# How-To

# Multiple Level Tracking

## MLT_4

Written by: Monika Ellwarth
August 2018

# 1 General use of MLT_4

The Multiple Level Tracking (MLT) procedure is a multiple-level pattern recognition code, written by Burkart Bovelet[1]. It is an IDL-based algorithm which is capable of identifying granules on high-resolution images of the solar photosphere. It is basically a routine to make statistical statements about the granulation pattern.

To use the main procedure **MLT_4**, the function **MLT_4F** and the procedure **LIST** had to get compiled first.
Calling sequence to run MLT_4 for demonstration under *Unix* or *Linux* with keyword parameter **unix=1** else under *MicroSoft Windows* with **unix=0**.

## 1.1 Call MLT_4 Procedure

The easiest way to call MLT_4 to read in a single image in byte-format is:

```
MLT_4, sbyt='bytearray.BYT, fram=bytarr(200,200)
```

**sbyt** Name of image in byte-formate, as a string

**fram** Byte array in the size of the byte image

### Choose Box

```
MLT_4, sbyt='bytearray.BYT, fram=bytarr(200,200),bxco=[1, 180, 5, 160]
```

**bxco** Coordinates of the image which define an area: $[x_0, x_1, y_0, y_1]$

## 1.2 Read in Time Series

To read in more than one byte-image at once, the easiest way is to add counting numbers to the corresponding filenames. For example:

```
testimg.byt0
testimg.byt1
testimg.byt2
.
.
.
```

Then the procedure can get called like:

```
MLT_4, sbyt='testimg.byt*, fram=bytarr(200,200),bxco=[1, 180, 5, 160]
```

[1]Source code: `http://wwwuser.gwdg.de/~astronom/`

## 1.3 Keywords

```
keyword (sbyt) = substring to match filenames in a series of BYTE frames
keyword (sint) = substring to match INTEGER frames corresponding to (sbyt)
keyword (smap) = substring to match BYTE maps with selective px flagged
keyword (stag) = string of file giving tagged INTEGER map OR tagged BOX
keyword (rpre) = number of precedent frames to track reappearing features
keyword (head) = array defined to skip header data when reading image file
keyword (fram) = array defined to read image data from each BYTE file
keyword (rnum) = [n1,n2] integer number range for (ns) sequential files
keyword (arpx) = latitude scale [" /px] for features represented by px
keyword (kmpx) = latitude scale [km/px] for features represented by px
keyword (xbox) = 1 for maximum box available in series of aligned frames
keyword (bxco) = intervals of box coordinates, (min,max) of columns, rows
keyword (ldet) = [top...base] array of descending BYTE detection levels
keyword (ddet) = BYTE increment to implement equidistant levels in (ldet)
keyword (lcel) = lowest BYTE level for cells to normalize intensities
keyword (imer) = normalized intensity level for merging shrunk features
keyword (nmer) = minimum number of shared contour-px to merge features
keyword (imex) = minimum of feature scaled intensity mean to skip merging
keyword (itop) = normalized intensity level to count (ntop) feature top px
keyword (ntop) = maximum count of pixels exceeding (itop) to skip merging
keyword (icut) = normalized cut-off level to shrink cellular features
keyword (igpx) = maximum size [px] of features rated as inter-granular ones
keyword (mnpx) = minimum number of pixels required for final feature sizes
keyword (mxpx) = maximum number of pixels required for final feature sizes
keyword (iscl) = value for scaling INTEGER intensity or else BYTE intensity
keyword (mnis) = minimum scaled intensity mean required for final features
keyword (mxis) = maximum scaled intensity mean required for final features
keyword (xedg) = 1 to exclude partial features cut by box edges, 0 to keep
keyword (xmap) = 0 to exclude features covering map pixels flagged by <(1B)
keyword (xmap) = 1 to interactively flag MAP pixels (2B) to select features
keyword (xmap) = 1 to exclude features covering map pixels flagged by <(2B)
keyword (xmap) = 2 to interactively flag MAP fields (3B) to select features
keyword (xmap) = 2 to exclude features covering map pixels flagged by <(3B)
keyword (zoom) = 1 to scale up box displayed, -1 to scale down, 0 to keep
keyword (tune) = 1 to use interactive tools for level tuning and monitoring
keyword (save) = 1 to save feature data, intermediate steps, plot figures
keyword (unix) = 1 if operating system is UNIX/LINUX (default) else WINDOWS
keyword (slab) = additional string to label filenames for tags and data
```

### sbyt (STRING)

String of serial filenames, required to end with digits for serial images set.
String to match filenames in a series of BYTE frames.

