slope of the offset dependence for the isolation test: neighbor_flux[%of_star_flux]<distance[pixels]*ISO_SLO+ISO_OFF (If ISO_OFF=0.0 and ISO_SLO=0.0 then star must not have any neighbors.)
BAD_MARGIN	14	margin used to remove multiple detections of the same star
CENTER_HW	10	half width of the raster centered on a star
FILTER_HW	1	filter half width
FWHM_FRAC	0,9	the percentage of the best seeing frames used for the variability search
VERBOSE	1	verbosity level (0=quiet, 1, 2, ...)

`mstack.par`

Keyword	Default value	Description
NY	1124	size of subsection in Y (???)
HIST_NBIN	2000	number of histogram bins (used for the background estimate)

Keyword	Default value	Description
HIST_LOW	0	minimum value in histogram
HIST_HIGH	2000	maximum value in histogram
NBIN_SMOOTH	6	histogram smoothing factor
BAD_VALUE	65000	value assigned to bad pixels
BKG_FRAC	0,3	moment for the background level estimate (0.5: median)
THRESHOLD	300	threshold for stars detection in counts
MIN_SCALE	0,5	minimum scaling factor between a frame and the reference frame
MAX_SCALE	2	maximum scaling factor between a frame and the reference frame
MIN_NGOOD	5	minimum number of frames with good counts to perform stacking at a given pixel
N_REJ_LOW	1	number of data points with lowest counts to reject
N_REJ_HIGH	1	number of data points with highest counts to reject
MEDIAN	0	0=use mean value for each pixel 1=use median for each pixel (WARNING! median well good only for frames with the same PSF)
VERBOSE	1	verbosity level (0=quiet, 1, 2, ...)

phot.par

Keyword	Default value	Description
APRAD	10	aperture radius
FITRAD	10	profile photometry radius
NORMRAD	15	PSF normalization factor
ANRAD1	20	inner annulus radius for background estimation
ANRAD2	25	outer annulus radius for background estimation
BKG_MODE	0	background estimate mode, when value is not 0 ANRAD values are used
NSIG_BKG	2	sigma clipping factor for pixels distribution
BAD_VALUE	-99,9	value assigned when program is unable to compute photometry
ERR_CODE	65000	value assigned to bad pixels
DBFORM	A	Photometric database format:A= ASCII table; B= binary file
VERBOSE	1	verbosity level (0=quiet, 1, 2, ...)

resample2.par

Keyword	Default value	Description
FIX_COSMICS	0	**0=strongly recommended** 1=remove cosmic rays: Rejects pixels with counts N_SIG_COSMICS above the median of neighbor pixels and the gradient is higher than MAG_GRAD **This procedure does not not work properly.**
MAX_GRAD	2	gradient limit for FIX_COSMICS
N_SIG_COSMICS	15	sigma limit FIX_COSMICS
FIX_RINGS	0	1=remove bi-cubic spline by linear interpolation **This procedure does not not work properly.**
N_SIG_RINGS	2	sigma threshold for FIX_RINGS
N_SIG_RM	2,5	sigma threshold for bad column removal (IS & KŻ)
GROW_RAD	0	growing radius for bad column removal
VERBOSE	1	verbosity level (0=quiet, 1, 2, ...)

`sfind.par`

Keyword	Default value	Description
MOHW	5	approximate FWHM used for the correlation
SMHW	4	maximum distance of the brightest pixel from the star's center
N_SIG	2	sigma clipping factor for pixels inside an annulus/circle (?)
SIG_X	1,5	starting sigma in X for the PSF model
SIG_Y	1,5	starting sigma in Y for the PSF model
C_MIN	0,7	minimum correlation to be accept as a star
APRAD	10	photometry aperture radius
ANRAD1	20	inner radius of the background annulus
ANRAD2	25	outer radius of the background annulus
ABS_THRESH	-1	threshold above the background for star detection If (ABS_THRESH<0) then SIG_THRESH is used
SIG_THRESH	1,5	sigma threshold above the background for star detection used only when (ABS_THRESH<0)
VERBOSE	1	verbosity level (0=quiet, 1, 2, ...)

xygrid.par

Keyword	Default value	Description
NDEG	2	degree of the polynomial for the astrometric transformation
SIGMA_F	2	sigma clipping factor
MAX_NITER	3	maximum number of iteration steps
VERBOSE	1	verbosity level (0=quiet, 1, 2, ...)

xymatch.par

Keyword	Default value	Description
NSUB	20	number of brightest stars taken for the preliminary cross-identification
LLIM	5	minimum length of the longest side of a triangle [pixels]
RLIM	5	maximum ratio of the sides length in a triangle
FVNO	0,2	ratio of the number of cross-identified triangles to total number of the triangles
LTOL	1	maximum difference of the lengths of the longest sides of the triangles being compared [pixels]
RTOL	0,02	maximum error of the side lengths ratio of the triangles being compared
CTOL	0,02	maximum error of the cosines of the angles in the triangles being compared
PTOL	0,5	final tolerance of the (X,Y) coordinates of the same star [pixels]
VERBOSE	1	verbosity level (0=quiet, 1, 2, ...)

