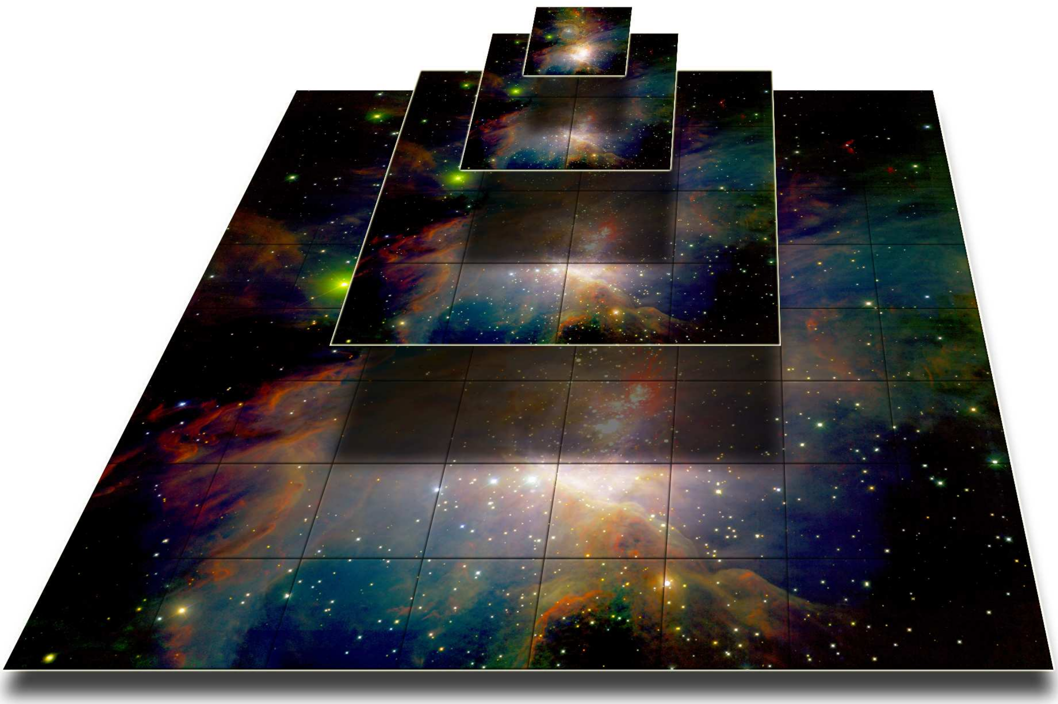
STIFF
*v*2.2
User's guide

E. BERTIN
Institut d'Astrophysique de Paris

November 10, 2011



Contents

1 What is STIFF?

STIFF is a program that converts scientific FITS[1] images to the more popular TIFF[2] format for illustration purposes. The main features of STIFF are:

- Accurate reproduction of the original surface brightnesses and colours
- Automatic or manual contrast and brightness adjustments
- Automatic sky background intensity and colour balance
- Adjustable colour saturation
- Colour-friendly gamma correction capabilities
- One or three input channels: gray-scale or true colour output
- Output with 8 or 16 bits per component
- Pixel rebinning and x/y flip options
- Support for arbitrarily large input and output images on standard hardware (BigTIFF support)
- Support for tiled, multiresolution pyramids
- Support for lossless and lossy compression methods
- Multi-threaded code with load-balancing to take advantage of multiple cores and processors.
- XML VOTable-compliant output of meta-data.

2 Skeptical Sam's questions

Skeptical Sam doesn't have time to test software extensively but is always keen on asking agressive questions to the author to find out if a program could fit his needs.

S.Sam: I don't understand the purpose of this software. There seem to be already a lot of convenient FITS viewers and batch converters available out there.

Author: Yes, but most of them do not do the best job at converting FITS image data to 8 bits. It is often said that 8-bit images stored in JPEG, PNG or TIFF files are unable to cope with the high-dynamic range of 16 bits CCD images from the professional astronomy world. This is not entirely true (actually, perceptually, they even offer a larger contrast), because in these 8-bit file formats the intensities are implicitely stored in a non-linear way. But strangely, most current FITS image viewers and converters seem to ignore this fact, which leads to an inconsistent translation of the FITS image content by simply rescaling linearly input pixel values. A first consequence is that the people working on astronomical images usually have to apply narrow intensity cuts or square-root or logarithmic intensity transformations to actually *see* something on their deep-sky images! A less obvious consequence is that colours obtained by combining images processed this way are not consistent across such a large range of surface-brightnesses.

[1] `http://fits.gsfc.nasa.gov/`

[2] e.g. `http://www.awaresystems.be/imaging/tiff.html`

The picture below shows the same 3 FITS images passed through a standard FITS image converter and STIFF. The same cuts in surface brightness were applied. If your display or printer is properly calibrated, you should be able to distinguish most of the faint details on the rightmost image. On the left image on the contrary, most of the faint stuff is missed. This is because the aforementioned standard software ignores the non-linear behaviour of your display.

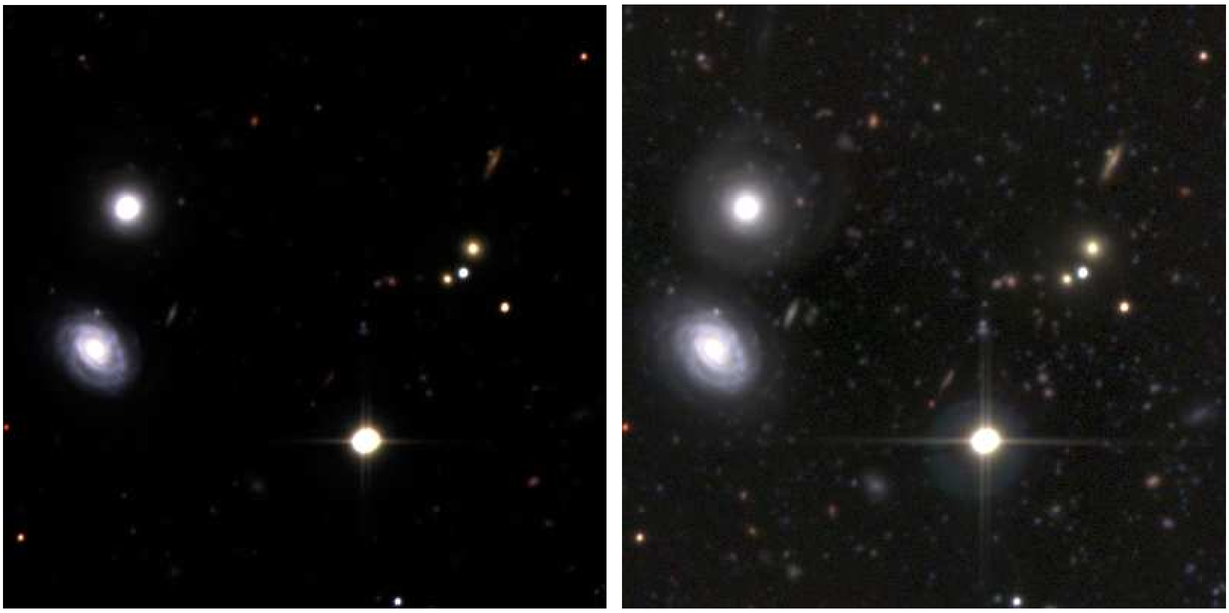


Figure 1: The same 3 R,G and B deep-sky images processed by a regular FITS image converter (left) and STIFF (right), using the same intensity cuts.

