```
####################################
###    MCSRED2 Quick instruction  ###
###        Ichi Tanaka, NAOJ      ###
####################################
Last Update: 2017-05-17
```

<<< Introduction >>>
The MCSRED package is the software for MOIRCS imaging data reduction. The MCSRED 2 is speficically for the data afte the detector change in late 2015. The current version is still preliminary and will be revised frequently in the future. Please keep checking the mcsred website.

http://www.naoj.org/staff/ichi/MCSRED/mcsred_e.html

This document is not intended to give the full description of the package in English, but aimed for the quick introduction on how to use it.

The distortion/mosaicking pattern will change every time we warm-up the MOIRCS. So BE VERY CAREFUL IF THE VERSION OF THE DISTORTION CORRECTION DATABASE MATCHES TO YOUR OBSERVATION DATE.

Usually the SS will give some instructions to the observers who wants to use the mcsred package during the observation.

If you only need the distortion correction package, you can use the task mcsgeocorr.cl in the package.

<<< 1. Preparation >>>
This package is developed under the IRAF version 2.16.1.
It also requires the SExtractor package.

(1-1) Install and edit login.cl
Firstly, extract the tarball of the package at the arbitary directory.

Edit login.cl as follows. Add the lines below.

```
        task $mcsred = (full path to the extracted directory)/MCSRED2/mcsred.cl
        set dir_mcsred  = "(full path to the extracted directory)/MCSRED2/"
```

Also, set these values:

```
        set     imtype          = "fits"
        set     stdimage        = imt2048
```

Some tasks suppose the use of the default parameter. So unlearn iraf.

```
        cl> unlearn iraf
```

(1-2) Define the environmental variable "MCSRED_DIR"
Example -- for bash case:
In your .bash_profile or .bashrc
export MCSRED_DIR="(full path to the extracted directory)/MCSRED2"

For tcsh case (.cshrc)
setenv MCSRED_DIR (full path to the extracted directory)/MCSRED2

(1-3) SExtractor
Install after downloading from the link below.
http://www.astromatic.net/software/sextractor

Note the executable for SExtractor before version 2.8 was "sex". But the most recent version may be "sextractor". It affects the task gsextcat.cl in mcsred2. Edit the relevant part of the task if the executable in your environment is "sextractor".

(1-4) Starting the task
Reload .bashrc etc, then start IRAF. In cl, type below.

```
ecl> mcsred
```

Then mcsred packeage will be loaded. Also start the image display tool such as ds9.

If you update MCSRED, you MUST reflesh the mcsred parameters as below (after loading the package).

```
mcsred> unlearn mcsred
```

(1-5) Configuration for mcsred2
Before the execution of the package, you have to choose the configuration file that matches to your data. After loading the package, load configpars as below.

```
mcsred> epar configpars

PACKAGE = mcsred
   TASK = configpars

(config = dir_mcsred$DATABASE/ana_oct16.cfg) the name of mcsred config file
(mode   =                    q)
```

The choice of config file is below.
- ana_dec15.cfg: for the data Dec 2015.
- ana_apr16.cfg: for the data Apr -- July 2016
- ana_oct16.cfg: for the data later than Aug 2016.

Note that the configuration file contains the distortion correction database. It will change the pixel scale of the image to 0.1160"/pix, while under the old detector (old mcsred package) it was set to ~0.1170"/pix.

The change of the pixel scale affects the preparation of the preimage which assumes 0.1170"/pix in wmdp_moircs.sav software. Please make sure to change the pixel scale to 0.1170"/pix when you use the data reduced by mcsred2 for preimage.

<<< 2. Pipeline Execution -- mcsall.cl >>>
The all-in-one task "mcsall.cl" will execute the following procedures for the data in one command. The task is for the reduction of the deep field data: i.e.,the input data should not contain large objects. Here "large" means larger than the size of the dither pattern.

If the target is so large that occupy more than half of the detector, then the task may not work appropriately. For the data taken by SKYNOD option in GETOBJECT command (for observation of such objects), we will have another task (ask the SS about the availability).

