SCAMP Documentation

Release 2.6.2

E. Bertin, A. Moneti

Mon Nov 06 2017

Contents

CHAPTER 1

Contents

1.1 Introduction

SCAMP[1] (Software for Calibrating AstroMetry and Photometry) is a computer program that computes astrometric projection parameters from source catalogues derived from FITS (Flexible Image Transport System)[2] images. The computed solution is expressed according to the WCS[3] standard. The main features of **SCAMP** are:

- Compatibility with **SExtractor**[4] FITS or Multi-Extension FITS catalogue format in input,
- Generation of WCS-compliant and **SWarp**[5]-compatible FITS image headers in output,
- Automatic grouping of catalogues on the sky
- Selectable on-line astrometric reference catalogue
- Automatic determination of scale, position angle, flipping and coordinate shift using fast pattern-matching
- Various astrometric calibration modes for single detectors and detector arrays
- Combined astrometric solutions for multi-channel/instrument surveys
- Highly configurable astrometric distortion polynomials
- Correction for differential chromatic refraction
- Proper motion measurements
- Multi-threaded code that takes advantage of multiple processors.
- VOTable[6]-compliant XML (eXtensible Markup Language)[7] output of meta-data.
- XSLT (eXtensible Stylesheet Language Transformations)[8] filter sheet provided for convenient access to metadata from a regular web browser.

1 http://astromatic.net/software/scamp
2 http://fits.gsfc.nasa.gov
3 http://www.atnf.csiro.au/people/mcalabre/WCS/index.html
4 http://astromatic.net/software/sextractor
5 http://astromatic.net/software/swarp
6 http://www.ivoa.net/documents/VOTable
7 http://en.wikipedia.org/wiki/XML
8 http://en.wikipedia.org/wiki/XSLT

1.2 Installing the software

1.2.1 Hardware requirements

SCAMP runs in (ANSI) text-mode from a shell. A window system is not necessary for basic operation.

The amount of memory required depends mostly on the size of the input catalogues and on the number of exposures and astrometric "contexts" (stable instruments) involved in the astrometric solution. Each detection in the input catalogues amounts to about 140 bytes, plus a few tens of kbytes for every FITS table. To this one should add the memory space used by the normal equation matrix, which is $8 \times N_{\rm T}^2$ bytes, with, in the default **SCAMP** configuration,

$$N_{\rm T} = N_{\rm ast} \times N_{\rm arr} \times N_{\rm P_{arr}} + (N_{\rm exp} - N_{\rm ast}) \times N_{\rm P_{foc}},$$

where $N_{\rm arr}$ is the number of focal plane arrays (extensions) in each exposure, $N_{\rm P_{arr}}$ the number of polynomial terms describing the static distortion pattern of each array (20 for a $3^{\rm rd}$ in x and y), $N_{\rm P_{foc}}$ the number of polynomial terms for the exposure-dependent focal plane distortion pattern, and $N_{\rm exp}$ the number of exposures. Actually one should probably double the memory space used by the normal equation matrix to account for buffers in the ATLAS library. It is not uncommon to see memory usage amounting to gigabytes when many large mosaic exposures are involved. For instance, computing an astrometric solution with $N_{\rm P_{arr}} = 60$ and $N_{\rm P_{foc}} = 6$ for a set of 500 exposures of a 60-CCD camera, each with 10,000 detections, spread over three runs and five bands, may consume as much as 8GB of memory.

Although multiple CPU cores are not required for running **SCAMP**, they can dramatically reduce execution time, especially when the solution is computed over a large number of exposures.

1.2.2 Obtaining SCAMP

For Linux users, the simplest way to have **SCAMP** up and running is to install the standard binary package the comes with your Linux distribution. Run, e.g., `apt-get scamp` (on Debian) or `dnf scamp` (Fedora) and **SCAMP**, as well as all its dependencies, will automatically be installed. If you decided to install the package this way you may skip the following and move straight to the *next section*.

However if **SCAMP** is not available in your distribution, or to obtain the most recent version, the **SCAMP** source package can be downloaded from the official GitHub repository[9] . One may choose one of the stable releases[10], or for the fearless, a copy of the current master development branch[11].