S.Sam: Well within many image viewers you are generally afforded a choice of non-linear transformations to apply in order to make the faint stuff stand out more clearly in the images.

Author: Sure, but with the limited selection of choices you are given colours will not be accurately rendered, and some manual tweaking will be necessary. The purpose of STIFF is to produce beautiful pictures in an automatic and consistent way.

3 License

STIFF is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. STIFF is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with STIFF. If not, see `http://www.gnu.org/licenses/`.

4 Installing the software

4.1 Software and hardware requirements

STIFF has been developed on Unix machines (GNU/Linux), and should compile on any POSIX-compliant system, provided that the following libraries/packages have been installed:

- LIBTIFF V3.6 and above[3] (`http://www.libtiff.org`)
- LIBJPEG V6.0 and above (`http://www.ijg.org`)
- ZLIB V1.2 and above (`http://www.zlib.net`)

All these libraries generally come installed with regular Linux distributions. Users requiring BigTIFF support for very large images (larger than about 1 Gpixel) however, will need to upgrade to LIBTIFF version 4.0, in beta stage as of January 2010. Note that a 64 bit system is required for processing images as large in STIFF.

The software is run in (ANSI) text-mode from a shell. The amount of memory required depends mostly on the size of the input images; in most cases, "physical" memory requirements are well below 100 megabytes. A large fraction of the program is parallelised, and will take advantage of multiple processor cores. Performance is also heavily dependent on the sequential disk read/write speed; it is recommended to operate STIFF with files located on a fast storage system such as a local RAID array, and avoid network-based file systems.

4.2 Obtaining STIFF

The easiest way to obtain STIFF is to download it from the official Astr*O*matic web site (`http://astromatic.net/software/stiff`). There can be found the latest versions of the package as standard `.tar.gz` and binary RPM archives, documentation, development repositories, and a support forum. The package is also directly available from the IAP anonymous FTP site at `ftp://ftp.iap.fr/pub/from_users/bertin/stiff/`.

4.3 Installation

For systems that support RPM packages, it is recommended to install the binary version of STIFF with the following command (typed as `root`):

```
rpm -U --force --nodeps stiff-x.x-1.y.rpm
```

where `x` is the version number and `y` the architecture (`i386` or `x86_64`). To install from the source package, you must first uncompress and "untar" the archive:

```
gzip -dc stiff-x.x.tar.gz | tar xvf -
```

A new directory called `stiff-x.x` should now appear at the current position on your disk. You should then just enter the directory and follow the instructions in the file called "`INSTALL`".

[3] Use the `--with-tiff-libdir` and `--with-tiff-incdir` options to specify the library and include paths if the libTIFF package has been installed at unusual locations.

5 Using STIFF

STIFF is run from the shell with the following syntax (single-channel mode):

% `stiff` *image* `-c` *configuration-file* `[-`*Parameter1 Value1*`]` `[-`*Parameter2 Value2 ...*`]`,

or in 3-channel mode:

% `stiff` *image_red image_green image_blue* `-c` *configuration-file* `[-`*Parameter1 Value1*`]` `[-`*Parameter2 Value2 ...*`]`.

The part enclosed within brackets is optional. Any "`-`*Parameter Value*" statement in the command-line overrides the corresponding definition in the configuration-file or any default value (see below).

5.1 The Configuration file

Each time STIFF is run, it looks for a configuration file. If no configuration file is specified in the command-line, it is assumed to be called "`stiff.conf`" and to reside in the current directory. If no configuration file is found, STIFF will use its own internal default configuration.

5.1.1 Creating a configuration file

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

```
% stiff -d >stiff.conf
```

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

5.1.2 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. Integers can be given as decimals, in octal form (preceded by digit `0`), 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. It is possible to include spaces in a string by enclosing the string in double-quotes `"`.

5.1.3 Configuration parameter list

Here is a list of all the parameters known to STIFF. Please refer to next section for a detailed description of their meaning. "Advanced" parameters are indicated with an asterisk. They must be used with caution, and may be re-scoped or removed in future versions.

`BIGTIFF_TYPE`* `NONE` *keyword*

BigTIFF support policy:

`AUTO`	Automatically switch to BigTIFF format for very large files if the TIFF library supports it
`NEVER`	Never use BigTIFF format and stick to basic TIFF in all cases
`ALWAYS`	Always use BigTIFF (if the TIFF library offers BigTIFF support), even for small files

`BINNING` `1` *integers*$(1 \leq n \leq 2)$

Pixel binning factor for both axes, or along each axis.

`BITS_PER_CHANNEL` `8` *integer*: 8 or 16

Number of bits per channel in the output image.

`COMPRESSION_QUALITY`* `90` *integer*

Quality factor (between 0 and 100) for lossy compression algorithms. Smaller numbers lead to a smaller file size but more prominent compression artifacts.

`COMPRESSION_TYPE`* `LZW` *keyword*

Image compression algorithm used in the TIFF file:

`NONE`	No compression
`LZW`	Lempel-Ziv-Welsh lossless algorithm
`JPEG`	JPEG lossy algorithm
`DEFLATE`	Deflate lossless algorithm
`ADOBE-DEFLATE`	Adobe-deflate lossless algorithm

`COLOUR_SAT` `1.0` *float*

Colour saturation factor.

`COPY_HEADER` `N` *Boolean*

when set to `Y`, a copy of the (first) FITS image header replaces the content of the `ImageDescription` tag in the output TIFF file .

`COPYRIGHT` `AstrOmatic.net` *string*

Copyright/author string to include in the output TIFF image header (`Copyright` tag).

`DESCRIPTION` `"STIFF image"` *string*

Description string to include in the output TIFF image header (`ImageDescription` tag).

`FLIP_TYPE`* `NONE` *keyword*

Flip the image:

`NONE`	No flipping
`X`	about the X axis
`Y`	about the Y axis
`XY`	about both the X and Y axes

`GAMMA` `2.2` *float*

Image *gamma* (exponent of the display intensity transfer curve) for `POWER-LAW` `GAMMA_TYPE`s.

`GAMMA_FAC` `1.0` *float*

gamma correction factor for the luminance image component.

`GAMMA_TYPE` `POWER-LAW` *keyword*

Gamma correction:

`POWER-LAW`	pure power law
`SRGB`	SRGB standard gamma correction
`REC.709`	ITU-R recommendation BT.709 (HDTV) standard gamma correction

`IMAGE_TYPE` `TIFF` *keyword*

Output image format:

`AUTO`	based on filename extension
`TIFF`	TIFF image format (scanline)
`TIFF-pyramid`	TIFF pyramid (tiled)

`MAX_LEVEL` `0.995` *floats* $(1 \leq n \leq n_{\rm ima})$

Upper end of image dynamic range (see `MAX_TYPE`).

`MAX_TYPE` `QUANTILE` *keywords* $(1 \leq n \leq n_{\rm ima})$

Meaning of the `MAX_LEVEL` parameter:

`QUANTILE`	`MAX_LEVEL` is taken as the upper quantile of the pixel histogram
`MANUAL`	`MAX_LEVEL` is directly interpreted as a pixel value

`MEM_MAX*` `1024` *integer*

Maximum amount of (physical) memory that can be used for processing.

