Keith's Image Stacker

Legal Stuff

Table of Contents

This program highly affordable **shareware**. For the endless hours of work I have put into creating this utility I ask that you please pay me **$15.00**, which automatically registers you for all future versions. In return I will place you on an extremely infrequent mailing list to be directly notified of version updates when they occur. I can be paid in two ways. One, by PayPal to the account *kwiley@cs.unm.edu* (for payments under $12.00) or *kwiley@keithwiley.com* (for payments over $12.00), or two, by personal check or cash. Since my mailing address may change from time to time, I will not put it here. Instead, email me if you are interested in paying by that method.

Thank you very much.

Keith Wiley
100915
kwiley@keithwiley.com
http://keithwiley.com
http://keithwiley.com/software/keithsImageStacker.shtml

Introduction

Keith's Image Stacker is an image processing utility that provides you with many sophisticated image enhancement tools, including:

- Translational and rotational alignment
- Darkframe subtraction and flatfield division
- DeBayering (demosaicing)
- Normalization
- Averaging and/or summing (collectively known as "stacking")
- Unsharp-masking
- Laplacian sharpening (which is also known by the more general term wavelet sharpening, although there are actually many wavelet-based image-sharpening algorithms, of which Laplacian sharpening is only one)
- Level adjustment
- Gaussian blur
- Wavelet shrinkage denoising
- Fourier Transform editing (for removal of cyclic or periodic artifacts such as electrical interference patterns)
- Star deringing and debloating
- Many other operations as well

The primary purpose of this program is to stack images.

This program is mainly oriented toward astrophotographers, who often take numerous shots of an object and then want to stack the shots to get a single final image that is superior to the original shots. In recent years webcams and video cameras have become popular tools for astrophotography, and as such, are capable of capturing hundreds of individual images of an object in a given night. The basic approach to processing this data is to average the set of images on a per-pixel basis. By doing so, the signal-to-noise ratio of the stacked image is greater than each of the individual images by a factor of the square root of the number of images stacked.

This documentation assumes you are an astrophotographer and is written as such. If you are not and you wish to use the program to stack nonastrophotos (or to use the various other tools for image processing), that's fine. Simply follow the documentation along and figure out how to use the program. There are sample images provided in the documentation directory which you can use as a tutorial if you wish. I offer instructions on each of the two common methods of astrophotography, as well as a command reference page. Take your pick:

- Introduction
- Dynamic Range and 32-bit to 8-bit Image Projection -- It is strongly recommended that you read this first
- About RAW Images -- Please read this before asking any questions with the word *RAW* in them
- Planetary Imaging
- Deep Sky Imaging

- Multithreading Options
- FAQ (Frequently Asked Questions) -- Look here first, look here last, whatever, just look here before complaining about the program
- Command Reference

Dynamic Range and 32-bit to 8-bit Image Projection

It is important to understand how dynamic range is handled by this program, for reasons stated below. Dynamic range in a discrete-valued function such as integer-valued pixels corresponds directly to the number of unique values that can be assumed by a particular sample (pixel). For the moment, ignore the fact that color pixels are composed of three independent values, traditionally a brightness value for red, green, and blue. Simply realize that the description given here applies to the red, green, and blue components of an image and its pixels with relatively independence.

So, a single pixel's brightness, say in the red channel, can assume one of a given number of integer values. The range of those values dictates the available dynamic range of the image. What does this mean? Simply put, the dynamic range represents the ratio between the brightest and darkest values that can be registered in an image, so increased dynamic range means an image can represent a wider variation of bright and dark without saturating to pure white at the light end, and emptying to pure black at the dark end. Most common image formats use 8 bits per pixel. They are often called 32-bit images because 8 bits are used for each of the red, green, and blue channels with a 8 more bits left over for reasons of computer efficiency that don't need to be elaborated on here (sometimes they are more accurately referred to as 24-bit images for this reason). The point is, even though they are called 32-bit or 24-bit images, there are actually only 8 bits of information available for the brightness of a pixel in each channel.

The dynamic range is represented as a log value of this range, specifically, *log(brightest/darkest)*. For 8 bits, or 256 possible values, this comes to *log(255/1) = 2.4*. This is woefully insufficient to represent many astronomical objects such as galaxies and nebulae, which span a much greater dynamic range than this.

Stacking images accomplishes two goals. The first, mentioned on the front page of the documentation, is that it reduces the noise in stacked image compared with one of the original images. The second is that stacking increases the dynamic range. For example, two images summed together, have a maximum pixel value of 510, with a dynamic range of 2.7.

This program faithfully stores the full dynamic range of a stack of 8-bit images. It does this by storing the stacked image using 32 bits per red, green, and blue component, or 96 bits total. This dynamic range would not risk saturation unless you stacked more than *2^24* (16,777,216) individual 8-bit images, not very likely (and then only if some pixel in every image -- the same pixel that is -- had a value of 255). Computers are incapable of producing images on their screens that use more than 8 bits per pixel. Similarly, virtually all known image formats store images at 8 bits per pixel. This means there is no way to truthfully show a stacked image on the screen or save it as a common image file. The image file problem can be solved by saving the stack in a less common image format that can accommodate increased bits per pixel. Two such formats are 16-bit TIFFs and 16 or 32-bit FITs.

To draw the stack on the screen, or to save the stack as an ordinary image file -- an option under the

File menu -- the program must project the 32-bit stack down into an 8-bit image. There is nothing complicated about this, but it is somewhat arbitrary. The reason is that a pure projection from 32 to 8 bits would make any realistic stack project into a pure black 8-bit image because even a realistic stacked image uses only a minute fraction of its entire 32-bit range (remember that the maximum number of possible images is 16,777,216, so a realistic stack of, say, 1000 images, uses only the very bottom of the dynamic range). For this reason, generally only a lower subrange of the 32-bits is projected into the 8-bits of the image. The question is, what should that lower subrange be? This is where things become arbitrary, and **it is important to understand this section of the documentation** because it will help you understand why **the program appears to automatically alter the brightness of the stack sometimes**.