The mcsall task contains the following steps.
 0) [Preparation] Check input images, combine NEXPOSURE data.
 1) Making object masks (qobjmask.cl)
 2) Flat fielding. If no flat is found, it makes self flat.
 3) Image cleaning by sbself+mcsimsurfit (or allskysub.cl).
 4) Additional line-column bias-drift cleaning (optional).
 5) Distortion Correction (mcsgeocorr)
 6) Object detection (gsextcat)
 7) Catalog matching, registering images, and the final combine (gmkgtrimage)

Additionally, the task for making input list (listprep), domeflat making task (mkdome), and the mosaicking task (dmosimg) are necessary for the essential reduction.

(2-1) Preparing the input lists
Firstly, the listfile of the processing data for each channel is necessary. The task "listprep" will make the files.

Suppose that your dithered images start from MCSA00012345 to MCSA00012384.fits and the data directory is /scr/DATA/. Then type,

```
mcsred> listprep /scr/DATA/ obj 12345 12384
```

Then the list of files for each detector will be generated.

```
        obj_1.lst ... the file list for channel 1.
        obj_2.lst ... same for channel 2.
```

!! Note !!
We recommend to keep the raw data in the different directory from the working directory. This is because the pipeline changes the fits header of the input data.

(2-2) image check.
The task "mcsall" requires a rough range of the seeing size as well as the saturation level of the data for process.

For the quick fwhm check of the raw data in the "listprep" output file, you can use the task imcheck. It displays each image after subtracting adjacent image.

```
mcsred> imcheck obj_2.lst step=1
```

For the saturation, we can only need heavy saturation level (non-linearity is not the matter). If the sky level is 15000 ADU and the center of the psf is reversed in hole due to the heavy saturation at more than 35000 ADU, the number necessary here is 35000 - 150000 = 20000 ADU (or less).

(2-3) Preparing Dome Flat Data (optional)
We recommend to use dome flats for data reduction, especially for the data in K-band window. We will take the dome flat data for the filters we use during a contiguous observing period. Even if there is no dome flat data under your proposal ID, there should be the dome flat data you need for your data near your observing date. You can find them using the "calibration search" in STARS archive.

Assume that the lamp-on dome data starts from MCSA00010101.fits to MCSA000010120.fits、and the lamp-off dome starts from MCSA00010121.fits to MCSA000010140.fits. and assume that they are in /scr/RAW/.

```
  mcsred> listprep /scr/RAW dmon 10101 10120
  mcsred> listprep /scr/RAW dmoff 10121 10140
```

The above command will make dmon_1.lst, domeon_2.lst, dmoff_1.lst, and domeoff_2.lst in the current directory.

```
  mcsred> mkdome dmon_1.lst DomeFlat_ch1.fits offsub+ offlist=dmoff_1.lst
  mcsred> mkdome dmon_2.lst DomeFlat_ch2.fits offsub+ offlist=dmoff_2.lst
```

This will make the dome flat data DomeFlat_ch?.fits (?=1/2).

Note that we can use dark frames (under the same exposure time) as a lamp-off dome data except the filters in K-band window.

(2-4) Run the mcsall task
The task (or pipeline) "mcsall" treats the images of each detector independently. For mosaicking of two channels we describe later.