1.2.3 Software requirements

SCAMP has been developed on GNU/Linux[12] machines and should compile on any POSIX[13]-compliant system (this includes Apple OS X®[14] and Cygwin[15] on Microsoft Windows®[16], at the price of some difficulties with the configuration), provided that the *development* packages of the following libraries have been installed:

- **ATLAS**[17] V3.6 and above[25],
- **FFTw**[18] V3.0 and above[26],

[9] https://github.com/astromatic/scamp
[10] https://github.com/astromatic/scamp/releases
[11] https://github.com/astromatic/scamp/archive/master.zip
[12] http://en.wikipedia.org/wiki/Linux
[13] http://en.wikipedia.org/wiki/POSIX
[14] http://www.apple.com/osx
[15] http://www.cygwin.com
[16] http://www.microsoft.com/windows
[17] http://math-atlas.sourceforge.net
[25] Use the `--with-atlas` and/or `--with-atlas-incdir` options of the **SCAMP configure** script to specify the **ATLAS** library and include paths if **ATLAS** files are installed at unusual locations.
[18] http://www.fftw.org
[26] Make sure that **FFTw** has been compiled with **configure** options `--enable-threads --enable-float`.

- **PLPlot**[19] V5.9 and above.

On Fedora/Redhat distributions for instance, the development packages above are available as `atlas-devel`, `fftw-devel` and `plplot-devel`. **PLPlot** is only required for producing diagnostic plots. Note that **ATLAS** and **FFTw** are not necessary if **SCAMP** is linked with Intel®'s MKL (Math Kernel Library)[20] library.

1.2.4 Installation

To install from the GitHub source package, you must first uncompress the archive:

```
$ unzip scamp-<version>.zip
```

A new directory called `scamp-<version>` should now appear at the current location on your disk. Enter the directory and generate the files required by the autotools[21], which the package relies on:

```
$ cd scamp-<version>
$ sh autogen.sh
```

A **configure** script is created. This script has many options, which may be listed with the `--help` option:

```
$ ./configure --help
```

No options are required for compiling with the default GNU C compiler (**gcc**) if all the required libraries are installed at their default locations:

```
$ ./configure
```

Compared to **gcc** and the librairies above, the combination of the Intel® compiler (**icc**) and the MKL[22] libraries can give the **SCAMP** executable a strong boost in performance, thanks to better vectorized code. If **icc** and the MKL are installed on your system[27] , you can take advantage of them using

```
$ ./configure --enable-mkl
```

Additionally, if the **SCAMP** binary is to be run on a different machine that does not have **icc** and the MKL installed (e.g., a cluster computing node), you must configure a partially statically linked executable using

```
$ ./configure --enable-mkl --enable-auto-flags --enable-best-link
```

In all cases, **SCAMP** can now be compiled with

```
$ make -j
```

An `src/scamp` executable is created. For system-wide installation, run the usual

```
$ sudo make install
```

You may now check that the software is properly installed by simply typing in your shell:

```
$ scamp
```

which will return the version number and other basic information (note that some shells require the **rehash** command to be run before making a freshly installed executable accessible in the execution path).

[19] http://www.plplot.org
[20] http://software.intel.com/intel-mkl
[21] http://en.wikipedia.org/wiki/GNU_Build_System
[22] http://software.intel.com/intel-mkl
[27] The Linux versions of the Intel® compiler and MKL are available for free to academic researchers, students, educators and open source contributors.

1.3 Using SCAMP

SCAMP is run from the shell with the following syntax:

```
$ scamp Catalog1 [Catalog2 ...] -c configuration-file [-Parameter1 Value1 -
↪Parameter2 Value2 ...]
```

The parts enclosed within brackets are optional. The file names of input catalogues can be directly provided in the command line, or in lists that are ASCII files with each catalogue name preceded by `@` (one per line). One should use lists instead of the catalogue file names if the number of input catalogues is too large to be handled directly by the shell. Any *-Parameter Value* statement in the command-line overrides the corresponding definition in the configuration file or any default value (see below).