Stacks are traditionally represented in one of two formats, an average or a sum. An average shows a stack with the same brightness as each of the individual images that contributed to the stack (by dividing the stack's accumulated values by the number of images stacked). A sum shows the stack with a brightness that increases linearly with the number of images stacked (by simply showing the stack's pure values, perhaps saturating to white in pixels that sum to a value greater than 255). However, the stack can in truth be represented by any arbitrary brightness that might be desired.

This program attempts at all times to maximize the use of the 8-bit dynamic range of the projected image when projecting the 32-bit stack into an 8-bit image. It does this by finding the single brightest pixel in the entire stacked image and calculating the necessary divisor to bring that value under 256. Since the brightest pixel will now be divided enough to fit into 8-bits (under 256), all other pixels in the image are guaranteed to fit as well (since they will only become an even lower value after being divided by the same divisor). This divisor changes periodically (usually upwards) throughout use of the program when certain operations may require it to do so. This change will be perceived as a dimming or darkening of the entire stacked image on the screen (since the pixels are being divided by a larger divisor). This should not concern you at the time of the operation that resulted in the dimming of the stack. After the operation has completed you can then use the Histogram/Level Adjustment interface to make the image bright again. The advantage of doing things this way is that the image will not be saturated by such operations that would otherwise max out the pixel values and produce saturated pixels in the image (saturating some pixels would lose valuable contrast information in regions of the image).

About RAW Images

The term *raw*, sometimes capitalized *RAW*, can be used in at least four fundamentally different ways, in addition to any number of minor distinctions. The four major usages are:

- **RAW 1** -- The vague concept of an image that hasn't been processed yet.
- **RAW 2** -- A form of low-level circuit-board modification that can be performed on some webcams.
- **RAW 3** -- A file format in which absolutely no meta-information is provided which would be necessary in order to properly interpret the file contents as an image.
- **RAW 4** -- A file format which stores the direct output of a digital CCD sensor without performing any alterations on the data.

Here follows a brief description of each of the four types...

RAW 1

Raw, usually written lowercase, is used throughout the documentation and throughout the astrophotography community to refer to a single unstacked unprocessed image that has been brought into the workspace. Such images have not, for example, undergone alignment, darkframe subtraction, flat field division, denoising, etc. Since this usage of raw refers to a general notion of an image and not to a specific file format, it makes no prescription on the incoming file format. Such images can originate from digital camera RAWs (type 4 below), TIFFs, JPEGs, or any other conceivable file format.

RAW 2

RAW refers to a low-level circuit board (soldering) modification on webcams. It was developed by Steve Chambers and permits the capture of images comparable to the RAW 4 format described below, but without all the proprietary corporate malarkey. This usage of the term RAW is briefly referenced in the documentation in the section on DeBayering.

RAW 3

RAW refers to a file format in which pixel values are laid out in an single contiguous array in the file with no meta-information (as would often be stored in a file header for example) and for which no agreed-upon convention or standard indicates the file's structure. For example, there is no explicit way to know the dimensions of the image, the bit depth used to record color values, the byte order (big vs. little endian) or whether the RGB channels are interlaced or separated. Such information must be provided at the time a RAW file is opened so that the file can be interpreted. Many image programs support this file format including Graphic Converter, Adobe Photoshop, and Keith's Image Stacker.

RAW 4

RAW refers to the proprietary file formats used to store CCD data on digital cameras without any processing, such as gamma correction, bit depth projection, or JPEG compression. Every camera manfacturer (and perhaps every camera model) uses a different RAW file format and those formats are private corporate entitites, and thus are relatively inaccessible to most programmers, short of reverse engineering. Professional software packages, such as Photoshop, are able to support RAW files because companies like Adobe pay licensing fees to the camera companies to use their file formats. As such, these kinds of RAW files are inaccessible by Keith's Image Stacker.

Planetary Imaging

To see an example of planetary imaging in action, look at the About Box after launching the program. The following is a list of the necessary steps for processing planetary images. I will go through each one in order, from start to completion:

- Bringing the raw frames into the workspace
- DeBayering the frames (if required)
- Optionally resampling the frames (usually unncessary)
- Translationally and rotationally aligning the frames
- Optionally cropping the frames (usually unncessary, since the Open Movie dialog lets you do this)
- Sorting and selecting the best frames for stacking
- Optionally saving the workspace for later continuation (optional, based on whether you need to stop at any time)
- Stacking the frames
- Optionally blurring the stack (usually unnecessary)
- Sharpening the stack
- Level adusting the stack
- Denoising the stack
- Removing electrical interference patterns from the stack
- RGB channel alignment on the stack

Bringing the raw frames into the workspace

Keith's Image Stacker supports most common image formats and the Quicktime movie format. To bring separate images or a movie of images into the workspace, simply choose the appropriate menu option under the File menu..

It is important to note that the dimensions of the image will always be equal to the first image brought into the workspace. If you subsequently bring in other images that have other dimensions, they will be squeezed to fit into the present workspace dimensions (which is almost always an unacceptable situation, so don't do it in the first place).

NOTE: You cannot stack images that are greater than 4091 pixels wide. This problem is becoming increasingly prevelant as people use cameras with ever-increasing resolutions. Bottom line, such cameras are much more modern than Image Stacker or its capibilities can handle. You can open larger images into the workspace, operate on images in the workspace, and even save images from the workspace, but you cannot produce a stack greater than 4091 pixels wide. This is a limitation of Apple's Quickdraw PixMap structure and is entirely beyond my control. The only way to fix this would be to rewrite the program using more modern graphics APIs...which I am unlikely to do. The solution is to crop or resample your images before stacking. You can do this either in Image Stacker or in another program before bringing the images into Image Stacker.

When opening individual image files you can select multiple files and bring them all in at once. When opening a Quicktime movie you have several options to help you guide the file-opening process. One of the most important tools at this stage is the ability to crop the movie frames as they are brought in on the fly, rather than cropping them after they have already been brought into the workspace. The advantage of this is that the individual frames never have to take up more memory than their initially cropped size, thereby allowing you to bring in a much larger number of frames. You can specify whether or not the cropping rectangle should align to each frame in the movie as the frames are read. This is a good idea since it prevents cropping of the primary object in the image should the object drift outside your originally specified rect.

There are some other features of the Open Movie Dialog as well. You can choose to only see every Nth frame when you clip on Accept all Remaining where N is a number you can set. By choosing not to see all frames, the opening process will go a little more quickly.