Firstly check the parameters of mcsall.

```
mc> lpar mcsall
       inlist = "obj_1.lst"     The list of raw images (listprep output)
         flat = "DomeFlat_ch1.fits"  step2: The name of the flat frame.
       resimg = "OBJ_1.fits"     The name of the final resulting image
        (jump = 1)              Jump to the i-th procss
         (bye = 7)              Stop after the i-th procss
      (second = no)             Is this the 2nd process?
        (disp = yes)            Display the process?
        (rail = no)             Step1: do the rail remain on chip 1?
       (rsize = 136)            Width of the vignetted area by rail on ch1
     (msksize = 3)              step1: Object expansion size (pix)
     (sthresh = 1.1)            step1: Threshold for object mask.
    (skipmask = no)             Step1: skipping mask making process?
        (dosf = no)             step2: Yes if use the self-flat frame.
    (eachflat = yes)            step2: Yes if flat is made for each images
       (adsub = no)             step3: yes for advanced sky subtraction for NB
```

```
        (nsf = 5)            step 3: the nsf parameter for sky
       (nmin = 7)            step 3: the nmin parameter for sky
      (exelc = yes)          step4: execute the line-column background subtr
      (crrej = yes)          Step5: Cosmic-ray cleaning during mcsgeocorr?
      (minfw = 2.)           Step6: Minimum FWHM for matching catalog
      (maxfw = 10.)          Step6: Maximun FWHM for matching catalog
      (satur = 20000.)       Step6: Saturation counts
      (thres = 5.)           Step6: SExtractor: detect_thresh
       (conn = 10)           Step6: SExtractor: connected pixel
       (nmat = 40.)          Step7: nmatch parameter in xyxymatch
      (fstop = yes)          Step7: Pause the process after xyxymatch and be
    (combine = "average")    Step7: type of final combine?
     (reject = "pclip")      Step7: type of rejection
     (weight = "sigma")      Step7: weight -sigma or exptime?
     (lsigma = 3.)           Step7: lower sigma clipping factor
     (hsigma = 3.)           Step7: upper sigma clipping factor
       (list = "")
       (mode = "q")
```

Parameters:
jump, bye ... You can stop (bye) or start (jump) at any steps in the proce dure described above (2-0).

dosf, flat ... If you process the data with self flat, run mcsall with 'dos f=yes' (or dosf+) as well as the name of the resulting self flat (flat). If not, you set 'dosf=no' with supplying the name of the external flat frame name (e.g. DomeFlat_ch1.fits). If there is already the flat frame exists under dosf+, the task will end in err.

second ... If you run the mcsall for the 2nd part of the "2-path proces s", you can run the command with second=yes. This will automatically set the nec essary parameters for the 2nd path. In other words, if we run the 2nd path witho ut setting the parameter to yes, the mask file name on the header is deleted and overwritten to the quick version of the mask. Be careful.

rail ... If your data contain the ~100 pix-width 'shadow' at the smal lest-x edge of channel-1 data, you should run the task with 'rail=yes'. The shad ow is caused by the MOS-protection rail which might have remained on the data du ring the exposure. And also, this parameter is useful when there is a narrow rai l (~30pix) exists. Try rail+ with the rsize=30. It will mask out ~30pixel region at the edge.

sthresh, msksize ... The parameters that defines the object mask. Usually defail t is okay. Be careful not to set too low threshold, as it may result in large nu mber of masked regions for the spurious objects.

skipmask ... Skip mask making process in Step 1. This is for quick look r eduction for the first part of the 2-path process (described later).

eachflat ... If yes (and dosf=yes), each self flat frame is made for indi vidual frame excluding the frame for flatfielding. Note that the self flat frame from the first part of the step 2 is not used though it is created for the case . This is important when the number of frame for process is not many (<20). If t he processing frame is less than 10, it is forced to "yes" to avoid the affectio n of it to the faint objects which escaped the object masking on the flatfield d ata.

skipsky (TBD) ... skipping median-sky subtraction process. If you process the (broadband) data with self-flats that are made with less than ~10 frames under n ormal condition, you may want to skip the median sky process for better S/N. Thi s switch is TBD for mcsred2.

adsub ... Execute the advanced sky subtraction? If yes, the task will f irst remove the sky variation before median-sky subtraction (it uses master medi an sky and fit residual sky using polynominal fit plane for each, subtract it in dividually to the input data before subtracting median sky). This option is main ly for the NB data with rapid sky variation. Executing the adsub process for Bro adband data or some NB data has almost no difference. This process is a bit time -consuming.

nsf ... It defines the number of frames used to make the median-sky data. The median-sky data will be generated using the 2x"nself" frames before an d after the image for subtraction (the sky data should not include the image its elf for subtraction). If 'nself=0' then all images except the image for subtrac

tion will be used. If sky change is large with fringing patterns, smaller number
 (2-3) will be better to subtract global pattern. But smaller number will result
 in poorer pixel-to-pixel noise.

nmin ... The smallest number of frames used for making median sky whe
n the number of data before or after the image for sky subtracton is less than n
self.