1.3.1 Input files

Catalogues

Catalogue files read by **SCAMP** must be in **SExtractor**[29]'s "FITS_LDAC" binary format. It is strongly advised to use **SExtractor** version 2.4.4 or later. The catalogues *must* contain all the following measurements in order to be processable by **SCAMP**:

- Centroid coordinates. They must be specified with the `CENTROID_KEYS` configuration parameter (default: `XWIN_IMAGE` and `YWIN_IMAGE`).
- Centroid error ellipse, as defined by the `CENTROIDERR_KEYS` configuration parameter (default: `ERRAWIN_IMAGE`, `ERRBWIN_IMAGE` and `ERRTHETAWIN_IMAGE`).
- Factors controlling astrometric distortion. These are set with the `DISTORT_KEYS` configuration parameter (default: `XWIN_IMAGE` and `YWIN_IMAGE`).
- Flux measurements, defined by the `PHOTFLUX_KEY` configuration parameter (default: `FLUX_AUTO`).
- Flux uncertainties, defined by the `PHOTFLUXERR_KEY` configuration parameter (default: `FLUXERR_AUTO`).

In addition, it is strongly advised (but not mandatory) to include the following optional **SExtractor** measurements:

- `FLAGS`, `FLAGS_WEIGHT` and/or `IMAFLAGS_ISO` flags for filtering out blended and corrupted detections.
- the `FLUX_RADIUS` half-light radius estimation for filtering out both small glitches and extended objects.
- `ELONGATION` to help filtering out objects such as trails and diffraction spikes
- a measurement of the object "spread" compared to that of the PSF model: `SPREAD_MODEL` and its uncertainty `SPREADERR_MODEL`. These measurements are not used by **SCAMP** for selection, but they get propagated to the output catalogues.

`.ahead` header files

The binary catalogues in "FITS_LDAC" format read by **SCAMP** contain a copy of the original FITS image headers. These headers provide fundamental information such as frame dimensions, World Coordinate System (WCS) data and many other FITS keywords which **SCAMP** uses to derive a full astrometric and photometric calibration. It is often needed to change or add keywords in some headers. Editing FITS files is not convenient, so **SCAMP** provides read (and write) support for "external" header files. External headers may either be real FITS header cards (no carriage-return), or ASCII files containing lines in FITS-like format, with the final line starting with the ''END '' keyword. Multiple extensions must be separated by an ''END '' line. External "headers" need not contain all the FITS keywords normally required. The keywords present in external headers are only there to override their counterparts in the original image headers or to add new ones.

[29] http://astromatic.net/software/sextractor

Hence for every input (say, `xxxx.cat`) FITS catalogue, **SCAMP** looks for a `xxxx.ahead` header file, loads it if present, and overrides or adds to image header keywords those found there. `.ahead` is the default suffix; it can be changed using the `AHEADER_SUFFIX` configuration parameter.

It is often useful to add/modify the same FITS keyword(s) in *all* input catalogues. **SCAMP** offers the possibility to put these keywords in one single external header file, which is read before all other `.ahead` files (but after reading the catalogue headers). The name of this file is `scamp.ahead` by default; it can be changed using the `AHEADER_GLOBAL` configuration parameter. **show an example of a typical .ahead file**

1.3.2 Output files

`.head` header files

SCAMP itself generates FITS header keywords, containing updated astrometric and photometric information. These keywords are written in ASCII to external header files, with the `.head` filename extension by default (the suffix can be changed with the `HEADER_SUFFIX` configuration parameter). In combination with the original image files, these .head headers are ready to be used by the **SWarp**[30] image stacking tool *[1]*.

The astrometric engine at the heart of **SCAMP** and **SWarp** is based on Mark Calabretta's **WCSLIB**[31] library *[2][3]*, to which we added support for the `TPV` description of polynomial distortions[35]. All celestial coordinate computations are performed in the equatorial system, although galactic and ecliptic coordinates are supported in input and output.

Output catalogues

SCAMP can save three kinds of catalogues: local copies of the reference catalogues downloaded from the Vizier server (see §[chap:astref]), a "merged", calibrated version of input catalogues (§[chap:mergedcat]), and a "full" calibrated version of input a reference catalogues (§[chap:fullcat]).