NOTE: The workspace is visibly represented by a window titled *Frames*. There is no window titled, for example, *Workspace*. Sorry for any confusion on the matter.

DeBayering the frames

It is crucial that this step be performed prior to resampling, and not after, assuming you intend to resample the frames. Performing these two operations in the opposite order will not work properly at all.

This step is only necessary if the images were captured with a Steve-Chambers RAW-modded webcam that captures the Bayer matrix without deBayering it. If that is the case, you must deBayer the frames before any other processing can seriously be performed. If you have no idea what DeBayering is, or if you have no specific reason to suspect your images are Bayered, you can save yourself a fair amount of headache by skipping this section completely. Bottom line, if you don't what this section is talking about, then you don't need it.

There are four possible Bayer patterns, and only one pattern is right for each frame. However, different frames are likely to require different patterns. The best way to handle this situation is to manually deBayer the reference frame only, by moving the mouse over the first clip and tapping the **'B'** key. A dialog will be presented in which you can choose a Bayer pattern and a deBayer algorithm. Trial and error on the four patterns should produce the correct deBayered image. You can then deBayer all the other frames in the working space automatically by selecting all, tapping the **'B'** key, and then selecting the option in the deBayer dialog that specifies that the deBayer pattern should automatically be matched to the reference frame.

There are also a number of deBayering algorithms available. They are sorted from top to bottom in order of increasing complexity and required processing time, but also increasing quality of the final result. It is highly recommend that you use the linear interpolation method in all cases. It isn't too much slower than the other methods and will produce much higher quality results.

Optionally resampling the frames

If you're just starting out with astrophotography or are just trying to figure this program out for the first time, then skip this step entirely.

You may wish to resample the frames prior to stacking. This can be particularly useful if you resample to a larger size since you can then align to a finer resolution than at the original size. If you do this, it is probably a good idea to resample the final stack back to the original size after stacking the frames (which isn't presently supported by this program, you can use any other image-processing program to do that though). To resample the frames simply pick a resampling option from the Resample menu under the Operations menu.

Aligning the frames

NOTE: I realize that this section is quite long. However, you will be able to edit your images and get them stacked much more easily if you master the interface for aligning frames. Please read this section carefully...then reread it.

NOTE: Assuming you open your images from a Quicktime Movie (as opposed to a set of individual image files), and assuming that you specify a clipping rect in the Open Movie dialog (any rect at all even if it spans the entire raw dimensions of the frame), translational alignment is almost unnecessary as a processing step anymore, because the process of bringing in the clips from the movie file automatically aligns them extremely well. This process uses centroid alignment however, and operates prior to any deBayering that may subsequently be required on the clips. Therefore, you may wish to perform further alignment, but only minor alignment changes should be necessary. In such a case, the **'3'** key method of alignment will probably be fastest and produce a good final alignment. Furthermore, rotational alignment may still be required.

The first thing you must do besides resampling is align the frames. You must become familiar with the Clips Window and how it works if you are going to quickly navigate and manipulate a large number of clips, so play around with it for a while. Notice that as you move the mouse, the clip under the mouse is always highlighted with a thick yellow border. Also notice that the highlighted frame is always shown in the Frame Inspector Window. Realize, although this is obvious, that you don't have to use the mouse button to highlight a frame. Simply moving the mouse is enough. It is often useful to quickly move the mouse around over the clips to get a sense for the general quality or alignment of the frames.

To align the frames, you must first select a reference frame to which all other frames will be aligned. This should ideally be the highest quality frame in the set, but realistically, any good frame will work. The alignment will suffer if the reference frame is of poor quality so take some time in selecting the reference frame and make careful comparisons between the top comptetitors before moving on to the next step.

When you find a frame you like position the mouse over it and hit the **'R'** key. **'R'** triggers the Set-Reference-Frame operation and toggles the clip under the mouse with the very first clip in the upper left corner of the Clips Window. Each time you hit **'R'** these two clips trade places. The first clip is always referred to as the Reference Frame for the purposes of terminology. Many operations use the reference frame, not just alignment.

It is time to learn another Clips Window command. While moving the mouse around in the Clips Window, pause on a given clip for a moment. As stated above, that clip's frame is shown in the Frame Inspector Window. Now hold down the **'space'** key. The Frame Inspector Window instantly changes to show the Reference Frame. As long as the **'space'** key is down, the Frame Inspector Window will remain locked to the Reference Frame. When you release **'space'** the Frame Inspector Window will revert to its default behavior of showing the clip under the mouse. This is a very useful tool. There are numerous occasions when it is desirable to compare two frames in precise detail. This is the most feasible way to accomplish this goal. For example, say you are searching for a good reference frame to use for alignment. Rather than look at several frames and try to remember how they compare, simply pick one and set it as the Reference Frame. Then go to one of the other potential good frames and flip back and forth quickly with the **'space'** key. If the new clip seems better, set it as the Reference Frame. Then continue to explore the clips for another good frame. After doing this a couple times you will have found a very good clip.

Okay, you've picked a good Reference Frame. Now you must select it. Do this by shift-clicking on the Reference Frame. The clip will now appear with a thin yellow border. The reason you must do this is that you need the Reference Frame to be visible in the Frame Inspector Window after you move the mouse outside of the Clips Window. Ordinarily, the Frame Inspector Window is blank when this happens (because there is no way to guess which frame should be displayed). However, if exactly one frame is selected then that frame will be shown persistantly in the Frame Inspector Window except when the mouse is over the Clips Window.

Move to the Frame Inspector Window and click-drag a rectangle around a useful feature. Since this example demonstrates how to work on a planet, the most obvious feature is the bounding rectangle around the planet's perimeter. However, bear in mind that larger Operations Bounds Rectangles (as this rectangle is always called) will require longer periods of time for various operations to complete. In reference to alignment, this rectangle is defining an area which will either be compared by difference with each frame or by centroid location with each frame. In the case of difference alignment, each frame will be aligned to the Reference Frame by finding the translation or rotation that results in the minimum difference between that frame and the Reference Frame within the Operations Bounds Rectangle.