minfw, maxfw ... Objects within these sizes are used for image alignments. T
hese values should be determined carefully. Do not put larger minfw value than
the minimum size of the FWHM among all the input images. If the detected number
 of stars are too small, the matching procedure may fail. So the maxfw should b
e set to keep the number of input objects for image registeration around 50-150.
 Usually 2 times FWHM works.

satur, thres, conn ... The SExtractor detection parameters (SATURATION, DETECT_T
HRESH, CONNECT). For 'satur' you do not need to consider the number of coadds, a
s the task automatically treat it. But you should be aware that the satur value
is does not include the sky level; i.e., if the raw sky count is ~20000, the sat
ur parameter should be less than 20000 as the heavy saturation of the detector s
tarts at around 40000. Note that some non-linearity is not the matter, as the ca
talog here is just used for the positioning, not photometry.

combine, reject, weight ... parameters for final combine. Change appropriately t
o the data.

nmat ...the number of sample data for catalog matching (-> see the he
lp for IRAF xyxymatch: nmatch). At the same time, the number nmat is also used t
o set the number of cataloged objects for position matching in the process 6. Pl
ease avoid setting nmat > 50, as the xyxymatch process will take very long time.

fstop .... If yes, it will stop after xyxymatch before proceeding to fi
nal combine in step 7.

Also make sure that image display tool (ds9 etc) is launched if you run the task
 with 'display = yes'.

** Example **
(1) Suppose that the FWHM of the data is around 4.5-6.0 pixel. Then,

mc> mcsall obj_1.lst DomeFlat_1.fits Result_1.fits dosf- thres=3 conn=10 nmat=20

The non-specified parameter assumes the default values. The execution time for 1
0-frame input list will take about 10 min.

(2) If self flat is used...

mc> mcsall obj_1.lst SelfFlat_1.fits Result_1.fits dosf+ eachflat+ thres=3 nmat=
20

* If the SelfFlat_1.fits file already exists, the task end in error.
* If the number of input file is <20, eachflat+ is strongly recommended.

(3) If the data is NB imaging with rapid sky variation...

mc> mcsall obj_1.lst DomeFlat_NB_1.fits Result_1.fits dosf- adsub+ thres=1.5 nma
t=10

* adsub+ is for special sky subtraction for the case with rapid sky variation. F
or NB filters such as NB119, BrG, NB207 it will be effective.
* usually the NB data has fewer objects in each images. setting lower thres and
nmat may be better. Try and error is necessary (i.e., use jump and bye effective
ly).

<<< 3. Mosaicking >>>
Mosaicking of the detector will require the sensitivity ratio between channels.
The scale parameter sc is defined as

sensitivity(ch1)=sensitivity(ch2)*sc.

You will need the value when you do mosaicking of each shot using the product of the step 5 of the mcsall task (flat-fielded, sky-subtracted, and distortion-corrected files: e.g., gcclSBsblMCSA00012345.fits etc). Three steps will be required to make a combined mosaic image.

(3-1) mosaic task -- dmosimg / mosgcsbim
The task dmosimg creates the mosaic image of each exposures using the list of the flatflelded, sky-subtracted, and distortion-corrected images. Or, there is another task mosgcsbimg for input of listfile from the Step 5.

(3-1-1) dmosimg.cl
Firstly prepare the input file list.

```
mc> files gc*fits > gcall.lst
mc> type gcall.lst
gcSBsbflMCSA00032123.fits
gcSBsbflMCSA00032124.fits
gcSBsbflMCSA00032125.fits
gcSBsbflMCSA00032126.fits
....
```

Note that the input data must contain the data for channel-1 and 2 from each single exposure as a set. Make sure that the list is the order of chip1, chip2, chip1, chip2, ...