`MIN_LEVEL` `0.001` *floats* $(1 \leq n \leq n_{\rm ima})$

Lower end of image dynamic range (see `MIN_TYPE`).

`MIN_TYPE` `GREYLEVEL` *keywords* $(1 \leq n \leq n_{\rm ima})$

Meaning of the `MIN_LEVEL` parameter:

`QUANTILE`	`MIN_LEVEL` is taken as the lower quantile of the pixel histogram
`MANUAL`	`MIN_LEVEL` is directly interpreted as a pixel value.
`GREYLEVEL`	`MIN_LEVEL` is interpreted as a grey level in the output image, a fraction of "pure white" (0.0 is pure black, 1.0 is pure white)

`NEGATIVE` `N` *Boolean*

Produces a negative image when set to `Y`.

`NTHREADS` `0` (multithreaded version) or `1` *integer*

Number of threads (processes) to be used for parallel computations. If `NTHREADS` is `0` the number of threads is automatically set to that of processor cores.

`OUTFILE_NAME` `stiff.tif` *string*

Output image file name.

`PYRAMID_MINSIZE*` `256` *integers* $1 \leq n \leq 2$

Minimum output image width and/or height (in pixels) at the topmost pyramid level in `TIFF-PYRAMID IMAGE_TYPE` mode.

`SKY_LEVEL` `0.0` *floats* ($1 \leq n \leq n_{\rm ima}$)
User-specified sky level in `SKY_TYPE MANUAL` mode.

`SKY_TYPE` `AUTO` *keywords* ($1 \leq n \leq n_{\rm ima}$)
Sky level determination in each input image:

`AUTO`	Sky level is automatically determined from the pixel data
`MANUAL`	Sky level is taken from the SKY_LEVEL value.

`SATUR_LEVEL` `40000.0` *floats* ($1 \leq n \leq n_{\rm ima}$)
Input image saturation level: colour is set to white if one of the images exceeds that level.

`TILE_SIZE`* `256` *integer*
Tile size in `TIFF-PYRAMID IMAGE_TYPE` mode.

`VERBOSE_TYPE` `NORMAL` *keyword*
Degree of verbosity on screen:

`QUIET`	No Output besides warnings and error messages
`NORMAL`	"Normal" display with messages updated in real time using ASCII escapes-sequences
`FULL`	Everything

`VMEM_DIR`* `.` *string*
Path of the directory where temporary files are written.

`VMEM_MAX`* `1048576` *integer*
Maximum amount of virtual memory (disk space in the `VMEM_DIR` directory) that can be used for processing.

`WRITE_XML` `Y` *Boolean*
If true (`Y`), an XML summary file will be written after completing the processing.

`XML_NAME` `stiff.xml` *string*
File name for the XML output of STIFF.

`XSL_URL`* `.` *string*
URL of an XSL style-sheet for the XML output of STIFF. This URL will appear in the `href` attribute of the `style-sheet` tag.

6 How STIFF works

6.1 Overview of the software

Basically, the action of STIFF can be summarized as reading one (monochromatic case), or three (colour case: one for each of the primary colours, respectively red, green, and blue) input FITS images, and saving a grayscale or colour TIFF image to disk. The work is done in two passes through the data:

1. Input images are examined, a histogram of pixel values is constructed, from which statistics are derived for the automatic determination of the low and high cuts in dynamic range.
2. The images are actually processed and converted to TIFF format.

The global layout of STIFF is presented in Fig. 2. Let us now describe in details the processing of STIFF, and how to control it.

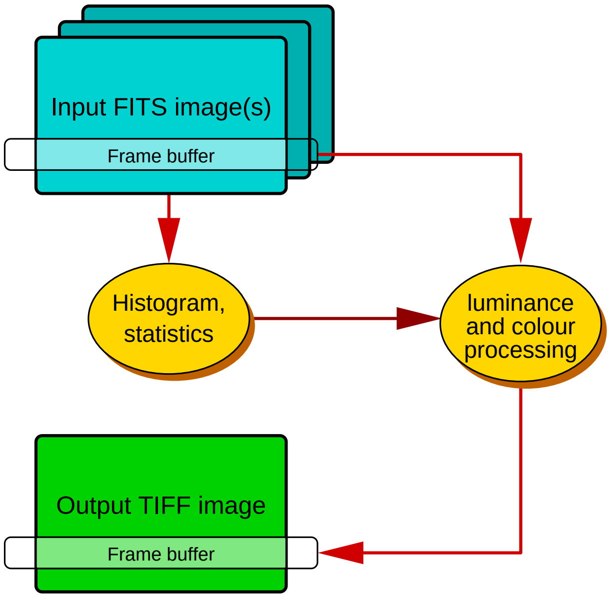


Figure 2: Global Layout of STIFF.

6.2 Gamma corrections

6.2.1 Gamma compensation

The true light intensity I' (or more formally, radiant emittance) radiated or reflected by a properly adjusted display device is related to the pixel value p stored in the computer's frame buffer through the power law

$$I' = p^{\gamma}, \tag{1}$$

where γ is what we will call the "gamma". This operation is sometimes called "gamma expansion". The display gamma is nowadays generally assumed to have an average value of 2.2, although in practice it may roughly vary from 1.7 to 2.5. A reverse transform (sometimes called "gamma compression") must therefore be applied before writing an image in GIF, JPEG, PNG or TIFF format. This also applies to other electronic image and video storage formats, including analog ones like Laserdiscs and VHS video tapes. The reason behind this non-linear relation between light intensity and voltage comes from the times when Cathode-Ray Tubes (CRTs) were ruling the world of electronic imaging: CRT screens have a natural response function $I' \propto p^{2.5}$ which has to be (at least partially) compensated for before being transmitted or stored. This non-linear transformation has many advantages[4]:

[4]More information on the issues related to gamma corrections can be found on the web; see, e.g. `http://www.cgsd.com/papers/gamma.html`, `http://www.normankoren.com/makingfineprints1A.html`, `http://www.poynton.com/GammaFAQ.html` or `http://www.marcelpatek.com/gamma.html`

- the recordable intensity range between the darkest and the brightest parts of an image is effectively increased (from 48dB to ≈ 100dB with 8 bits per pixel),
- quantization effects are reduced perceptually: the perception of light intensities by the human eye is itself non-linear and differences as small as 1% can be detected. Storing light intensities linearly with only 256 shades of grey would bring visible *banding* in the darker parts of an image,
- the signal is often photon-noise-limited on astronomical images; the standard deviation around the mean is proportional to the square-root of the flux. Hence storing the signal at the power of $1/\gamma$ gives a noise level close to constant over the pixel raster, which is optimal for quantization,
- although CRT displays are now disappearing, being progressively replaced by Liquid-Crystal Displays (LCDs), the latters do also have a "native" γ close to 2 in the dark parts of images: $I' \propto 1 - \cos(p/p_0)$ (however this is not the case for DLP, plasma and OLED devices).

In practice, several constraints with the display of real images (perceptual contrast matching, presence of ambient light, signal-to-noise ratio limitations, and colour-space conversion issues) have led engineers to alter the simple reverse law $p = I^{1/\gamma}$, and define instead

$$p = \begin{cases} a.I & \text{if } 0 \leq I < I_t \\ (1+b).I^{1/\gamma} - b & \text{if } I_t \leq I \leq 1 \end{cases} \quad (2)$$

Equation 2 allows the transfer curve to have a finite slope close to 0, and the effective gamma to vary smoothly with brightness.