Diagnostic files

Two types of files can be generated by **SCAMP**, providing diagnostics about the calibrations:

- *Check-plots* are graphic charts generated by **SCAMP**, showing scatter plots or calibration maps. The `CHECKPLOT_TYPE` and `CHECKPLOT_NAME` configuration parameters allow the user to provide a list of check-plot types and file names, respectively. A variety of raster and vector file formats, from JPEG to Postscript, can be set with `CHECKPLOT_DEV`. PNG is the default. See the CHECKPLOT section of the *configuration parameter list* below for details.
- An XML[32] file providing a processing summary and various statistics in VOTable[33] format is written if the `WRITE_XML` switch is set to `Y` (the default). The `XML_NAME` parameter can be used to change the default file name `scamp.xml`. The XML file can be displayed with any recent web browser; the XSLT[34] stylesheet installed together with **SCAMP** will automatically translate it into a dynamic, user-friendly webpage. For more advanced usages (e.g., access from a remote web server), alternative XSLT translation URLs may be specified using the `XSL_URL` configuration parameter.

1.3.3 The Configuration file

Each time it is run, **SCAMP** looks for a configuration file. If no configuration file is specified in the command-line, it is assumed to be called `scamp.conf` and to reside in the current directory. If no configuration file is found, **SCAMP** uses its own internal default configuration.

[30] http://astromatic.net/software/swarp
[31] http://www.atnf.csiro.au/people/mcalabre/WCS/wcslib
[35] The `TPV` description was originally proposed by E. Greisen
[32] http://en.wikipedia.org/wiki/XML
[33] http://www.ivoa.net/documents/VOTable
[34] http://en.wikipedia.org/wiki/XSLT

Creating a configuration file

SCAMP can generate an ASCII dump of its internal default configuration, using the `-d` option. By redirecting the standard output of **SCAMP** to a file, one creates a configuration file that can easily be modified afterward:

$ scamp -d >scamp.conf

A more extensive dump with less commonly used parameters can be generated by using the `-dd` option.

Format of the configuration file

The format is ASCII. There must be only one parameter set per line, following the form:

Config-parameter *Value(s)*

Extra spaces or linefeeds are ignored. Comments must begin with a `#` and end with a linefeed. Values can be of different types: strings (can be enclosed between double quotes), floats, integers, keywords or Boolean (*Y/y* or *N/n*). Some parameters accept zero or several values, which must then be separated by commas. Values separated by commas, spaces, tabs or linefeeds may also be read from an ASCII file if what is given is a filename preceded with `@` (e.g. *@values.txt*). Integers can be given as decimals, in octal form (preceded by digit O), or in hexadecimal (preceded by *0x*). The hexadecimal format is particularly convenient for writing multiplexed bit values such as binary masks. Environment variables, written as `$HOME` or `${HOME}` are expanded.

Configuration parameter list

Here is a list of all the parameters known to **SCAMP**. Please refer to next section for a detailed description of their meaning. Some "advanced" parameters (indicated with an asterisk) are also listed. They must be used with caution, and may be rescoped or removed without notice in future versions.

and M. Calabretta in a 2000 draft, but abandoned in later versions *[4]*. Following adoption in **SCAMP** and in large data processing centers it eventually became a registered FITS convention in 2012[36], and is now included in recent versions of the **WCSLIB**.

1.4 Output Catalogues

SCAMP is run from the shell with the following syntax:

1.4.1 Merged Catalogues

In addition to astrometric header information, **SCAMP** can write out "merged" catalogues (one per field group). These catalogues contain the calibrated coordinates and magnitudes of a union of all detections from input catalogues that passed the **SCAMP** acceptation criteria (S/N, flags). Merged coordinates and magnitudes are computed using a weighted average of all overlapping measurements, and are accompanied by estimates of formal errors and 1-sigma uncertainties on individual measurements. The `MERGEDOUTCAT_TYPE` configuration parameter sets the format of the merged catalogue; it is set to `NONE` by default, which means that no catalogue is created. The available formats are `ASCII` (pure ASCII table), `ASCII_HEAD` (ASCII table with a small header describing the column content), and `FITS_LDAC` (FITS binary table). `FITS_LDAC` is the recommended format; `FITS_LDAC` files are smaller, carry the data with full precision, and can be read with popular software such as **fv** and **TOPCAT**. They can be converted to ASCII format with the `ldactoasc` command-line utility, which is part of the **SExtractor** package.