It is time to actually align the frames. Let's do a translational alignment first since this is the more common case (rotation being the alternative). Select all of the frames. Now press and *hold down* just one of the **'1'**, **'2'**, **'3'**, **'4'**, or **'5'** keys. While holding down the number key you have chosen, click on any selected clip. Since they are all selected, you can click anywhere in the Clips Windows (except in the lower right corner which will potentially be blank). This usage is common to most operations that act on clips. You perform the operation by clicking on any clip that is presently selected. The operation will then occur on all the selected clips.

Which number should you use? Keys **'1'**, **'2'**, and **'3'** initiate translational difference alignments. The **'1'** key should be used for really bad alignments, 15 pixels or more. The **'2'** key should be used for alignments between 5 and 15 pixels. The **'3'** key should be used only for very tiny alignments, up to 5 pixels. All three methods will align in all situations. However, each method will perform more quickly than the other two methods within its specified range.

The **'4'** key initiates translational centroid alignment. If you use this method, you should always follow it with a minor difference alignment with the **'3'** key, since it is not as precise as the difference alignments. However, the centroid alignment is generally faster than the difference alignments when the alignment error is large and is therefore recommended as a first-pass approach to translational alignment.

The **'5'** key initiates translational cross-correlation alignment which is calculated using the Fourier Transform of the frame. Although highly toted in the literature, it has some drawbacks. It performs more slowly than centroid alignment, but probably more precisely than centroid alignment. It is unclear whether it runs faster or slower than difference alignment and whether it performs better than difference alignment. Finally, it can get thoroughly confused sometimes, which can result in completely incorrect results.

It is also worth noting that you can choose whether or not to translationally align the red, green, and blue channels separately. The "difference" method of aligning frames aligns, as one would expect, by matching the difference between the frame being aligned and the Reference Frame. The default behavior is for this difference to be calculated using the average value of a given pixel's red, green, and blue components in the two images. By selecting the Align RGB Channels Separately menu option under the Operations Menu, you can align the red, green, and blue channels of an image separately, so that rather than having one translation for the image, you have three. This can be particularly useful for objects imaged low on the horizon where the atmosphere often distorts the blue channel out of sync with the red channel.

In order to align the RGB channels separately you simply check the corresponding menu item and then align in a normal fashion by holding down the **'1'**, **'2'**, or **'3'** key to specify an auto-alignment search size and clicking on a clip.

Rotational alignment works in a similar fashion to the difference-based methods initiated by **'1'**, **'2'**, or **'3'**. To perform a rotational alignment hold down one of the **'6'**, **'7'**, or **'8'** keys. Where as for translational alignment larger translational errors are more efficiently aligned with the **'1'** key, likewise for rotational alignment, larger rotational errors are more efficiently aligned with the **'6'** key. The **'6'** is most efficient for rotations that are off by more than four degrees, the **'7'** is most efficient in the two to four degree range, and the **'8'** key is most efficient below two degrees.

NOTE: You can manually nudge any clip's rotation or translation. While no clips are selected move the mouse over a single clip, or alternatively make a selection of clips by shift-clicking or shift-click-dragging over the clips in the Clips Window. Now use the arrows keys to nudge the frame's translation or the entire selection of frames. By holding down shift, option, and/or control while using the arrow keys you can cause larger nudges. Each of the three shift/option/control keys is equivalent in this usage. The trick is that the more of these three keys you hold down, the larger a nudge you get, so you have four possible nudge sizes, none of the shift/option/control keys, one of them, two of them, or all three. To manually nudge rotation, use **'z'**, **'x'** to rotate by .36 degrees, and **'Z'**, and **'X'** to rotate by 3.6 degrees.

It is often helpful to manually nudge frames a little bit before autoaligning. This is especially useful if, after autoaligning, you realize the autoalignment made mistakes with some of the frames in the selection. In such a case, manually nudge the frames into a fast and loose alignment and then throw the autoalignment at them to get a precise alignment.

Also be aware that you can manually nudge the red, green, and blue channels separately. This is similar to performing autoalignment on the three channels separately. To manually nudge the channels separately, start by checking the Align RGB Channels Separately menu option under the Operations Menu. Then, while holding down the **'1'**, **'2'**, or **'3'** key, tap or hold down an arrow key (be aware that this won't work in larger nudges with the shift, option, or control keys. You must nudge separate channels in single pixel increments by only holding down a number key). Only one channel will be nudged, corresponding to the numerical key being held down, **'1'** for red, **'2'** for green, or **'3'** for blue. This can be quite confusing since the same number keys can specify two entirely different things depending on the situation. When clicking with the mouse they specify auto-alignment search sizes. When using the arrow keys they specify a single channel. Sorry about this. I am working on a new interface for a future version that will alleviate this problem.

The last thing that needs to be commented on is the difficulty of performing both translational and rotational alignment on a single frame. This is an inherently difficult task. The best approach is to go back and forth between the two, first translationally aligning and then rotationally aligning. In many cases the auto-alignment process will seem to get stuck, meaning that autoalignment locks into an alignment that it can't escape from (it believes is ideal) even though you can easily tell (with the **'space'** key) that the alignment is suboptimal. This occurs because of the conflict between translation and rotation. In such a case, translation has locked such that it cannot find a better translation without first rotating, but rotation suffers the same problem in with respect to translation. The solution is to manually nudge the rotational alignment and then resume auto-aligning starting with translational alignment again. There are more intelligent algorithms that could do a better job of this but I have not had a chance to design or implement them yet. Sorry.

Optionally cropping the frames

You may wish to crop the frames after they have been aligned. This is never a requirement, but it can free up more of the computer's memory so that subsequent operations go faster. To crop, select exactly one frame (I always select the reference frame for this kind of work). Then move to the Frame Inspector Window and click/drag a rectangle. Then simply pick the Crop menu command under the Operations menu.

Optionally normalizing the frames

Sometimes the frames will vary in brightness. One common cause of this is having thin cloudcover blow across the planet while imaging. Some of the sorting methods will benefit from normalizing the frames prior to sorting. Normalization makes all the images the same brightness. Normalization requires an Operations Bounds Rectangle, so first select one frame, move to the Frame Inspector Window, and click/drag a rectangle around a feature such as a planet or a moon. Then, to normalize an individual frame, deselect all, and hit the **'N'** key while the mouse is over that frame's clip. To normalize several frames, make a selection by shift-click/dragging over the clips and then hit the **'N'** key.