STIFF applies automatically a gamma correction (compression) to pixel values before storing the data to TIFF. The `GAMMA_TYPE` configuration parameter lets the user choose between three types of gamma corrections: a "pure" power law with a user-selected γ and $b = I_t = 0$ (`POWER-LAW`, the default), the sRGB standard gamma correction, with $\gamma = 2.4$, $a = 12.92$, $b = 0.055$ and $I_t = 0.00304$ (`SRGB`), and the ITU-R rec. BT.709 gamma correction, with $\gamma = 2.22$, $a = 4.5$, $b = 0.099$ and $I_t = 0.018$ (`REC.709`). In "pure" power-law mode, γ may be modified by using the `GAMMA` configuration parameter. The defaut value is 2.2. It is usually not recommended to change it, especially for colour images; use `GAMMA_FAC` instead (see 6.2.2). The sRGB correction (Stokes et al. 1996) has been optimized for displays in the office environment, and is the default standard for images displayed on the web. The ITU-R rec. BT.709[5] is the standard for HDTV media.

The three types of gamma compensation are shown in Fig. 3, as well as the end-to-end transfer curves in viewing conditions typical of a computer LCD monitor watched in a dark environment.

Figure 4 shows images generated by STIFF using the three different `GAMMA_TYPE`s. The sRGB curve, and even more notably the Rec.709 curve, have both a tendency to bury much of the dark tones of the image into a black background. It is advised to stick to the pure power-law transfer curve for scientific or general use, as STIFF images currently lack the TIFF metadata with the proper transformation tables that would make sRGB and rec. BT.709 corrections handled automatically by software that supports advanced colour-management. The sRGB correction may however be useful for illustration purposes when the image is to be displayed on a dark background, or when the sky pixel values are very noisy. Rec. BT.709 is to be reserved for

[5] `http://www.itu.int/rec/R-REC-BT.709`

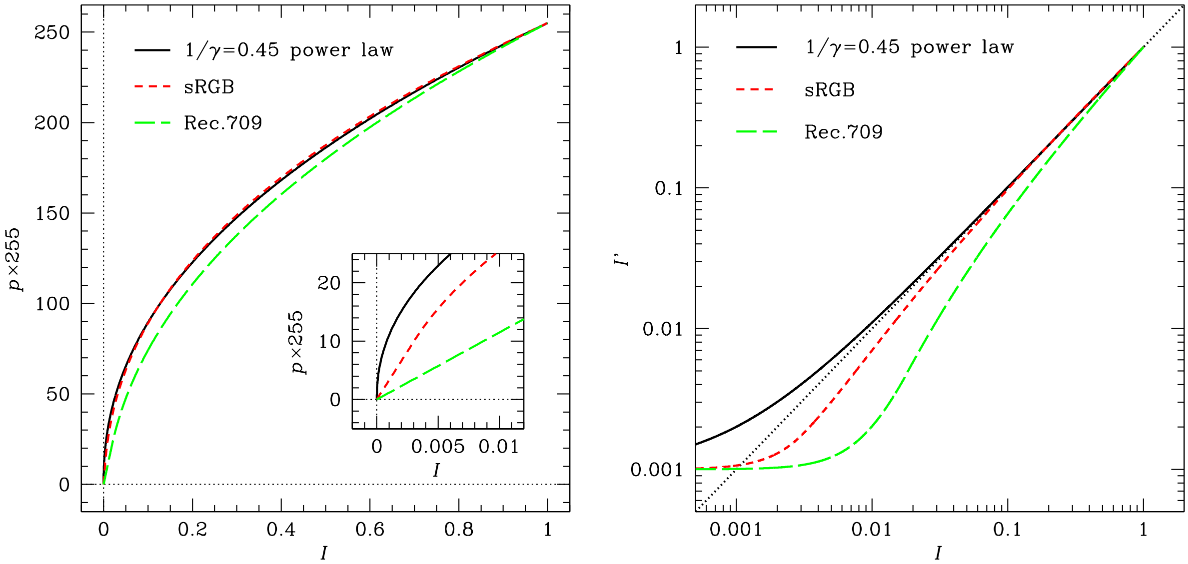


Figure 3: Three types of gamma compression performed by STIFF. *Left*: encoded pixel values (for an 8-bit image) vs relative original image intensity; *black curve (continuous)*: `GAMMA_TYPE POWER-LAW`, *red (short dashes)*: `GAMMA_TYPE SRGB`, *green (long dashes)*: `GAMMA_TYPE REC.709`. Note the important difference at low intensities (*insert*). *Right:* Comparison between the end-to-end transfer curves, assuming a 2.2 display gamma and a minimum black level at 0.1% of full white.

movies intended to be displayed in a totally dark environment (e.g. for the big screen). Note that when `GAMMA_TYPE` is set to `REC.709` the intensity mapping follows the Rec. BT.709 coding: black at 16 and reference white at 235 for an 8-bit output.

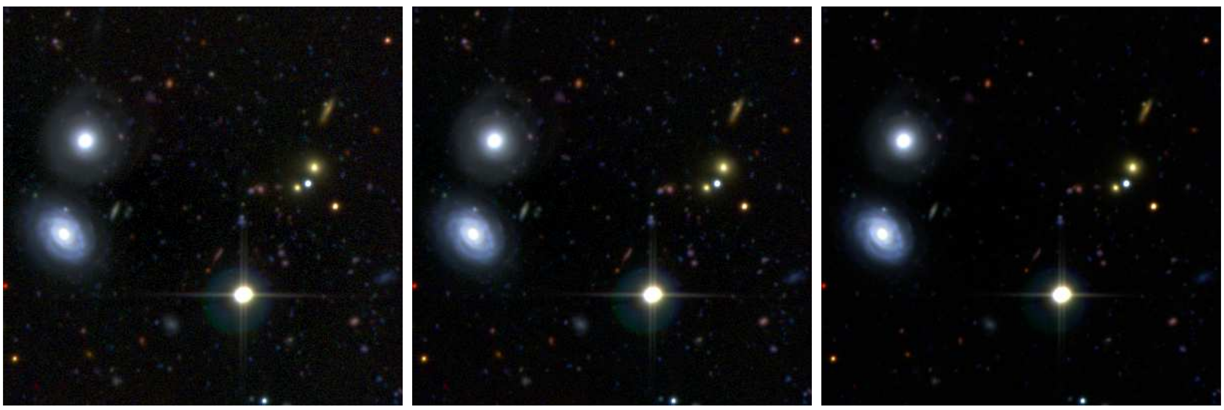


Figure 4: Deep sky colour images generated by STIFF using three different `GAMMA_TYPE`s. *From left to right:* `POWER-LAW`, `SRGB`, and `REC.709`.

6.2.2 Colour-friendly gamma corrections

In some cases, one may want to enhance or reduce the contribution of low surface brightnesses to the final image, without saturating brighter regions. For this a non-linear transformation has to be applied, and modifying the gamma correction is the easiest way to achieve this. Increasing or decreasing γ will compress or expand the contrast scale, respectively. Although this works fine with monochromatic images, it affects colours on multi-channel images. Fig. 5 shows using a simulation the effect of increasing γ to enhance the extended wings of a galaxy profile: here as no background light is present the consequence is only a desaturation of the colours. When several components of various colours are superimposed, significant shifts in colour can occur.