List of distributed files

The package is distributed in a single gzip-ed tar file: `diapl_src.tgz`

It contains the following files:

aga/
aga/init_difimages.c
aga/apply_norm.c
aga/aga.c
aga/local_clip.c
aga/max.c
aga/xy_convolve.c
aga/make_domains.c
aga/get_params.c
aga/write_sector.c
aga/clip_domains.c
aga/base_func.c
aga/mask_badpix.c
aga/get_domains.c
aga/spatial_convolve.c
aga/clean_domains.c
aga/makefile
aga/make_vectors.c
aga/defs.h
aga/funcs.h
aga/lubksb.c
aga/write_kernel.c
aga/ludcmp.c
aga/local_solution.c
aga/clone_domains.c
aga/quick_sort.c
aga/expand_matrix.c
aga/pfitsio1.c
aga/pfitsio1.h
aga/read_sector.c
aga/errmess.c
aga/errmess.h
aga/get_headlen.c
cross/
cross/cross.c
cross/crf.c
cross/get_params.c
cross/get_max.c
cross/makefile
cross/get_sector.c
cross/defs.h
cross/fourn.c
cross/funcs.h
cross/pfitsin1.c
cross/pfitsin1.h
cross/pack_data.c
cross/errmess.c
cross/errmess.h
cutfitsim/
cutfitsim/makefile
cutfitsim/swap.c
cutfitsim/swap.h
cutfitsim/cutfitsim.c
cutfitsim/errmess.c
cutfitsim/errmess.h
cutfitsim/pfitshead.c
cutfitsim/pfitshead.h
diapl.pdf
diapl.sxw
fitshedit/
fitshedit/makefile
fitshedit/fitshedit.c
fitshedit/errmess.c
fitshedit/errmess.h
fitshedit/pfitshead.c
fitshedit/pfitshead.h
fwhm/
fwhm/makefile
fwhm/fwhm.c
fwhm/pfitsin.c
fwhm/pfitsin.h
fwhm/errmess.c
fwhm/errmess.h
getpsf/
getpsf/write_psf.c
getpsf/center.c
getpsf/is_peak.c
getpsf/get_gaussian.c
getpsf/bkg.c
getpsf/refine_max.c
getpsf/psf_core.c
getpsf/get_params.c
getpsf/init_psf.c
getpsf/clip_stars.c
getpsf/copy_obj.c
getpsf/makefile
getpsf/local_fit.c
getpsf/local_max.c
getpsf/defs.h
getpsf/silly_phot.c
getpsf/fitpsf.c
getpsf/getpsf.c
getpsf/funcs.h
getpsf/lubksb.c
getpsf/ludcmp.c
getpsf/quick_sort.c
getpsf/is_isolated.c
getpsf/psf_bitmaps.c
getpsf/pfitsin1.c
getpsf/pfitsin1.h
getpsf/errmess.c
getpsf/errmess.h
getpsf/psf_stars.c
getvar/
getvar/aperphot.c
getvar/bkg.c
getvar/read_kernel.c
getvar/center_stars.c
getvar/make_kernel.c
getvar/psf_core.c
getvar/get_params.c
getvar/init_psf.c
getvar/base_func.c
getvar/makefile
getvar/get_fwhm.c
getvar/read_psf.c
getvar/make_vectors.c
getvar/defs.h
getvar/get_phot.c
getvar/getvar.c
getvar/covar_sig.c
getvar/funcs.h
getvar/get_repeaters.c
getvar/neighbor.c
getvar/quick_sort.c
getvar/find_stars.c
getvar/im_convolve.c
getvar/indexx.c
getvar/pfitsio1.c
getvar/pfitsio1.h
getvar/cross_id.c
getvar/make_psf.c
getvar/centroid.c
getvar/spatial_coeffs.c
getvar/errmess.c
getvar/errmess.h
im2float/
im2float/makefile
im2float/im2float.c
im2float/pfitsio.c
im2float/pfitsio.h
im2float/errmess.c
im2float/errmess.h
install.csh
LIB/
LIB/base_func.c
LIB/lubksb.c
LIB/lubksb.h
LIB/ludcmp.c
LIB/ludcmp.h
LIB/swap.c
LIB/swap.h
LIB/quick_sort.c
LIB/indexx.c
LIB/indexx.h
LIB/pfitsin.c
LIB/pfitsin.h
LIB/pfitsio.c
LIB/pfitsio.h
LIB/pfitsin1.c
LIB/pfitsin1.h
LIB/pfitsio1.c
LIB/pfitsio1.h
LIB/errmess.c
LIB/errmess.h
LIB/pfitshead.c
LIB/pfitshead.h
mkushortone/
mkushortone/makefile
mkushortone/mkushortone.c
mkushortone/pfitsio.c
mkushortone/pfitsio.h
mkushortone/errmess.c
mkushortone/errmess.h
mstack/
mstack/mstack.c
mstack/histogram.c
mstack/bin.c
mstack/get_params.c
mstack/write_sector.c
mstack/makefile
mstack/defs.h
mstack/get_peak.c
mstack/funcs.h
mstack/pfitsio1.c
mstack/pfitsio1.h
mstack/read_sector.c
mstack/median.c
mstack/errmess.c
mstack/errmess.h
PARS/
PARS/getpsf.par
PARS/xygrid.par
PARS/mstack.par
PARS/instrument.par
PARS/phot.par
PARS/fwhm.par
PARS/sfind.par
PARS/cross.par
PARS/resample2.par
PARS/xymatch.par
PARS/site3.par
PARS/aga.par
PARS/getvar.par
phot/
phot/bkg.c
phot/read_kernel.c
phot/make_kernel.c
phot/psf_core.c
phot/get_params.c
phot/init_psf.c
phot/base_func.c
phot/makefile
phot/get_fwhm.c
phot/read_psf.c
phot/make_vectors.c
phot/defs.h
phot/get_phot.c
phot/funcs.h
phot/phot.c
phot/im_convolve.c
phot/indexx.c
phot/pfitsin1.c
phot/pfitsin1.h