There are two ways to normalize. You can normalize the brightest pixel in the Operations Bounds Rect or the average brightness within the Operation Bounds Rectangle. The second method works better but make sure you don't saturate the frames or you will be throwing away useful information.

Sorting and selecting the best frames for stacking

It is rarely advantageous to stack the entire set of frames. Instead, you usually want to select the good frames, and exclude the bad frames. Exactly what it means for a frame to be good or bad can be a little difficult to specify. Therefore, there are numerous possible methods by which to sort and include/exclude frames from the stack.

The simplest and usually the most accurate method is "manual frame selection", whereby you personally inspect each frame and rule it accepted or rejected. To do this, start passing the mouse over the Clips Window. All frames are included in the stack by default. To reject a particular frame, click on it with the mouse. It will be Xed out in red in the Clips Window and a small red X will appear in the Frame Inspector Window whenever that frame is shown in the Frame Inspector Window. With a little practice you can move through a large number of frames rather quickly.

A hint about manual frame selection: You should have already selected a good frame for the reference frame since you did this to align. Therefore, it is easy to compare the rest of the frames to the reference frame by using the space-bar command mentioned above. As you inspect the frames toggle back and forth quickly to the reference frame to visually appreciate the difference between the reference frame and the frame you are inspecting at any given moment. This will give you a better indication as to whether to include the frame.

Sometimes you have hundreds of frames and it will simply take too much effort to manually inspect all the frames. In that case, it is best to sort the frames by some measure of quality and then only include some fraction of the frames at the beginning or end of the list. Each of the sorting methods requires that an Operations Bounds Rectangle be defined. Select only one frame (I usually use the reference frame), move to the Frame Inspector Window, and click/drag a rectangle around the feature you want to use for sorting.

Each sorting method works best on a specific kind of feature. Details about this are provided below.

Be aware that some sorting methods require a rectangle that is an exact power of 2. To get such a rectangle, pick the size of the rectangle from the Power^2 Dimensions menu under the Operations menu and then control/click/drag in the Frame Inspector Window. If the dimensions of the image are too small for a particular power^2 rectangle, no rectangle will be visible in the Frame Inspector Window.

After an Operations Bounds Rectangle has been defined, pick the sorting method you want from the Sort menu under the Operations menu. Some of the sorting methods will instantly sort the frames when you do this. Others will bring up a dialog box where you can control how the frames are sorted and which fraction of the frames are automatically included in the stack.

Details of the Sorting Routines

- **by difference (from reference frame)**
  This method sorts the frames by their RGB pixel by pixel difference from the reference frame (only within the Operation Bounds Rect). If your frames vary in brightness this method may not work very well. It might be work normalizing the frames first and then sorting them.
- **by value**
  This method sorts the frames by the degree of value (the brightness).
- **by value range**
  This method sorts the frames by the range in value (brightness) between the single darkest and brightest pixels. I have used this by defining a very small rectangle around a moon's shadow as it transits across Jupiter. Sharper images tend to have a darker shadow for the moon and are thus selectable using this method.
- **by Fourier Transform** Thise method sorts the frames by the ratio of the power within a frequency range to the power over the entire frequency range. It is best used if the frequency range you specify excludes the lowest frequencies. Frames with a higher ratio of high frequencies are generally sharper. When you use this method you will be presented with a dialog. In this dialog you can see the average power spectrum of all the frames on the left and the scores of all the frames, sorted in decreasing order, on the right. You can manually adjust the cutoff frequency on the left. The default is 0, which means that initially the images are not sorted at all. You must move the slider to the right before the frames become sorted. Generally values less than 10 or even less than 5 work best. The slider on the right lets you automatically specify how many frames should be included in the stack.

  Also note that the frames in the Clips Window are automatically sorted as you adjust the cutoff frequency and you can inspect the clips both by moving the mouse over the Clips Window and by holding down the space bar to compare a given frame to the reference frame.

Optionally saving the workspace for later continuation

At any point in time you may save the entire workspace, which consists of all the frames and all of their attributes (alignment, dark-frame subtraction, etc), and the dark frame and flat field frame if they are present. Saving the workspace does not save any work performed on a stack, such as sharpening or level adjustment however. In fact, saving the workspace completely discards the stack if you have generated one already.

To save the workspace at any time, simply choose the appropriate menu item under the File menu. First, a dialog box will be presented to you telling you how much disk space will be required to save the workspace. You may cancel at this time if you realize you don't have enough disk space. Otherwise, you can choose whether or not to attempt lossless compression and whether or not to save nonstack-included frames (red X) and then may proceed to save the workspace file.

The lossless compression scheme is a little unpredictable. Sometimes it will actually produce a workspace file that is larger than you would otherwise get if you don't attempt to compress the data at all. Fortunately, the dialog box shows you how large the file will be in both cases so you can choose the smaller one. Alternatively, you may opt for the uncompressed method because it is generally a slight bit faster to save and open even if it takes up more space on the harddrive.

Likewise, you can open a workspace file at any point in time. Doing so will completely discard any present workspace, so make sure this is what you really want to do.

Stacking the frames

After you have sorted or otherwise included and excluded frames from the stack it is time to actually produce a stack of the frames that are to be included in the stack. To stack the included frames, simply choose the Sum/Stack all included Frames command from the Operations menu. It is important to understand how the SumStack works in order to properly manipulate level adjustment controls later on. Each individual frame is 8 bits deep per red, green, and blue channel. This is not the case for the SumStack which is 32 bits deep per channel. This means you can theoretically stack 2 to the 24th frames before saturating the available dynamic range.

The stack that is produced will always be a pure average of the images. This is what you usually want in the case of planetary processing.

NOTE: Please see the note above in the section ***Bringing the raw frames into the workspace*** about the 4091 maximum image width limitation on stacks.

Optionally blurring the stack

This is usually unncessary, but sometimes the stack will have patterned defects in it, such as thin parallel lines. A small amount of blurring can occasionally fix these kinds of problems up. Simply choose the appropriate command under the Operations menu.

Sharpening the stack

It is important to sharpen an image before level adjusting it and not vs/va. Sharpening adds information to the original image, making parts of it brighter. If you were to adjust the levels first, bringing up the brightness of the image in any way at all, you would limit the degree to which you could subsequently sharpen the image before you hit the saturation limit, so always sharpen first, and then adjust the levels.