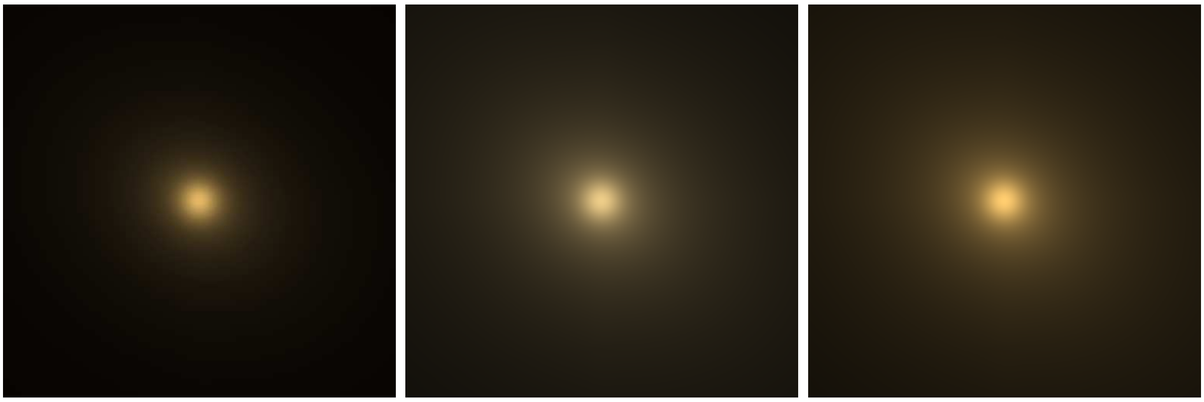


Figure 5: A simulated colour image of an early-type galaxy converted in TIFF by STIFF. *Left*: original gamma of 2.2. *Middle*: using a gamma of 3.3 affects the colour saturation. *Right*: applying a gamma correction factor of 1.5, to luminance only, preserves colour rendition.

As Lupton et al. (2004) remind us, such problems can be circumvented by applying the additional non-linear corrections only to the *luminance* part of the Red,Green,Blue (RGB) signal. We simply define[6] the luminance as

$$Y = \frac{R+G+B}{3}, \tag{3}$$

where R, G, and B are the red, green and blue component values at a given pixel before any non-linear transform is made to the signal. By applying a gamma correction factor γ_Y to the luminance only, one will store in the output TIFF file the r,g,b values

$$r \propto \left(\frac{R}{Y} Y^{1/\gamma_Y}\right)^{1/\gamma}, \tag{4}$$

$$g \propto \left(\frac{G}{Y} Y^{1/\gamma_Y}\right)^{1/\gamma}, \tag{5}$$

$$b \propto \left(\frac{B}{Y} Y^{1/\gamma_Y}\right)^{1/\gamma}. \tag{6}$$

In STIFF, γ_Y is represented by the configuration parameter `GAMMA_FAC`, which defaults to 1.0. By increasing `GAMMA_FAC`, one can enhance low-surface brightness features in the image without affecting the colours.

6.3 Controlling the image dynamic range

Despite the fact that, as we have seen, 8-bit images with a non-linear intensity scale can record surface brightnesses with a dynamic range as high as 100dB, it is generally preferable to restrict the intensity range in input by imposing a low and a high cut for pixel values. STIFF can set these cuts semi-automatically or manually for each channel (colour component). If one

[6]This definition is somewhat different from the one generally used in video when green is given much more weight.

chooses manual mode (MIN_TYPE MANUAL and MAX_TYPE MANUAL), the low $I_{\rm min}$ and high $I_{\rm max}$ cuts are directly set by the values of MIN_LEVEL and MAX_LEVEL, respectively. If MIN_TYPE and/or MAX_TYPE are set to QUANTILE, MIN_LEVEL and/or MAX_LEVEL are interpreted as quantiles (between 0.0 and 1.0) of the histogram of pixel values. This is the default for MAX_TYPE.

There is a third mode specific to MIN_TYPE called GREYLEVEL. In that mode, the lower cut $I_{\rm min}$ is automatically adjusted to have the sky background reach a given apparent grey level $p_{\rm grey}$ (specified by MIN_LEVEL) as a fraction of full white in the TIFF image. This is the default mode for MIN_TYPE, with a default MIN_LEVEL of $p_{\rm grey} = 0.001$ (a very dark grey). The gamma correction factor is taken into account:

$$p_{\rm grey} = \left(\frac{I_{\rm sky} - I_{\rm min}}{I_{\rm max} - I_{\rm min}}\right)^{1/\gamma_Y}. \tag{7}$$

Hence

$$I_{\rm min} = \frac{I_{\rm sky} - p_{\rm grey}^{\gamma_Y}\, I_{\rm max}}{1 - p_{\rm grey}^{\gamma_Y}}. \tag{8}$$

Note that the default $p_{\rm grey} = 0.001$ corresponds roughly to the decimal 8-bit pixel value 11 for a power-law gamma of 2.2, 3 for an sRGB gamma-correction, and only 1 for a Rec.709 gamma-correction.

The sky background intensity $I_{\rm sky}$ is estimated automatically from the input image(s) by STIFF when SKY_TYPE is set to AUTO, which is the default. Note that this estimation is rather crude (it is the median of the histogram of pixel values). Therefore it is sometimes better to switch to manual grey-level mode (SKY_TYPE MANUAL), for which the sky background level is directly read from SKY_LEVEL.

6.4 Creating colour images

STIFF can generate a composite RGB TIFF image from 3 co-aligned FITS images obtained in different channels. The 3 images must have the same number of pixels in each dimension. If the data are properly astrometered but not aligned, you might want to pass them through SWarp[7] first.

Without further processing, the colours obtained from images observed with broadband optical filters often look dull and disappointing. STIFF offers the possibility to increase the colour saturation of images, while maintaining their luminance and global white balance. To this aim we introduce an additional α parameter that linearly replaces each R, G, B input triplet with R', G', B' such that:

$$R' - G' = \alpha\,(R - G), \tag{9}$$
$$G' - B' = \alpha\,(G - B), \tag{10}$$
$$(R' + G' + B') = R + G + B, \tag{11}$$

from which we obtain

$$R' = Y + \alpha\,\frac{2R - G - B}{3}, \tag{12}$$
$$G' = Y + \alpha\,\frac{2G - R - B}{3}, \tag{13}$$
$$B' = Y + \alpha\,\frac{2B - R - G}{3}. \tag{14}$$

(15)

[7] http://astromatic.net/software/swarp

α is represented in STIFF by the `COLOUR_SAT` configuration parameter. `COLOUR_SAT` acts exactly like the saturation knob of a colour TV: if `COLOUR_SAT` is set to 0, the 3 channels are combined to form a black-and-white (greyscale) image; a `COLOUR_SAT` of 1.0 (the default) lets the input colour saturation unaffected, while a `COLOUR_SAT` above 1.0 exaggerates the colours. Fig. 6 shows the impact of various `COLOUR_SAT` settings on a deep-sky colour image. For images observed in contiguous channels, a `COLOUR_SAT` of 2.0 generally gives best results. To avoid funny-looking artifacts showing up with high `COLOUR_SAT`s for very bright objets, a per-channel clipping is applied to saturated pixels prior to exaggerating the colours (the saturation level is defined by the `SATUR_LEVEL` configuration parameter(s)). Note that this does not prevent pixels with values exceeding $F_{\rm max}$ in one channel or more to appear shifted in colour in the output image; for instance, the core of some bright, orange star may appear slightly yellowish.