### sint (STRING)

Corresponding INTEGER frames to BYTE frames. If there is a series of images, also a digit is required on the end of the name.
If this kexword is missing, the BYTE intensity is taken for the INTEGER intensity.

### smap (STRING)

String to match BYTE MAPS for reference.

### fram (ARRAY)

ARRAY which defining size of frame of BYTE frame.

### mnpx (INTEGER)

Gives the minimum number of pixels the size of the features will have.

### mxpx (INTEGER)

Gives the maximum number of pixels the size of the features will have.

## 1.4 Demo Code

```
device,retain=2                          ; provide BACKING PIXMAP for X-windows
device,true_color=24                     ; route image through RGB COLOR table
!prompt='> '                             ; establish a shortcut of PROMPT symbol

.run ./MLT_4F.pro                        ; 1. compile implemented FUNCTION MLT_4F
.run ./MLT_4.pro                         ; 2. compile the main PROCEDURE MLT_4
.run ./LIST.pro                          ; 3. compile tool to LIST output DATA


MLT_4, sbyt='IMAGE.BYT'                 ,$  ; STRING to match file of the BYTE frame
       sint='IMAGE.INT'                 ,$  ; STRING to match corresponding INTEGER
       fram=bytarr(399,401)             ,$  ; ARRAY of BYTE FRAME to read from files
       arpx=0.04                        ,$  ; SCALING of actual pixel width [arcsec/px]
       kmpx=29.0                        ,$  ; SCALING of actual pixel width [km/px]
       xbox=0                           ,$  ; 1 to use MAXIMUM FIELD, else define BOX
       bxco=[1,398, 60,321]             ,$  ; BOX [COLUMN lower,upper, ROW lower,upper]
       ldet=[251,0]                     ,$  ; BYTE RANGE applied for TOP-DOWN DETECTION
       ddet=10                          ,$  ; BINSIZE to define EQUIDISTANT BYTE levels
       lcel=0                           ,$  ; BASIC BYTE LEVEL to expand feat. to CELLS
       icut=0.35                        ,$  ; NORM. CUT-OFF threshold to SHRINK features
       imer=0.52                        ,$  ; NORM. reference threshold used for MERGING
       nmer=4                           ,$  ; MIN. of common CONTOUR px to MERGE features
       imex=1.15                        ,$  ; MIN. scaled brightness to EXCLUDE MERGING
       itop=0.85                        ,$  ; NORM. threshold to count (nmex) TOP px
       ntop=20                          ,$  ; NUMBER of px >(lmex) to EXCLUDE MERGING
       mnpx=4                           ,$  ; MIN. AREA SIZE [px] required for features
       xedg=1                           ,$  ; 1 to EXCLUDE PARTIAL features at BOX EDGES
       zoom=1                           ,$  ; 1 for SCALING UP else -1 for scaling DOWN
       tune=1                           ,$  ; 1 for INTERACTIVE mode, 0 for BATCH mode
       save=1                           ,$  ; 1 for ANALYSES and PLOTS else 0 to skip
       slab='DEMO'                          ; STRING to LABEL new files for data & tags
```

# 2 LIST

## 2.1 Call LIST Procedure

LIST restores a list of the identified granules, including their size in pixel and km. Read in the results from MLT_4, for one image, with:

```
LIST, file='MLT_4.data.arraynumber', kmpx=70
```

**file** File which has been generated by MLT_4

**kmpx** Size scale for km / px

LIST restores a list of the different granules of this image with the corresponding size values as well as the coordinates of a pixel inside a granule, like shown in Fig. 1.

```
================================================

FEATURES:    16         SIZE SCALE:  70.0 km / px

================================================

    TAG   AREA   DIAM       <I>    COL    ROW

     #     px     km      I_ph      x      y
    ---   ---    ---      ----    ---    ---
      1   358   1494      1.15     13     66
      2   242   1229      1.15     63     77
      3    99    786      1.17     70     76
      4   252   1254      1.19     41     45
      5   138    928      1.18     42     56
      6   214   1155      1.11     31     60
      7   298   1364      1.07     43     87
      8   101    794      1.13     36     36
      9   193   1097      1.09     69     55
     10   244   1234      0.97     38     19
     11   430   1638      1.11     58     17
     12   206   1134      0.98     58     51
     13   139    931      1.02     26     74
     14   128    894      1.03     21     89
     15   106    813      0.98     40     70
     16   106    813      0.84     22     12

================================================
```

Figure 1: Outcome of the LIST procedure. The different detected granules are tagged. The size is given back as size of the area in pixels, and the corresponding diameter in km. Additionally there is a mean intensity value given, as well as x and y parameters inside of the tagged granule.