Then run lpar dmosimg. Firstly check the parameters.

```
mc> lpar dmosimg
       inlist = "gcall.lst"    input file list
        froot = "mosimg"          the root name for mosaicked individual images
          (sc = 0.82)           Sensitivity scale factor of chip 2.
  (configpars = "")            the mcsred config parameter
        (corx = 0)        additional dx for ch2 to default mosaic rule
        (cory = 0)        additional dy for ch2 to default mosaic rule
        (sky = yes)            subtract sky?
        (disp = yes)            Display the result?
        (mode = "q")
```

* Setting the correct value on the scale parameter 'sc' is the critical part of the mosaic process. You can estimate the sc value from the instrument magunitudes of the standard star data which are taken with both detectors. Or, you can also estimate it by calculating the ratio of the measured domeflat or sky counts from each detector.

For the latter, there is a task for the purpose called sccalc.
e.g.,
mcsred> sccalc domeon_1.lst domeon_2.lst sc_out.out

Or, you can use the output of the task "mkdome". When you run the task, you will see the normalization factor of the dome flat data at the end of the task. By taking the ratio (ch1_count/ch2_count) you can estimate the sc.

It is known that the sc value could change slightly (a few % level) from time to time, due to the flexure of the filter turret and/or optics.

[Notice!] You cannot use the object sky data as an input of sccalc for NB filter, because the sky in NB filter is not 'flat'!

* The parameters corx and cory are for correcting the shift of the relative position of ch2 with respect to ch1 from the default mosaic rule. The shift of a few pixels is actually observed quite often due to the flexure of the optics. We have a simple task to measuring the offset value and will be described in the Note below.

* If you set "sky=yes", the median values around each detector center ([300:1700,200:1850]" region) will be measured and subtracted during the process.

* The output will be like below for the above case.

  mosimg_000?.fits (?=1,2,3,....)
  mos_gcall.lst ... the list of output images. File name is like "mos_//(input file list)".

Each output image has the dimension of (3636, 2048) for default settig with the pixel scale of 0.116 arcsec/pix.

*** Note on Mosaicking ***

Unfortunately, the mosaic rule (position of channel-2 image relative to the channel-1) is known to move slightly by the PA and Elevation due to possible flexure of optics (at most 1-2 pixel, though). As the task domsimg just apply the pre-defined mosaic rule (dx, dy, and rotation) to all the data, there may be a possbile astrometric error on the mosaicked image. If you have a reliable reference data taken by e.g., single CCD/array with properly distortion-corrected, it is best to use the data as the reference of the MOIRCS mosaicking.

Though the wcs mapping for mosaic may also work, you should make sure that there is enough number of reference stars for wcs map. Note that for many GSC or 2MASS stars the proper motion is already significant (they were taken long time ago already!) so the use of the coordinates for stars could introduce the significant error.

If there are several stars in your dataset that were observed in both channel-1 and 2 due to the dither, you may use them for the estimate of the correction (shift only) of the default mosaic rule. You can apply the correction to the default mosaic rule by setting corx and cory parameters to the measured offsets. Below is the example.(the tasks are not yet available for mcsred2! sorry.)

At first, run the dmosimg task with corx=0 and cory=0. use the output listfile for the task findms.

```
mcsred> findms mos_gcall.lst
```

... the task finds the pair of mosaicked images with the maximum separation in X direction. The output message contains the information (img1, img2, area1, area2) for parameters used for the offset calculation task moscorcalc.

```
mcsred> moscorcalc mosimg_0001.fits mosimg_0002.fits area1=[1830:2286,*] area2=[1583:2038,*]
```

... Make sure that the hidden object detection parameters (detmin, thres, minfw, maxfw, satur) properly. Then the task will run the IRAF geomap task. Carefully inspect the result of residuals shown by IRAF (you should be familiar to geomap task!). Then dx and dy will be calculated and shown on the terminal.

Note that due to the limited accuracy of the task, you should confirm how the result is satisfactory by eye inspection, or by iteration until the offset goes below 1 pixel level.

