```
Handling of IRBis -- summary

- scripts and all necessary files, as parameter file templates, are in $SCRIPTS
  (the directory imarec/, which is linked to $SCRIPTS, can be downloaded from
   https://gitlab.oca.eu/MATISSE/tools/-/tree/master/imarec)

- the scripts for easy handling of IRBis (mat_cal_imarec) are $SCRIPTS/mat_cal_imarec_all.2.csh and
  $SCRIPTS/mat_cal_imarec_all.csh  (for the first tests, $SCRIPTS/mat_cal_imarec_all.2.csh should be used)
  (in the following, only $SCRIPTS/mat_cal_imarec_all.2.csh is applied)

- the script is called with the parameter file mat_cal_imarec_all.2.par, which has to be copied from $SCRIPTS
  and edited:
        $SCRIPTS/mat_cal_imarec_all.2.csh mat_cal_imarec_all.2.par

- 1. action: estimation of the target size and image reconstruction parameter
     edit in the parameter file:
         set data       = ("insert the oifits data of the target, with/without path")
         set lambdaList = ("wavelength intervall for imaging; first run: 0.1 20.0 --> all spectral channels")
         set guess      = 1

- main results of 1. action in
  * Parameter.Estimation/data.parameter.txt:
  B) Estimated target size by fitting the V^2 data:
     * (2) Gaussian            --> FWHM = 7.518 mas (red. Chi^2 = 26.879,)
     * (3) Uniform disk        --> diameter = 10.119 mas (red. Chi^2 = 7013.583,)
     * (4) Fully darkened disk --> diameter = 11.539 mas  (red. Chi^2 = 481.906,)
     * (5) Lorentzian function --> FWHM = 8.209 mas (red. Chi^2 = 720.159,)

  C) Recommended size of the angular FOV and the size of the NxN pixel grid for the image reconstruction run:
    ==> a collection of NxN pixel grids and corresponding FOVs[mas]
        with the largest distance to the Fourier center = N/4 pixels (N/2 = highest frequency for a NxN pixel grid)
      * 16x16 pixels --> FOV = 10.5698 mas
      * 32x32 pixels --> FOV = 21.1396 mas
      * 64x64 pixels --> FOV = 42.2792 mas
      * 128x128 pixels --> FOV = 84.5584 mas
      * 256x256 pixels --> FOV = 169.117 mas
      (expirience: these FOV/N pairs should be used in image reconstruction!)

  * Parameter.Estimation/uv.ps (uv coverage); Parameter.Estimation/wavelengths.ps (spectral channels);
    Parameter.Estimation/gaussudfdda.ps (fits to the visibilities)
```

```
- 2. action: first image reconstruction run
   edit in the parameter file:
   * Switch image reconstruction ON
        set guess = 0
   * Select that NxN pixel grid where the corresponding FOV is ~4x target size, for example, in the case
     of an uniform disk size of ~10 mas, select:
      64x64 pixels --> FOV = 42.2792 mas
        set fov   = 42
        set grid  = 64
        set oradiusStart  = 20  (without limitations due to a binary mask, in order to not overlook details outside)
        set stepSize      = 1.0
   * Select one regularization function
        set regFuncs      = (-3) pixel difference quadratic enforcing smoothness (is a good beginning)
   * Select the fit start object and its size
        set startmode     =  # 2 = gaussian disc, 3 = uniform disc, 4 = fully darkened disk
        startparam        =  #  mode=2 -> FWHM [mas], mode=3 -> diameter [mas], mode=4 -> diameter [mas]
                              from Parameter.Estimation/data.parameter.txt (fit sizes) in B)

- main results of the image reconstruction run:
   * the results are in the directory BIS.*.Script2.E.1/
   * ASCII file E.liste contains Chi^2 values and image reconstruction parameters for each IRBis run E.? in one line
     (1. part: the runs are sorted according to increasing phase+visibility-qrec value (qrecmode = 1);
      2. part: the runs are sorted according to increasing phase-qrec value (qrecmode = 2))
   * in the postscript file qrecmode\=1/*.lin.ps are the result plots of each run sorted according to
     increasing phase+visibility-qrec value (qrecmode = 1 is in most cases the best choice)
   * the fits files of the reconstructions are in E.?/bestrec.fits (unconvolved)


- 2. action: second image reconstruction run
   * for this target (Uniform disk size ~10 mas) you should test with larger pixel grid, for example,
      128x128 pixels --> FOV = 84.5584 mas
        set fov   = 84
        set grid  = 128
        set oradiusStart  = 40
        set stepSize      = 1.0

- in further runs other regularization functions should be tested
        set regFuncs = (3 -4 4)
```