The columns present in the file are:

Name `SOURCE_NUMBER`

[36] http://fits.gsfc.nasa.gov/registry/tpvwcs.html

Content Source ID, the same as in "full" catalogue

Unit –

Name `NPOS_TOTAL`

Content Total number of overlapping positions

Unit –

Name `NPOS_OK`

Content Number of overlapping positions kept for astrometry

Unit –

Name `ALPHA_J2000`

Content (Weighted-)average Right-Ascension

Unit deg

Name `DELTA_J2000`

Content (Weighted-)average Declination

Unit deg

Name `ERRA_WORLD`

Content Position uncertainty (RMS) along major world axis (may include additional uncertainty computed by SCAMP)

Unit deg

Name `ERRB_WORLD`

Content Position uncertainty (RMS) along minor world axis (may include additional uncertainty computed by SCAMP)

Unit deg

Name `ERRTHETA_WORLD`

Content Position angle of error ellipse (CCW/world-x) (The current estimation of error ellipse parameters is still done very crudely)

Unit deg

Name `DISPALPHA_J2000`

Content Measured dispersion (RMS) of position along Right-Ascension

Unit deg

Name `DISPDELTA_J2000`

Content Measured dispersion (RMS) of position along Declination

Unit deg

Name `PMALPHA_J2000`

Content Apparent proper motion along Right-Ascension

Unit mas/yr

Name `PMDELTA_J2000`

Content Apparent proper motion along Declination

Unit mas/yr

Name `PMALPHAERR_J2000`

Content Proper motion uncertainty (RMS) along Right Ascension

Unit mas/yr

Name `PMDELTAERR_J2000`

Content Proper motion uncertainty (RMS) along Declination

Unit mas/yr

Name `PARALLAX_WORLD`

Content Trigonometric parallax

Unit mas

Name `PARALLAXERR_WORLD`

Content Trigonometric parallax uncertainty

Unit mas

Name `CHI2_ASTROM`

Content Reduced chi2 per d.o.f. of the proper motion/parallax fit

Unit `-`

Name `EPOCH`

Content (Astrometrically-)weighted-average of observation dates

Unit yr

Name `EPOCH_MIN`

Content Earliest observation date

Unit yr

Name EPOCH_MAX

Content Latest observation date

Unit yr

Name NMAG

Content Number of magnitude measurements in each photometric instrument"

Unit –

Name MAG

Content Vector of (ux-weighted-)average magnitudes

Unit mag

Name MAGERR

Content Vector of magnitude uncertainties

Unit mag

Name MAG_DISP

Content Vector of measured magnitude dispersions (RMS)

Unit mag

Name COLOR

Content Composite colour index computed by SCAMP

Unit mag

Name SPREAD_MODEL

Content (Weighted-)average of ‘‘SPREAD_MODEL‘‘s

Unit –

Name SPREADERR_MODEL

Content SPREAD_MODEL uncertainty

Unit –

Name FLAGS_EXTRACTION

Content Arithmetic OR of SExtractor flags over overlapping detection

Unit –

Name FLAGS_SCAMP

Content SCAMP flags for this detection

Unit –

1.4.2 Full Catalogues

Other catalogues which can be produced by **SCAMP** are the "full" catalogues (one per field group). These catalogues contain the raw and calibrated coordinates and magnitudes of all individual detections that passed the **SCAMP** acceptation criteria. Each detection is linked to a parent source through to the SOURCE_NUMBER identifier, also present in the merged output catalogue. The CATALOG_NUMBER identifier link the detected source to its catalogue (as **SCAMP** could receive more than one input catalogue), CATALOG_NUMBER set to 0 beeing reserved to the reference catalogue. The FULLOUTCAT_TYPE configuration parameter sets the format of the full catalogue; the choice of options is the same as for MERGEDOUTCAT_TYPE.