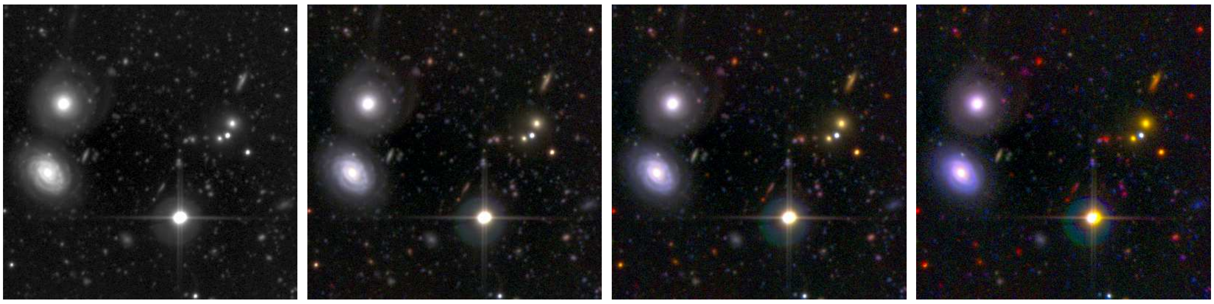


Figure 6: A three-channel deep-sky image converted to TIFF RGB by STIFF with 4 different values of `COLOUR_SAT`. From *left* to *right*: 0.0, 1.0, 2.0 and 4.0. Note the CCD saturation artifacts on top of bright stars.

6.5 Multi-resolution pyramids

Some astronomical wide-field images are both very large and extremely detailed, to the point that a regular display will not do justice to them, unless they get printed on posters with gigantic proportions. One way to improve the visual experience with a very detailed image is to allow the user to "navigate" quickly through it, as in web applications like Google maps$^{\rm TM}$. These web applications require images to be stored in tiles at multiple resolutions, like a "pyramid" of tiled images (Fig. 7).

The TIFF format allows the raster of pixels to be organised in tiles, and several images to be stored in a single file. This makes it the ideal format to store a large image at multiple resolutions. STIFF offers the possibility to generate automatically in the same TIFF file different versions of the image rebinned at octave intervals, and organised in tiles of arbitrary size. The position of each tile is indexed in the TIFF header, so that it can be accessed directly for display using the libTIFF function `TIFFReadTile()`. This feature is used by e.g., the IIPImage[8] server. Examples of results obtained on large astronomical images are shown on the Astr*O*matic web site[9].

To generate a pyramidal TIFF file, the `IMAGE_TYPE` configuration parameter must be set to `TIFF-PYRAMID`, or if `IMAGE_TYPE` is left to `AUTO`, the output file name must have the `ptif` extension. Tile size is 256 pixels by default. Although it is recommended to leave it at 256, it can be changed using the `TILE_SIZE` keyword. `PYRAMID_MINSIZE` sets the minimum dimension(s) for the smallest image in the pyramid.

[8] `http://iipimage.sourceforge.net`
[9] `http://astromatic.net/gallery`

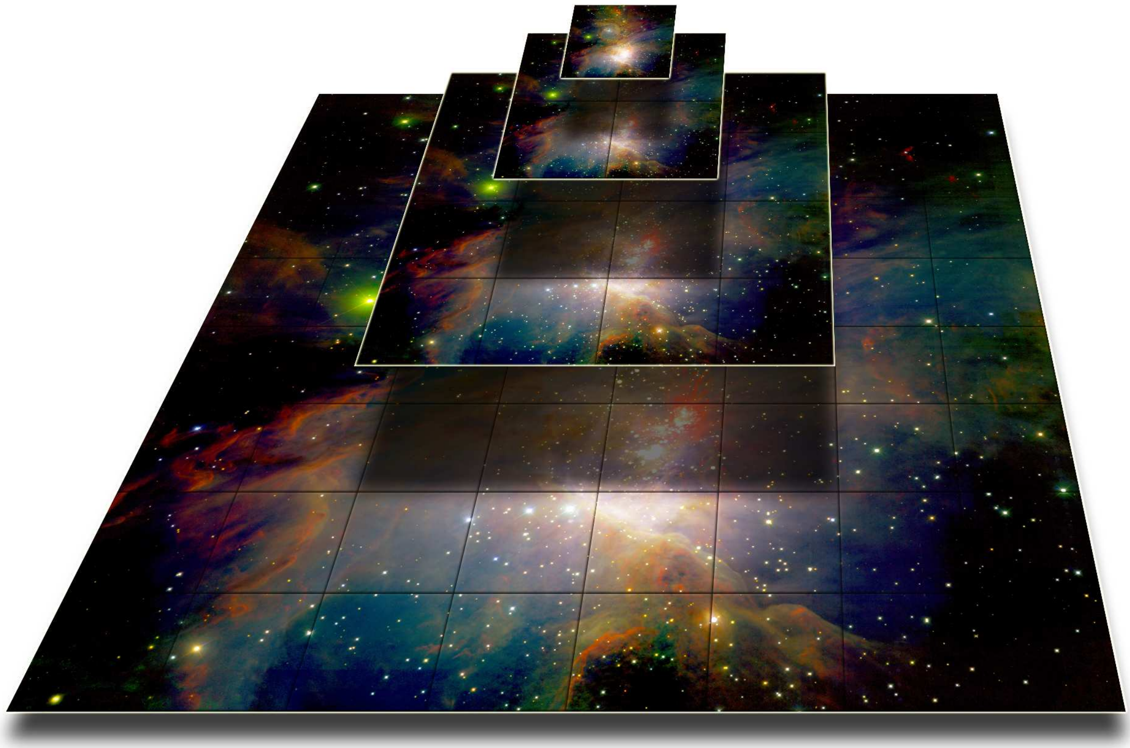


Figure 7: Depiction of a tiled, four-level image pyramid (Orion nebula observed with the WIRCAM instrument, picture credits CFHT/C.Marmo/Terapix).

6.6 Image compression

Versions 2.x and above of STIFF support image compression. In addition to no compression ("NONE"), the COMPRESSION_TYPE configuration parameter supports both lossless and lossy compression schemes. The supported lossless compression algorithms are LZW (the default), DEFLATE, and ADOBE-DEFLATE. All three offer rather similar performance, and the last two are merely there because of patent issues.

JPEG is a lossy compression scheme, identical to that found in .jpg images. JPEG compression leads to much smaller files than lossless algorithms, but it should be used sparingly in the astronomical context: deep-sky astronomical images generally exhibit a low signal-to-noise ratio per pixel. Noise, which is generally close to white, does not compress very well and eats up precious bits in the output code, which may lead to obvious artifacts. The level of JPEG compression can be adjusted with the COMPRESSION_QUALITY configuration parameter (in "quality" percentage); the higher the parameter, the more faithful to the original the result (at the price of a larger file size). The default for COMPRESSION_QUALITY is 90, which would be considered a fairly high value for "everyday" images, but which appears to be the right compromise for most astronomical images.

6.7 Very large images and memory handling

On 64-bit systems, STIFF can manage images with very large sizes (up to a few terapixels) both in input and output. The original TIFF format has itself a file-size limit of 2 to 4 Gbytes, which is roughly equivalent to a 1 Gpixel image. To handle larger output images, STIFF must rely on the BigTIFF modification to the TIFF format[10]. BigTIFF is implemented in version 4.0 of the LIBTIFF library, which is only available in beta form (but otherwise perfectly functional) as of January 2010. Note that the binary version of the STIFF package comes linked with LIBTIFF v4.0.