phot/make_psf.c
phot/spatial_coeffs.c
phot/errmess.c
phot/errmess.h
README
resample2/
resample2/spline.c
resample2/spline.h
resample2/splint.c
resample2/splint.h
resample2/makefile
resample2/rm_bad_col.c
resample2/rm_bad_col.h
resample2/res_bad_col.c
resample2/res_bad_col.h
resample2/defs.h
resample2/hedit.c
resample2/hedit.h
resample2/poly.c
resample2/poly.h
resample2/ran1.c
resample2/ran1.h
resample2/resample2.c
resample2/indexx.c
resample2/indexx.h
resample2/pfitsio1.c
resample2/pfitsio1.h
resample2/bicspl.c
resample2/bicspl.h
resample2/gasdev.c
resample2/gasdev.h
resample2/errmess.c
resample2/errmess.h
resample2/fix_cosmics.c
resample2/fix_cosmics.h
SCRIPTS/
SCRIPTS/mktpllist.csh
SCRIPTS/pipe.csh
SCRIPTS/shifts.csh
SCRIPTS/template.csh
SCRIPTS/diapl_setup.csh
sfind/
sfind/covar.c
sfind/covar.h
sfind/aperphot.c
sfind/aperphot.h
sfind/bkg.c
sfind/bkg.h
sfind/sfind.c
sfind/makefile
sfind/defs.h
sfind/indexx.c
sfind/indexx.h
sfind/pfitsin1.c
sfind/pfitsin1.h
sfind/centroid.c
sfind/centroid.h
sfind/errmess.c
sfind/errmess.h
template/
template/makefile
template/hedit.c
template/hedit.h
template/pfitsio1.c
template/pfitsio1.h
template/errmess.c
template/errmess.h
template/template.c
xygrid/
xygrid/sigma.c
xygrid/sigma.h
xygrid/makefile
xygrid/defs.h
xygrid/fitn.c
xygrid/fitn.h
xygrid/xygrid.c
xygrid/lubksb.c
xygrid/lubksb.h
xygrid/ludcmp.c
xygrid/ludcmp.h
xygrid/poly.c
xygrid/poly.h
xygrid/errmess.c
xygrid/errmess.h
xymatch/
xymatch/tri_gen.c
xymatch/tri_gen.h
xymatch/xymatch.c
xymatch/makefile
xymatch/xy_lin.c
xymatch/xy_lin.h
xymatch/defs.h
xymatch/bright_end.c
xymatch/bright_end.h
xymatch/triangles.c
xymatch/triangles.h
xymatch/indexx.c
xymatch/indexx.h
xymatch/errmess.c
xymatch/errmess.h
xymatch/refine.c
xymatch/refine.h