There are two sharpening methods, Unsharp Masking and Laplacian Sharpening. I am not going to explain how they work in a deep way here. Instead I simply provide a quick description.

Unsharp Mask

You can specify the radius of blur and the intensity of the mask. The radius determines at what scale you sharpen the image, smaller radii specify fine details. The mask intensity determines how strongly the image is sharpened at the scale specfied by the radius.

Laplacian Sharpening

Laplacian Sharpening takes some practice. When you select this method the first thing that will happen is that six different masks will be generated by blurring the image at six different radii. You will then get the Laplacian Sharpening dialog in which you can adjust the mask intensities at each level individually. The masks correspond to different frequency components of the image. The sliders for the mask intensities are sorted from top to bottom in order of decreasing frequency, so the sliders at the top control masks corresponding to high frequencies (fine details), and the sliders at the bottom control masks corresponding to low frequencies (large patterns in the image).

There are checkboxes next to each slider that can be used to pin the slider to the value 0 or 1 when selected. The checkbox for 1 is useful to toggable back and forth to get a feel for what effect a particular slider is having on the image. The checkbox for 0 is useful for completely removing a particular mask's information from the resulting image, say to remove a noisy mask.

Level adusting the stack

One of the simplest steps is level adjustment. Choose the Show Histograms/Alter Levels command under the Operations menu. A dialog box will appear which allow you to alter low and high cutoffs and gamma for each channel. Note that by click/dragging in the small thumbnail image inside the dialog box you can define what area used for calculating the histograms that are shown. Since this rectangle effects the histograms and the histograms are used by the Auto Levels button, this rectangle also has an effect on how the Auto Levels button operates.

Denoising the stack

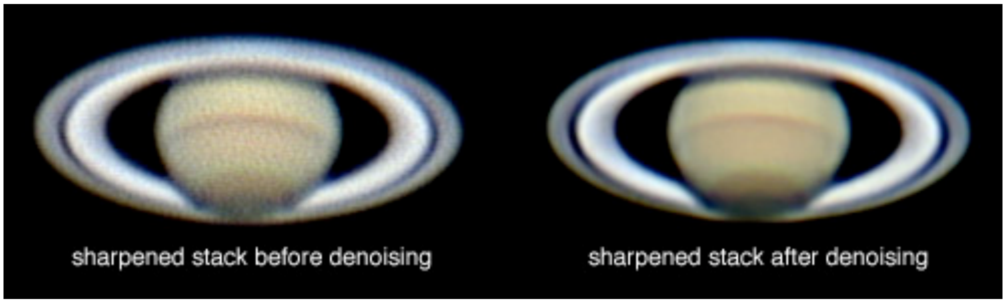
You may want to attempt to remove some of the noise from the final image. This can be done with the

Wavelet/Shrinkage Denoise command under the Operations menu. The dialog box will let you specify a preview area and a set of wavelet coefficient thresholds for wavelet shrinkage using a steerable pyramid with four oriented filters. Don't worry about what that means too much. Just select a preview area that bounds an interesting part of the image and set thresholds that you feel best reduce the noise without damaging the image too much. Much like the Laplacian Sharpening Interface, the sliders nearer the top of the window adjust high frequencies and the sliders nearer the bottom adjust low frequencies. When you click Okay, the thresholds chosen will be applied to the entire image.

Also note that as you change the shape and size of the preview area the cost of generating preview is displayed in the dialog. As the preview area becomes larger, lower frequency sliders will be enabled. All of the sliders' corresponding thresholds will be applied to the image with the OK button is pressed, but only those which are enabled will have an effect on the preview. In addition, if a preview can be generated in under a half second, live previewing will be enabled, such that the preview is updated on the fly as a theshold slider is dragged or as a threshld slider is adjusted with the arrow keys.

Incidently, wavelet shrinkage denoising tends to reduce the constrast of an image, so you may want to increase the constrast after performing this operation.

The following image demonstrates the powerful effect of wavelet shrinkage denoising.


sharpened stack before denoising
sharpened stack after denoising

Removing electrical interference patterns from the stack

This step is almost never required in planetary imaging, but you should realize that it is available if you need it. Further description is offered in the instructions for deep sky processing.

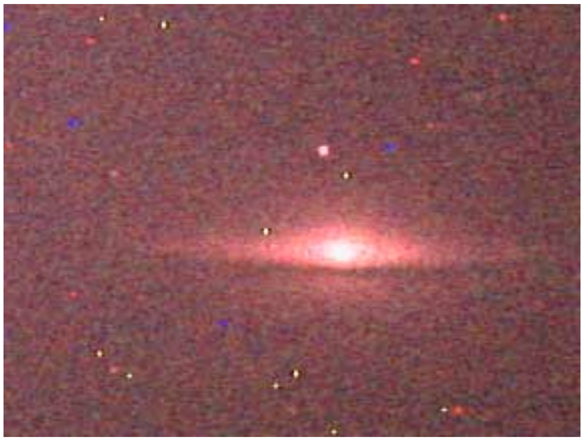
RGB channel alignment on the stack

You can manually align the red, green, and blue channels of the final stack if you wish. This can be useful if the channels are out of alignment, which often occurs when imaging objects low on the horizon, where the atmosphere distorts the blue channel. Make sure the FrameStack window is in the foreground. Then, while holding down the **'1'**, **'2'**, or **'3'** keys, tap or hold down an arrow key. Each number key corresponds to the red, green, or blue channel obviously.

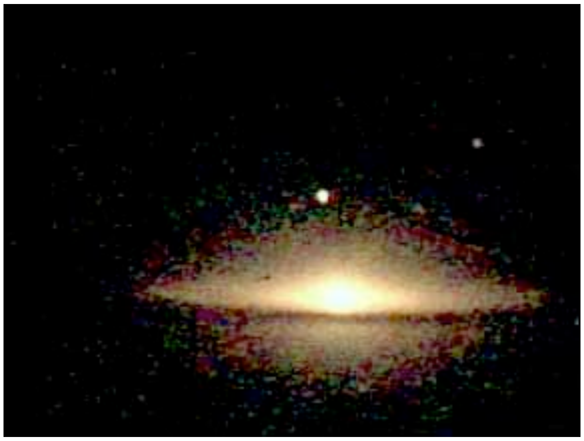
That's it. The stack is complete. You can save the stacked image now and you're done.