(3-1-2) mosgcimg.cl

The task uses the output list file of the task mcsgeocor or the process 5 in mcsall (gcclSBsbfl**_?.lst). Input paraeters are as follows.

```
noddata> lpar mosgcsbim
      inlist1 = "gcclSBsbflbsobj_1.lst" distortion-corrected image list for chip 1
      inlist2 = "gcclSBsbflbsobj_2.lst" distortion-corrected image list for chip 2
         rout = "mos_obj"       rootname for output image
           sc = 1.0167          Sensitivity scale factor for chip 2
        (disp = yes)            display the mosaicked images?
  (configpars = "")             the mcsred config parameter
        (corx = 0)       additional dx for ch2 to default mosaic rule
        (cory = 0)       additional dy for ch2 to default mosaic rule
       (mode = "q")
```

The output fits images are "mos_obj0001.fits" "mos_obj0002.fits" ..., and its listfile named "mos_objall.lst" for the above case.

(3-2) Object detection

```
mc> lpar gsextcat
       infile = "@mos_gcall.lst" Input image name (allow @list format)
       detmin = 12              -DETECT_MINAREA parameter in sextractor (int).
       thresh = 3.5             -DETECT_THRESH parameter in sextractor (real).
        minfw = 2.5              Minimum size of objects for output catalog.
        maxfw = 12.             Maximum size of objects for output catalog.
        (disp = yes)            Display and check result by display?
```

```
       (satur = 15000.)        -SATUR_LEVEL parameter in sextractor.
       (recnt = no)            Execute recentering for SExtractor output?
      (nlimit = yes)           Limit the number of SExtractor catalog objects
        (cmax = 60)            Maximu number of objects in catalog
        (auto = yes)           Automatically define the detection region
         (xfr = 0.145)         detection avoiding region factor in x
         (yfr = 0.05)          detection avoiding region factor in y
         (cx0 = 311)           Beginning x coordinages for cut-out region
         (cy0 = 291)           Beginning y coordinages for cut-out region
         (cwx = 1460)          x size of the cut-out region
         (cwy = 1460)          y size of the cut-out region
        (mode = "q")
```

This is the object detection program used in mcsall (step 6). As the default th
e objects avoiding the edges of the images will be cataloged. Under 'auto=yes',
the region with x(y)fr * 100 % outside of the images are omitted from the detect
ion to reject any spurious (Or you can explicitly set the object detection range
 using cx0, cy0, cwx, cwy parameters).

 For mosaicked data case, xfr=0.05 can be used.

 If the task is run with 'nlimit = yes', the brightest cmax objects are selected
 from the original detection. The satur parameters are again excluding the sky l
evel, and the value for single shot exposure.

```
mc > gsextcat @mos_gcall.lst 10 5 3.8 10 auto+ xfr=0.05 yfr=0.05 nlimit+ cmax=30
```

The number of the catalogued objects should be less than 100 to avoid the failur
e of the object matching (by the IRAF xyxymatch) in the subsequent process (gmkg
trimages).

(3-3) Matching each mosaic images and combine -- gmkgtrimages
The object matching, registeration, and final combining can be executed using th
e task gmkgtrimages.cl.

```
c> lpar gmkgtrimages
        inlst = "mos_gcall.lst" The list of input images with associated cat
       output = "MOS_RESULT.fits" Name of result image
      (chkbox = yes)           Automatically define the final image size?
        (xbox = 300)           Dither box size in x in pixels
        (ybox = 300)           Dither box size in y in pixels
         (xc1 = 150)           Relative x coord of 1st image in dither box
         (yc1 = 150)           Relative y coord of 1st image in dither box
        (rail = no)            Do the rail remain on chip 1?
       (rsize = 136)           Width of the vignetted area by rail on ch1
     (combine = "average")     Type of the final combine operation
      (reject = "pclip")       Type of rejection
        (zero = "none")        Image sky zero level offset
      (weight = "sigma")       Weight: sigma or exptime?
      (fitgeo = "shift")       geomap type: SHOULD BE rotate/shift
      (lsigma = 3.)            Lower sigma clipping factor
      (hsigma = 3.)            Upper sigma clipping factor
       (pclip = -0.5)          Percentile clipping algorithm parameter.
        (nmat = 40.)           xyxymatch: nmatch parameter
         (tol = 10)            xyxymatch: tolerance parameter
        (nrej = 15)            xyxymatch: nreject parameter
        (sepa = 10)            xyxymatch: separation parameter
         (rat = 6)             xyxymatch: ratio parameter
       (fstop = no)            ask y/n after xyxymatch to proceed?
       (fskip = no)            skip xyxymatch and proceed to final combine?
        (skip = no)            Skip all processes except the final combine?
        (disp = yes)           display the final result?
      (gtrlst = "")            List of images for final combine
        (mode = "q")
```

The input file list (inlst) is the same as the input file for gsextcat (but with
out '@').

