Interface to Image Reconstruction

Éric Thiébaut[1], Jonathan Léger[1], Micehl Tallon[1], John Young[2]
Gilles Duvert[3], Guillaume Mella[3], Laurent Bourges[3] & Martin Vannier[4]

[1]*CRAL,* [2]*Cambridge University,* [3]*JMMC/IPAG,* [4]*Lagrange*

June 3, 2015

1 Introduction

Many image reconstruction algorithms have been developed to process optical interferometric data. Using these algorithms may require substantial expertise (for instance you may have to learn a specific programming language). One of the objectives of the OPTICON JRA4 is to provide a common graphical user interface (GUI) to drive such imaging algorithms in a *user friendly* way. For obvious reasons, we also want to avoid rewriting the image reconstruction algorithms as much as possible. A model of interaction between these two pieces of software under these constraints has to be developed. This document aims at specifying a common software interface for image reconstruction algorithms.

On the image reconstruction side, this involves specifying exactly the inputs: OI-FITS file for the data, the initial image, and other settings such as the regularization and its hyper-parameters.[1] Given these inputs, the image reconstruction algorithm should start the reconstruction producing two kind of outputs:

Intermediate outputs: These consist in iteration informations printed on the standard output (for instance) and captured by the GUI and also the current solution in the form of an image, the corresponding model of the measurements (*e.g.*, the complex visibilities). These outputs will be used by the GUI to display the current solution, to compare the actual data with the current model, *etc.*

Final outputs: These are essentially the final image and consistent information about the pixel size, the orientation, *etc.* but also about the input

[1]In a previous discussion, John Young suggested to have a text file with a simple format to gives the values of the input parameters. The format of this file must be simple and such that it can be easily edited by a human and easily read by a software. Here a different proposition is made which exploits the flexibility of the FITS format.

data (*i.e.* to avoid a *traceability issue*), the reconstruction method and its parameters.

A few related points have to be considered:

1. The command line image reconstruction algorithms should (or not?) switch off their own display capabilities.

2. There must be a way to interrupt the image reconstruction algorithms so as to make sure they do release all their resources (no zombie process, undetached shared memory, etc.).

3. Restarting an image reconstruction (to continue the iterations or perhaps with a few settings changed) should simply be a matter to re-start the algorithm with, as its input image, the last produced image.

4. In case of errors, we must adopt some kind of conventions to make sure that the GUI can properly account for the errors.

To summarize, on entry:

- the data;
- image parameters (dimensions, pixel size, *etc.*);
- initial image;
- regularization and hyper-parameters;

On output (for the final and intermediate results):

- the current/final image;
- the model of the data;
- other relevant information;

To make the interfacing as simple as possible, it is proposed in this document to use a single entry file with al input parameters and data and a single output file with all the results. The flexibility of the FITS file format is exploited to store all these informations and data.

All image reconstruction software must be able to run from the command line and with two arguments: the names of the input and output files (possibly with some flag to indicate that the software must be run in a specific way). For instance:

```
command --batch input output
```

2 Input format

All existing algorithms take their input data in the OI-FITS format (Pauls et al., 2005). This format stores the optical interferometric data in FITS binary tables. More generally, a FITS file (Pence et al., 2010) consists in a number of so-called header data units (HDU), each HDU having an header part with scalar parameters specified by FITS keywords[2] and a data part. The header part obeys a specific format but it is textual and easy to read for an human. The data part usually consist in binary data stored in various forms. In this document we only consider FITS *images* (that is a multidimensional array of values of the same data type) and FITS *binary tables*. These are sufficient for our purposes.

By adding FITS HDU or introducing new FITS keywords, it is possible to provide any additional data (*e.g.* the initial image) and parameters required to run the image reconstruction process. Thus, it is proposed that the input file be a valid OI-FITS file (resulting from the merging of all the optical interferometric data to process) completed by HDU with additional binary data needed for the reconstruction (for instance, the initial image) and with scalar input settings provided as the values of FITS keywords. This is detailed in the following subsections.

2.1 Scalar input parameters

The first HDU, the so-called primary HDU, of a FITS file can only contains a, possibly empty, FITS image. As optical interferometric data are all stored in binary table, the primary HDU is almost empty. It is proposed that the input (scalar) parameters be provided as FITS keywords in the primary HDU. There are however some exceptions like the pixel size and the dimensions of the image which are specified by those of the initial image (see below) which may be stored in an other HDU. Non-scalar parameters, for instance the prior image for maximum entropy regularization, are stored in dedicated HDU (as images or binary tables) and are identified by the value of their `EXTNAME` keyword. Table 1 gives some examples of the input parameters.

2.2 Wavelength range

The primary objective is to consider monochromatic image reconstruction. The interferometric data is generally available at many wavelengths. The result will, in fact, be a gray image of the target built from the data in the wavelength range (inclusive) specified by the FITS keywords `WAVE_MIN` and `WAVE_MAX`. The wavelength range is given in meters.

[2] A FITS keyword consist in upper case latin letters, digits, hyphen or underscore characters and has at least one character and at most height characters.

Table 1: FITS keywords used to specify the input parameters. The *Image Parameters* are given in the HDU which stores the initial image (identified by the value of the `INIT_IMG` keyword), all other parameters are in the primary HDU.