The columns present in the file are:

Name SOURCE_NUMBER

Content Source ID, the same as in "merged" catalogue

Unit –

Name CATALOG_NUMBER

Content Catalogue ID, 0 beeing reserved to the reference catalogue

Unit –

Name EXTENSION

Content FITS extension of the parent image (always set to 1 for single extension images)

Unit –

Name ASTR_INSTRUM

Content Astrometric instrument (context) ID

Unit –

Name PHOT_INSTRUM

Content Photometric instrument (context) ID

Unit –

Name X_IMAGE

Content x pixel coordinate of centroid

Unit pixel

Name Y_IMAGE

Content y pixel coordinate of centroid

Unit pixel

Name ERRA_IMAGE

Content Position uncertainty (RMS) along major error ellipse axis

Unit pixel

Name ERRB_IMAGE

Content Position uncertainty (RMS) along minor error ellipse axis

Unit pixel

Name ERRTHETA_IMAGE

Content Position angle of error ellipse (forced to 0 by current versions of SCAMP which isotropise input uncertainties.)

Unit deg

Name ALPHA_J2000

Content Calibrated Right-Ascension of centroid in the ICRS (at epoch of observation)

Unit deg

Name DELTA_J2000

Content Calibrated Declination of centroid in the ICRS (at epoch of observation)

Unit deg

Name ERRA_WORLD

Content Position uncertainty (RMS) along major world axis (may include additional uncertainty computed by SCAMP)

Unit deg

Name ERRB_WORLD

Content Position uncertainty (RMS) along minor world axis (may include additional uncertainty computed by SCAMP)

Unit deg

Name ERRTHETA_WORLD

Content Position angle of error ellipse (CCW/world-x) (The current estimation of error ellipse parameters is still done very crudely)

Unit deg

Name EPOCH

Content Julian date at start of observation

Unit yr

Name MAG

Content Calibrated magnitude

Unit mag

Name `MAGERR`

Content Magnitude uncertainty (may include additional uncertainty computed by SCAMP.)

Unit mag

Name `MAG_DISP`

Content Vector of measured magnitude dispersions (RMS)

Unit mag

Name `SPREAD_MODEL`

Content `SPREAD_MODEL` parameter

Unit –

Name `SPREADERR_MODEL`

Content `SPREAD_MODEL` uncertainty

Unit –

Name `FLAGS_EXTRACTION`

Content SExtractor flags

Unit –

Name `FLAGS_SCAMP`

Content SCAMP flags for this detection

Unit –

CHAPTER 2

Indices and tables

- genindex
- modindex
- search

Bibliography

[1] E. Bertin, Y. Mellier, M. Radovich, G. Missonnier, P. Didelon, and B. Morin. The TERAPIX Pipeline.[37] In D. A. Bohlender, D. Durand, and T. H. Handley, editors, *Astronomical Data Analysis Software and Systems XI*, volume 281 of Astronomical Society of the Pacific Conference Series, 228. 2002.

[2] E. W. Greisen and M. R. Calabretta. Representations of world coordinates in FITS.[38] *aap* , 395:1061–1075, 2002.

[3] M. R. Calabretta and E. W. Greisen. Representations of celestial coordinates in FITS.[39] *aap* , 395:1077–1122, 2002.

[4] M. R. Calabretta, F. Valdes, E. W. Greisen, and S. L. Allen. Representations of distortions in FITS world coordinate systems.[40] In F. Ochsenbein, M. G. Allen, and D. Egret, editors, *Astronomical Data Analysis Software and Systems (ADASS) XIII*, volume 314 of Astronomical Society of the Pacific Conference Series, 551. July 2004.

[37] http://adsabs.harvard.edu/abs/2002ASPC..281..228B
[38] http://adsabs.harvard.edu/abs/2002A&A...395.1061G
[39] http://adsabs.harvard.edu/abs/2002A&A...395.1077C
[40] http://adsabs.harvard.edu/abs/2004ASPC..314..551C