Parameters:
chkbox,xbox,ybox,xc1,yc1   ... If chkbox=yes, the task automatically measur
e the size of the dither box, and set the appropriate final image size consideri

ng the range of the dither. However, it does not work well if the data contains
the rotated images. If chkbox is set to no, the parameters xbox ybox xc1 yc1 wil
l define the size of the dither box and the position of 1st image within the dit
her box.

rail, rsize   ... If the MOS rail remains in ch1 edge, set rail=yes. The task
cuts the first ~100 pixels (which is defined as rsize) of the chip 1 data. The s
ize vignetted by rail can be variable.

fitgeo ... If the data to combine is the one taken in single night, the images
can be registered each other by 'shift' or 'rotate'. However, the matching can b
e more complex if the data were taken in multiple epoch with high/low elevation.
In such case, you can test matching by setting this to 'general'.

lsigma, hsigma, pclip, combine, reject, zero, weight ... see parapmeters in imco
mbine.

nmat   ... The same as the nmatch parameter in the iraf xyxymatch. If it is set
 larger than 50, the process will take long time. Keep the parameter <50.

tol, nrej, sepa, ratio ... The same parameters in xyxymatch. See the iraf help.

fstop .... If it is set yes, the task will pause after the catalog matching proc
ess. If the matching is successful, you can proceed to the image registeration a
nd final combine process by ansering "yes" at the pause.

fskip .... You can use the xyxymatch result you ran last time and proceed to the
 image registeration and final combine process.

skip, gtrlst ... see below.

disp ... display the result.

Please note that the final resulting image will be scaled to the unit-exposure (
1sec) by dividing by the Exposure time. The weight will be applied by the sky r
ms value (sigma) or the exposure time (exposure) during final combine.
The list of the registered, exposure-corrected images for final combine (=gtrlst
) is automatically set to 'GTRmos_gcall.lst' for the current case.

e.g.)
mcsred> gmkgtrimages mos_gcall.lst MOSRESULT.fits fitgeo=rotate nmat=30 tol=2 fs
top+

Then all data is registered 'GTRmos....fits" with the maskname of 'mask_GTRmos..
.fits". The list of the images (gtrlst) for the above case is 'GTRmos_gcall.lst'
.

If you want to combine the same data by a different combine/rejection algorithm
after the first run, or if you want to delete some files from the final combine-
file list (gtrlst), you can use the parameter "skip". With "skip=yes", it just r
epeat the final combine process only using the registered images (i.e., files in
 gtrlst) of the previous run.

The matching log etc will be written in "gmpmos_gcall.lst.log" for this case.

exp_(output image name).pl ... exposure map data.
sgm_(output image name) ... sigma image
(output image name).pl ... mask image for region less than 60% of the total expo
sure.

<<< 4. Two-path process >>>
Making better object masks can be done using the result of the final combined im
age made by mcsall. Repeating the reduction procedure by these better object ma
sks (two-path process) is often recommended in the NIR image reduction process.
In MCSRED, the task `invmask' will create the object masks using the resulting i
mage of mcsall (or more precisely, gmkgtrimages.cl). Below is the example of the
 two-path processing.

2-path process has three steps.
1) A quick data reduction by mcsall.cl for individual channnel.
2) Making the master object mask by invmask.cl using the result of (1),

3) Reduce the data again by mcsall.cl using the result of 2).