Deep Sky Imaging

The following images show an example of what can be accomplished by stacking deep sky images. This example was made from the exact set of images available in the Tutorial Images folder of the documentation, so you should be able to replicate or improve on the results yourself. If the raw images were not compressed highly or the stack consisted of more frames, the results would be even more impressive. You can see a much better version of this stack on my website which uses uncompressed raw frames and many more raw frames than those provided here.:



single original frame



stack of 9 frames (factor of 3 signal to noise ratio improvement over a single frame) after level adjustment

Many of the steps in processing deep sky images are similar to processing planetary images. Rather than repeat the same information, I will make reference to the planetary imaging documentation in the relevant places.

The following is a list of the necessary steps for processing deep sky images. I will go through each one in order, from start to completion:

- Bringing the raw frames into the workspace
- DeBayering the frames (if required)
- Optionally subtracting a dark frame from the frames
- Optionally dividing a flat field from the frames
- Optionally resampling the frames (usually unnecessary)
- Aligning the frames
- Optionally cropping the frames
- Optionally normalizing the frames (usually unnecessary)
- Sorting and selecting the best frames for stacking
- Optionally saving the workspace for later continuation (optional, based on whether you need to stop at any time)
- Stacking the frames

- Optionally blurring the stack (usually unnecessary)
- Optionally sharpening the stack (usually unnecessary)
- Level adusting the stack
- Denoising the stack
- Removing electrical interference patterns from the stack
- Optionally beautifying stars
- RGB channel alignment on the stack

Bringing the raw frames into the workspace

Bringing individual frames into the stack is performed exactly as in planetary processing.

DeBayering the frames

It will be necessary to deBayer the frames if the images came from a RAW modded camera that transfers the Bayer matrix to the computer. This is performed exactly as in planetary processing.

Optionally subtracting a dark frame from the frames

Although I list this is an optional step, it is practically required for any decent deep sky image processing.

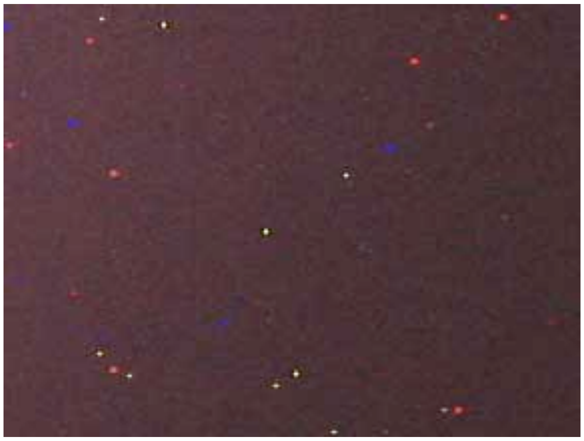
CCDs accumulate several kinds of noise over the duration of a long exposure. The worst offenders are thermal noise, hot pixels, and amp glow.

Thermal noise is a steady influx of noise that results from the ambient temperature of the CCD and its nearby circuitry. This produces a smooth swathe of buildup in the pixel values evenly across the entire image. Although the rate of thermal noise accumulation can be slowed by keeping the CCD as cool as possible, there is always some degree of thermal noise buildup.

Hot pixels are physical defects in the CCD where specific pixels fully saturate very quickly, usually in a matter of seconds. Hot pixels look like bright speckles of various colors that have been peppered across the image. They always appear in exactly the same place on all images, since they are physically present in the CCD sensor itself.

Amp glow is the result of the webcam circuitry which attempts to boost the image signal before sending it across the USB cable to the computer. The amplifier is located on one side of the CCD and causes a gradient of value accumulation that builds up unevenly across the image.

Here is an example of a dark frame:



All of these defects are easily dealt with by subtracting a dark frame from the individual frames prior to stacking them. A dark frame is a frame which received no light and represents an exposure time equal to the actual frames themselves. The dark frame represents all the value in an image of a given duration that did not actually result from real light hitting the CCD. By subtracting this value out, the frames are left only with real information about light that hit the CCD.

To open a dark frame image simply choose the appropriate menu command under the File menu. Bare in mind that the dark frame must be the same dimension as the individual frames. It is also important not the crop the frames before opening a dark frame. Open the dark frame first. Then you can crop the frames and the dark frame will be cropped as well.

Once a dark frame has been opened, you can subtract it from an individual frame by control-clicking on the frame's clip in the Clips Window. Alternatively, you can select any number of frames and control-click on any selected frame to subtract the dark frame from all of the selected frames at once. You can always undo a frame's dark frame subtraction by control-clicking on its clip again.

Any frame that has been dark frame subtracted will have its clip marked with a 'D' in the upper right corner of the clip.

Hint: It is a good idea to take several dark frames and stack them, producing an average dark frame before processing your actual images. Using the average dark frame image as the dark frame for your actual images will produce more accurate results in your final image.

Optionally dividing a flat field from the frames

While this is certainly an optional step, and you can definitely get good results without it, if your imaging system is producing uneven light sensativity across the CCD, your results will improve dramatically if you start using flat field division.

CCDs are often unevenly sensative to light across their entire surface. In addition, the optics of a telescope projecting an image onto a CCD can produce an uneven light sensativity across the CCD. This problem is dealt with using a flat field.

A flat field is an image of a diffuse light source, thereby capturing the light sensativity patterns present across the CCD for a given optical setup. Flat fields are usually acquired by capturing a picture of a twilight sky.

Here is an example of a flat field (although in there case there is only a minor degree of uneven light sensativity across the image):

Once a flat field image has been acquired, it can be divided out of the individual frames after dark frame subtraction and before stacking.

To open a flat field image simply choose the appropriate menu command under the File menu. The same issues that applied to the dark frame apply to the flat field, so don't crop the frames until after opening a flat field frame, if you intend to use a flat field at all.

To divide a flat field from a single frame control-shift-click on the frame's clip, or make a selection of clips and control-shift-click on any selected frame. You can always undo a frame's flat field division by control-shift-clicking on its clip again.