The `BIGTIFF_TYPE` configuration parameter sets the policy regarding BigTIFF usage — STIFF must have been linked with LIBTIFF version 4.0 or above. In `AUTO` mode (the default), STIFF automatically switches the output file to BigTIFF format if the output file is likely to exceed the 2GB limit. If `BIGTIFF_TYPE` is set to `ALWAYS`, TIFF output files always use the BigTIFF variant, regardless of file size. If `BIGTIFF_TYPE` is set to `NEVER`, output files stick to regular TIFF in all cases, which may lead to errors if they are too large.

STIFF is able to manipulate input images even if their size far exceeds the machine's memory budget, by doing "memory mapping" on temporary files. These temporary files are used whenever the amount of memory required for processing exceeds the value of the `MEM_MAX` configuration parameter (1024 Mbytes by default), and is less than `VMEM_MAX` value (1048576 Mbytes by default, that is, 1Tbyte). Temporary files are written to the directory path specified by the `VMEM_DIR` configuration parameter. The default path for `VMEM_DIR` is the current execution directory ("`.`"). One should make sure that the disk system holding the `VMEM_DIR` path is fast and local to the processing machine.

6.8 FITS header copy

Setting `COPY_HEADER` to `Y` instructs STIFF to copy the first FITS image header to the output TIFF file, under the `ImageDescription` tag. The `NAXIS`i, `CRPIX`i, `CDELT`i and `CD`i_j are automatically updated to reflect geometric changes made to the image with the `BINNING`, `FLIP_TYPE` or pyramidal tiling options.

Note that ImageMagick's `convert`[11] tool propagates the header information of TIFF files written by STIFF while converting to JPEG format. The resulting JPEG (colour!) images can be loaded in the Aladin[12] visualisation software (Fernique et al. 2009), which will automatically take advantage of the World Coordinate System information copied from the original FITS header, if present (Fig. 8).

6.9 XML output

An XML file providing a processing summary and various statistics in VOTable 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 `stiff.xml`. The XML file can be displayed with any recent web browser; the XSLT stylesheet installed together with STIFF will automatically translate it into a dynamic, user-friendly web-page (Fig. 9). 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[13].

[10] `http://www.awaresystems.be/imaging/tiff/bigtiff.html`
[11] `http://www.imagemagick.org`
[12] `http://aladin.u-strasbg.fr`
[13] See `http://www.astromatic.net/2009/10/05/understanding-astromatic-metadata-files` for more details and hints on how to use XML metadata files.

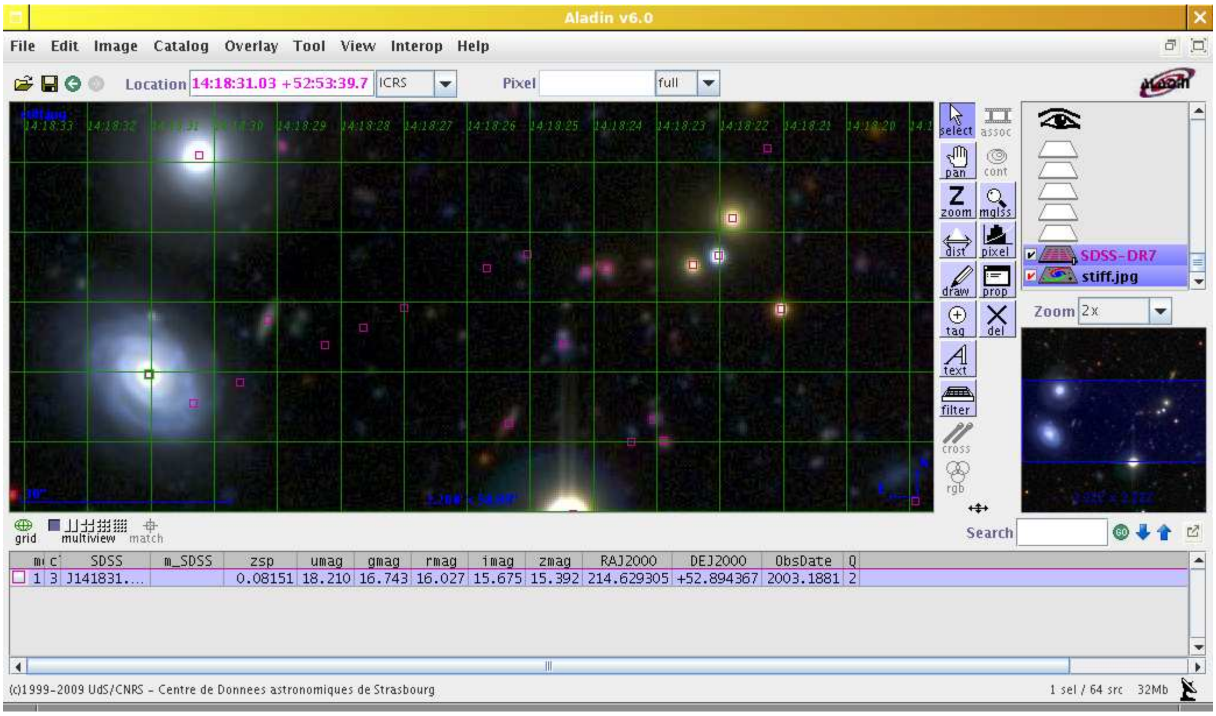


Figure 8: STIFF colour image created with the `COPY_HEADER` option, converted to JPEG with `convert` and displayed in ALADIN 6.0 with the SDSS catalogue in overlay.

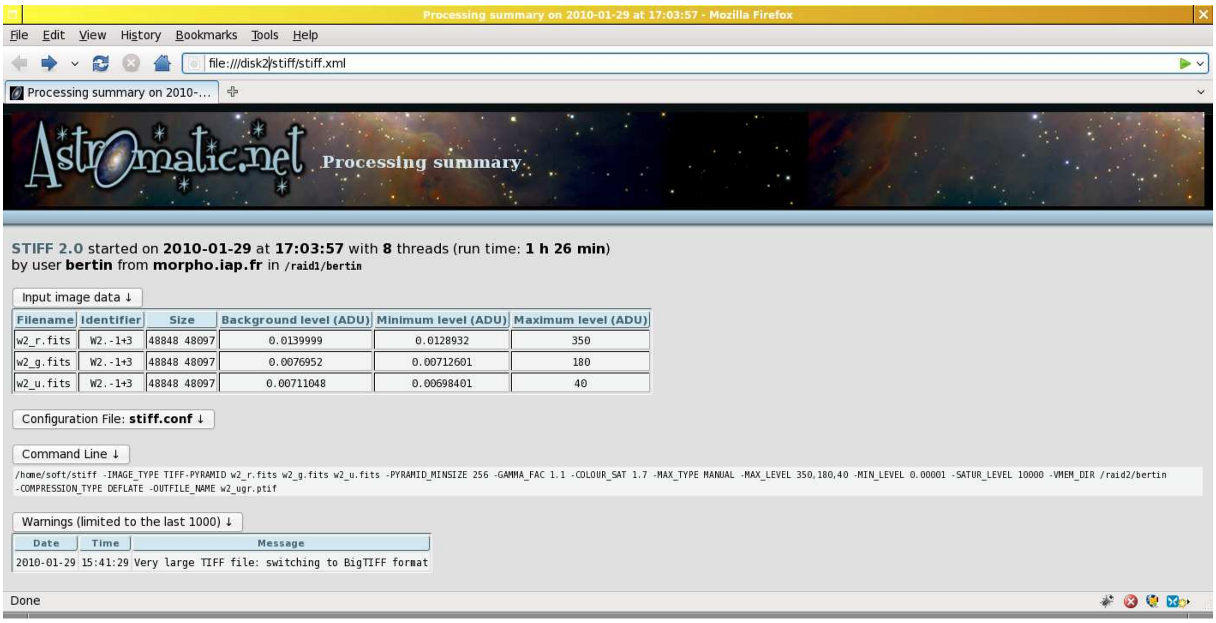


Figure 9: Example of a STIFF XML metadata file displayed in the Firefox browser using the provided XSLT stylesheet.