Data Selection		
Keyword	Type	Description
`TARGET`	string	Identifier of the target object to reconstruct
`WAVE_MIN`	real	Minimum wavelength to select (in meters)
`WAVE_MAX`	real	Maximum wavelength to select (in meters)
`USE_VIS`	logical	Use complex visibility data if any
`USE_VIS2`	logical	Use squared visibility data if any
`USE_T3`	logical	Use triple visibility data if any
Algorithm Settings		
Keyword	Type	Description
`INIT_IMG`	string	Identifier of the initial image
`MAXITER`	integer	Maximum number of iterations to run
`RGL_NAME`	string	Name of the regularization method
`RGL_WGT`	real	Weight of the regularization
`RGL_ALPH`	real	Parameter α of the regularization
`RGL_BETA`	real	Parameter β of the regularization
`RGL_PRIO`	string	Identifier of the HDU with the prior image
Image Parameters		
Keyword	Type	Description
`NAXIS1`	integer	First dimension of the image
`NAXIS2`	integer	Second dimension of the image
`CDELT1`	real	Pixel size along first dimension of the image (in radians)
`CDELT2`	real	Pixel size along second dimension of the image (in radians)
Algorithm Results		
Keyword	Type	Description
`LAST_IMG`	string	Identifier of the final image

2.3 Initial image

All reconstruction algorithms are iterative and require an initial image to start with. The initial image is provided as a FITS image in one of the HDU of the input file. The primary HDU can be used to store the initial image in the primary HDU (however *cf.* the discussion about which image to store in the primary HDU). The pixel size and the dimensions of the initial image determine that of the sought image. The dimensions are given by the FITS keywords `NAXIS1` and `NAXIS2` while the pixel size is given by the FITS keywords `CDELT1` and `CDELT2` (both values must be the same).

It is not intended that the image reconstruction algorithm be able to deal with any world coordinate system (WCS) nor with any coordinate units. The images are therefore stored according to a given WCS with given units. First image axis corresponds to the right ascension (RA) and second image axis corresponds to the declination (DEC). Specify the units, the orientation and the ordering of pixels? The same conventions (WCS, units and orientation) hold for any output image produced by the reconstruction algorithm. For any external software to display correctly the images, the parameters of the WCS must be completely and correctly specified. This requires to set `CRPIXn`, `CRVALn`, `CTYPEn`, and `CUNITn` keywords. `CROTAn` may be omitted as its default value (zero) according to FITS standard is suitable for us.

3 Output format

It is proposed that the output format be as similar as possible as the input format. The output file must provide the reconstructed image but also some informations for analyzing the result. In particular, as each imaging algorithm may implement its own method to estimate the complex visibilities given the image of the object, it is necessary that the model of every fitted data be computed by the algorithm itself rather than by another tool.

3.1 Final/current image

To compare the initial and the final images, they must be stored in different HDU, the FITS keyword `EXTNAME` is used to distinguish them. As previously explained, the dimensions, pixel size and orientation of the output image(s) are the same as the initial image.

As most image viewers[3] are only capable of displaying the image stored in the primary HDU, we suggest to store the initial image in the primary HDU for the input file but to store the final or current image in the primary HDU of the output file. The idea is to have the most relevant image stored in the primary HDU. For the image reconstruction algorithm and for any software designed to display or analyze the results, the different images are distinguished by their names (given by their `EXTNAME` keyword). OK but the last image should have

[3] not all softwares deal with FITS files

Table 2: Colums inserted in OI-FITS binary tables to store the values given by the model. *NWAVE* is the number of wavelengths.

New columns in OI_VIS table		
Label	Format	Description
VISAMPMODEL	D(*NWAVE*)	Model of the visibility amplitude
VISPHIMODEL	D(*NWAVE*)	Model of the visibility phase in degrees
New column in OI_VIS2 table		
Label	Format	Description
VIS2MODEL	D(*NWAVE*)	Model of the squared visibility
New columns in OI_T3 table		
Label	Format	Description
T3AMPMODEL	D(*NWAVE*)	Model of the triple-product amplitude
T3PHIMODEL	D(*NWAVE*)	Model of the triple-product phase in degrees

a specific name, perhaps set LAST_IMG keyword in the primary HDU with name of the final image and let all reconstructed images be called as:

```
EXTNAME = 'OUTPUTn'
```

with *n* the iteration number.

In order to continue the iterations of a previous reconstruction run, the image reconstruction algorithm may be started with the final image instead of the initial one. To that end, there must be some means to specify which image to start with for the image reconstruction software. This is the purpose of the INIT_IMG keyword (see Table 1) in the primary HDU which indicates the name of the initial image.

3.2 Model of the data

The OI-FITS format (Pauls et al., 2005) specifies that optical interferometric data be stored in binary tables as columns with specific names. As there is no restriction that the tables only contain the columns specified by the standard, we propose to store the model of the data in the same tables by adding new columns. The names of the new columns are listed in Table 2. In this way it is very easy to compare the actual data and their model as computed from the reconstructed image and the instrumental model assumed by the reconstruction algorithm. Another advantage of this convention is that the same format can be exploited to store the values given by model fitting software.

References

T. A. Pauls, J. S. Young, W. D. Cotton, and J. D. Monnier. A data exchange standard for optical (visible/IR) interferometry. 117:1255–1262, November 2005. doi: 10.1086/444523.

W. D. Pence, L. Chiappetti, C. G. Page, R. A. Shaw, and E. Stobie. Definition of the Flexible Image Transport System (FITS), version 3.0. *A&A*, 524: A42, Nov 2010. ISSN 1432-0746. doi: 10.1051/0004-6361/201015362. URL http://fits.gsfc.nasa.gov/fits_standard.html.