1) A quick data reduction by mcsall.cl for individual channnel.

mcsred> mcsall raw_1.lst tempSFlat_1.fits TEMP_result_1.fits skipmask+ dosf+ eac
hflat- adsub- nsf=2 nmin=4 exelc- thres=3 minfw=3 maxfw=10 nmat=20

"skipmask+ (skipmask=yes)" ... Skip the quick object mask routine for speed up.
"dosf+" ... Use the self flat (name is "tempSFlat_1.fits") for flatter sky. Use
"eachflat-" for speed-up.
"adsub-" ... default.
"nsf=3 nmin=5" ... using 3 frames each before and after the image for sky subtra
ction. nsf=2 or 3 is used for speed up.
"exelc-" ... ignore bias drift pattern for speeding.
"thres=3 minfw=3 maxfw=10 nmat=20" ... image-dependent parameters. smaller nmat
may be better for speeding.

2) Making the master object mask by invmask.cl using the result of (1),
  mcsred> invmask gcclSBsbflbsraw_1.lst TEMP_result_1.fits

invmask.cl has the parameters below.

```
* invmask.cl
       gcfile = "gcclSBsbflbsraw_1.lst" the list of gc*fits images
       resimg = "TEMP_result_1.fits" result image from 1st roop
       (sigma = 2.5)            gaussian sigma for smoothing to resimg
          (sc = 1.1)            threshold value for object mask
        (mkflat = no)           Make new selfflat using the new mask?
        (nflat = "")            The name of new flatframe
  (configpars = "")             the mcsred config parameter
        (mode = "q")
```

The input list (gcfile) is the output file list (gc*.lst) from the process 5 of
mcsall.cl.

Some adjustment may be better for sigma and sc for better master object mask.

Note that the task assumes execution under the directory in which the 1st step i
s executed.

Also note that usually the new selfflat is automatically made in the mcsall task
 so you can leave mkflat=no for most cases.

2') Deleting the temporary files by the 1st path.
the task clean2nd can be used to delete all the temporary files up to the proces
s above.

```
mcsred> lpar clean2nd
       inlist = ""              input listfile name (listprep output)
      (res1st = )               The combined channel data from the 1st process
      (del1st = no)             Delete the combined data from the 1st process?
        (mode = "q")
```

mcsred> clean2nd data_2.lst

If you do not need to keep the combined result from the 1st path, you can also d
elete them by adding "res1st=(result file name) del=yes" to it.

3) Reduce the data again by mcsall.cl using the result of 2).
  mcsred> mcsall raw_1.lst DomeFlat_1.fits RES_2nd_ch1.fits second=yes exelc+ ns
f=5 nmin=7 thres=3 minfw=3 maxfw=10 nmat=20

Here we do the full-reduction with "second=yes". Never forget to add this, in or
der to use the object mask made in 2).

The example above uses Dome Flat (DomeFlat_1.fits with dosf-). If self flat is u
sed, you can use the result of step 2, if the number of input file is much large
r than 20. However, the same file is anyway made in the Step 2. If the number of
 file is less than 20, do not forget to add eachflat=yes.

exelc+ is to remove bias drift pattern and should add here.

You will be asked to delete the file "CATgcc(inlist).lst" at the beginning of th

e Step 5 if you did not delete the temporary files by clean2nd task. You may wan
t to delete the file before the start of the 2nd mcsall.

[5] In the end...
Though they are quite messy and primitive, I strongly recommend to read the sour
ce cl scripts of MCSRED when you use the package. The better results may be ach
ieved for better choice of the task parameters inside the scripts (IRAF task con
tains quite many operation parameters).

Please refer the MCSRED website as well as the credit for Ichi Tanaka as an auth
or when you publish any scientific papers based on MCSRED pipeline. There is als
o a paper that briefly describe about the MCSRED package (Ichi Tanaka et al. 201
1, PASJ, 63, S415), and I would be grateful if you can refer the paper.

The MCSRED package may be continuously updated in the future. Please ckech the
most recent update through the website below (or you can contact me via email).

http://www.naoj.org/staff/ichi/MCSRED/mcsred.html

Useful comments/suggestions are greatly appreciated. Questions should be direct
ed to ichi@naoj.org.

If you need any instruction, I can demonstrate the use of mcsred if you can stay
 Subaru Telescope Office.

==
ICHI TANAKA, Ph.D

Subaru Senior Support Astronomer
(MOIRCS)

650 North A'ohoku Place
Hilo, Hawaii 96720
Tel: +808-934-5981
Fax: +808-934-5099
===