7 Hints for creating "realistic" deep-sky colour images

The main motivation for writing STIFF was to create vivid yet realistic images of the sky. In the following we will leave aside purely aesthetic considerations to insist on a few simple basic principles that might help us reach that goal.

The first problem is obviously that, even in the optical domain, it is difficult to actually *see* the colours of resolved deep-sky astronomical objects. The colours of many HII regions and planetary nebulae can be seen through the eyepiece of large telescopes, but for galaxies it is another story. Their surface brightnesses are one to two orders of magnitude lower and remain mostly in the scotopic regime of the human vision.

The second difficulty is that the spectral response curves of astronomical imagers rarely match those of our cones.

Fortunately, the colours of astronomical sources result essentially from a handful of well known physical phenomena: the slope of a thermal or non-thermal emission continuum, emission/absorption lines, reddening and diffusion by molecules, and cosmological redshift, that will guide us on our way to represent multi-channel data.

7.1 Ordering of channels

A first rule to make the physical interpretation of colours easier and consistent from a case to another will obviously be to let the red component represent the largest effective wavelengths and the blue one the shortest wavelengths.

7.2 Colour balance

Away from nebular and HII regions, the light from point-sources and galaxies is largely dominated by continuum emission in the near-optical domain. Despite the fact that the spectral responses of the observation filters do not match those of our cones, we might want at least to have the colours of regular, unreddened stars to fall roughly on the blackbody track in a CIE diagram (Fig. 10). That is, from red-orange to white-blue (in particular, green stars or galaxies are seldom observed!). Experience shows that this is quite easy to achieve for contiguous broadband filters in the optical/near-IR, after adjusting 2 of the 3 `MAX_LEVEL`s.

7.3 The sky background

Although this is not realistic, it is strongly advised to use a dark, neutral grey colour for the sky background (actually the real night sky, from the ground or in orbit around the earth, is redder than the sun in the visible, see Leinert et al. 1998). This is what automatically done with the `MIN_TYPE GREYLEVEL` option (the default). A dark grey sky does not affect too much the perception of object colours, and prevents the faintest objets from disappearing in a completely black background. The intensity of the dark grey is set by default to `MIN_LEVEL` 0.001, that is 0.1% of full white, a good value for a high contrast display screen. For print-outs, one may prefer a slightly higher value, since many printers have a tendancy to "bury" low surface brightnesses in dark backgrounds.

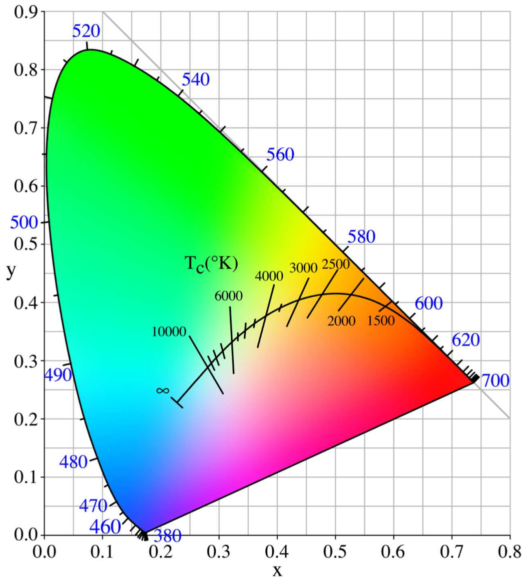


Figure 10: CIE chromaticity diagram (source: *Wikimedia Commons*). The dark line is the track followed by a blackbody with increasing temperature (from right to left).

8 Examples

Example 1 The following command was used to generate one of the colour image pyramids which can be seen at `http://astromatic.net/gallery`:

```
stiff -IMAGE_TYPE TIFF-PYRAMID d1_i.fits d1_r.fits d1_g.fits -GAMMA_FAC 1.1
-COLOUR_SAT 2.2 -MAX_TYPE MANUAL -MAX_LEVEL 550,350,160 -MIN_LEVEL 0.00001
-SATUR_LEVEL 10000 -COMPRESSION_TYPE DEFLATE -OUTFILE_NAME d1_gri.ptif
```

Example 2 The following Python script generates a small 64x64 pixel TIFF image of each detection at the coordinates read in columns 1 and 2 of the ASCII file whose filename is given as second argument:

```
#!/usr/bin/env python
import sys,re,os
if len(sys.argv) < 2:
        print "Syntax: "+ sys.argv[0] + "<image> <catalog>"
        sys.exit()
ima = sys.argv[1]
cat = sys.argv[2]
f=open(cat)
num = 1
for line in f.readlines():
        if line[0]!='#':
                col = [float(split) for split in re.split('\s+', line)[0:-1]]
                alpha = col[0]
                delta = col[1]
                os.system("swarp -VERBOSE_TYPE QUIET -RESAMPLE N -IMAGE_SIZE 64 \
                        -CENTER_TYPE MANUAL -CENTER %10f" \
```

```
                %alpha + ",%-10f" %delta + " " + ima)
        os.system("stiff -VERBOSE_TYPE QUIET coadd.fits \
                -OUTFILE_NAME obj_%06d.tif" %num)
        num += 1
f.close()
```

9 Troubleshooting

I have trouble to properly adjust the contrast and brightness of large mosaic images in `MIN_TYPE QUANTILE` mode.

Because of memory constraints, for large frames the quantile is not computed on all the pixels at once. The quantile algorithm in STIFF may be confused if, in addition, a large fraction of pixels is set to constant values (because of gaps, image margins or overscans). Currently the best solution is then to return to `MIN_TYPE MANUAL` mode.

10 Acknowledgements

Many thanks to the TERAPIX crew for early testing, to Henry Joy McCracken for proof-reading an early version of this documentation, to Jean Mouette at IAP for checks on his calibrated monitor, to Rodolfo Barbá for suggesting the header copy mechanism, and Emmanuel Pietriga for help with debugging the Mac OS X version

References

[1] Bertin E., 1999, *SExtractor 2.1, User's manual*, IAP

[2] Fernique P., Boch T., Bonnarel F., 2009, *Astronomical Data Analysis Software and Systems XVIII ASP Conference Series* **411**, 559

[3] Leinert C. et al. 1998, *A&ApS* **127**, 1

[4] Lupton R., et al. 2004, *PASP* **116**, 133

[5] Stokes M., Anderson M., Chandrasekar S., Motta R., 1996, `http://www.color.org/sRGB.xalter`