Any frame that has been flat field divided will have its clip marked with an 'F' in the upper right corner of the clip.

Optionally resampling the frames

This doesn't serve much purpose when processing dark sky images, but if you find it useful, it can be done. The process is described in the section on planetary processing.

Aligning the frames

Aligning dark sky images is essentially identical to aligning planetary images. The only difference is deciding what sort of feature to define a rectangle around. I generally align dark sky images in two stages. I first define a rectangle around the main object, say the core of a galaxy or nebula, and do a **'1'** translational alignment on it. After that, I define a very small rectangle around a sharp feature like a star and do a **'3'** translational alignment. If no stars are present in the image, then clearly it is impossible to do the second step.

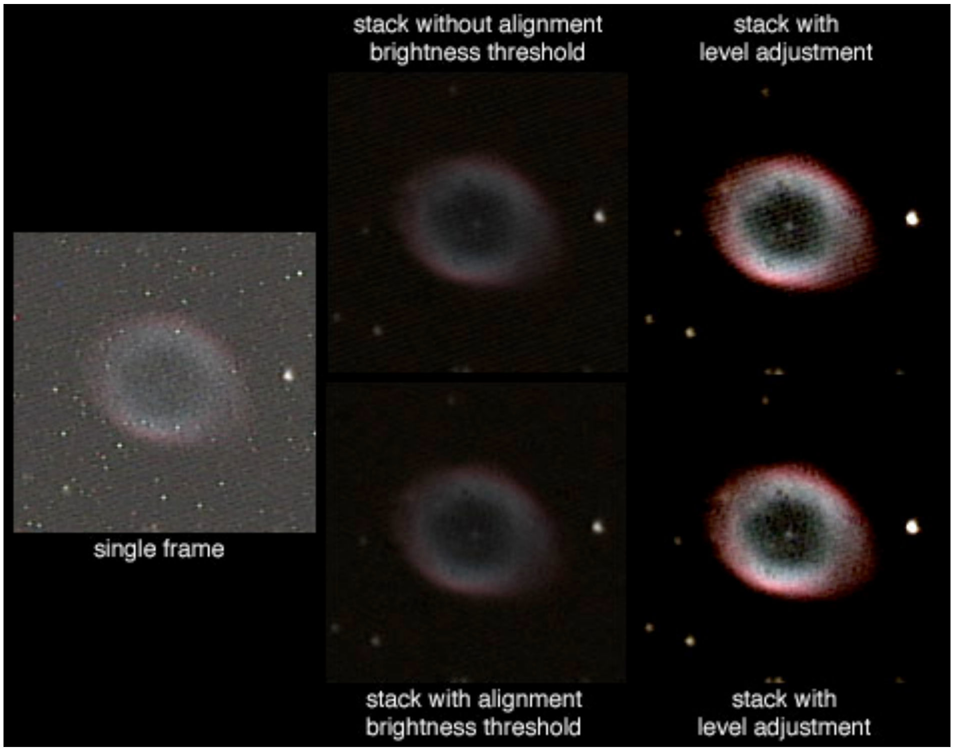
Don't forget to rotationally align your images as well. Long exposure photography, and long duration photography in which imags are captured over a long period of time, *e.g.*, minutes, hours, etc. may very well suffer from some rotation if the telescope's polar alignment is not precise (or doubtlessly if an alt-az mount is used instead of an equatorial mount). The method for rotationally aligning images and the issues that arise when both translationally and rotationally aligning images are discussed in detail in the section on planetary processing. Read that section carefully.

It is often useful to help the translational alignment along if the frames are dispersed over a large area by manually nudging them into a very coarse alignment prior to autoaligning. This can be done to an extremely sloppy degree and will help a lot.

It is also important to be aware that you have the option of setting a brightness threshold for the difference alignment algorithm. You can find this command under the Operations Menu. The menu item is called Set Alignment Brightness Threshold. This command allows you to state that only pixels of a minimum brightness will contribute to the difference calculation and therefore to the alignment procedure. This is particularly useful when the background exhibits artifacts such as faint thin diagonal lines (the infamous *herring bone* effect). These artifacts would otherwise be enhanced during alignment. This would have the undesirable effects of introducing the artifact into the final stack while simultaneously corrupting the precision of the alignment on the actual object in the image. By thresholding the alignment, the artifact can be minimized or eliminated in addition to obtaining a sharper more precise alignment on the object itself.

Note that in dialog boxes such as the one presented for setting the brightness threshold you can adjust the values with the arrow keys and shift-arrow keys, as well as typing numbers into the boxes.

The following image shows an example of the effect of setting an alignment brightness threshold. Also notice that in the final images, the central star of the nova is sharper when using this feature. This occurs because the alignment is more accurately aligning the real information in the frames instead of getting corrupted by the interference pattern.



Optionally cropping the frames

If you wish to crop the frames, the best time to do it is right after aligning them. The process is described in the section on planetary processing.

Optionally normalizing the frames

The reasons why this might be useful and how to do it are described in the section on planetary processing.

Sorting and selecting the best frames for stacking

The process by which to sort deep sky images is identical to that for planetary images. However, in practice, you usually have far fewer long exposure images and the bad frames will really stand out. Selecting them and Xing them out should be fairly simply in the case of deep sky image processing, unlike the complicated process necessary for processing planetary images.

Optionally saving the workspace for later continuation

Similar to the description for planetary images, you can save and retrieve the state of the workspace at any time. This is useful if you need to stop processing a set of images for a while and are unable to leave the application open continously.

Stacking the frames

Stacking is performed exactly the same as in planetary processing. However, note that the stack starts out as a pure average of the images. While this is usually what you want in planetary imaging, what you really want in deep sky imaging is usually something more like a sum of the images, which will make the image brighter. This will be taken care of later when you level adjust the stack.

Planetary shots are generally extremely brief exposures, and as such do not exhibit hot pixels. This is not the case for long exposures. You already got rid of the hot pixels by performing dark frame subtraction. However, this can cause problems as well. If the individual frames line up too well, the subtracted hot pixels will cause black holes to appear in the stacked image. Even if alignments drift slightly across the CCD, you can still get faint dark streaks in the stacked image. There are two ways to combat this problem.

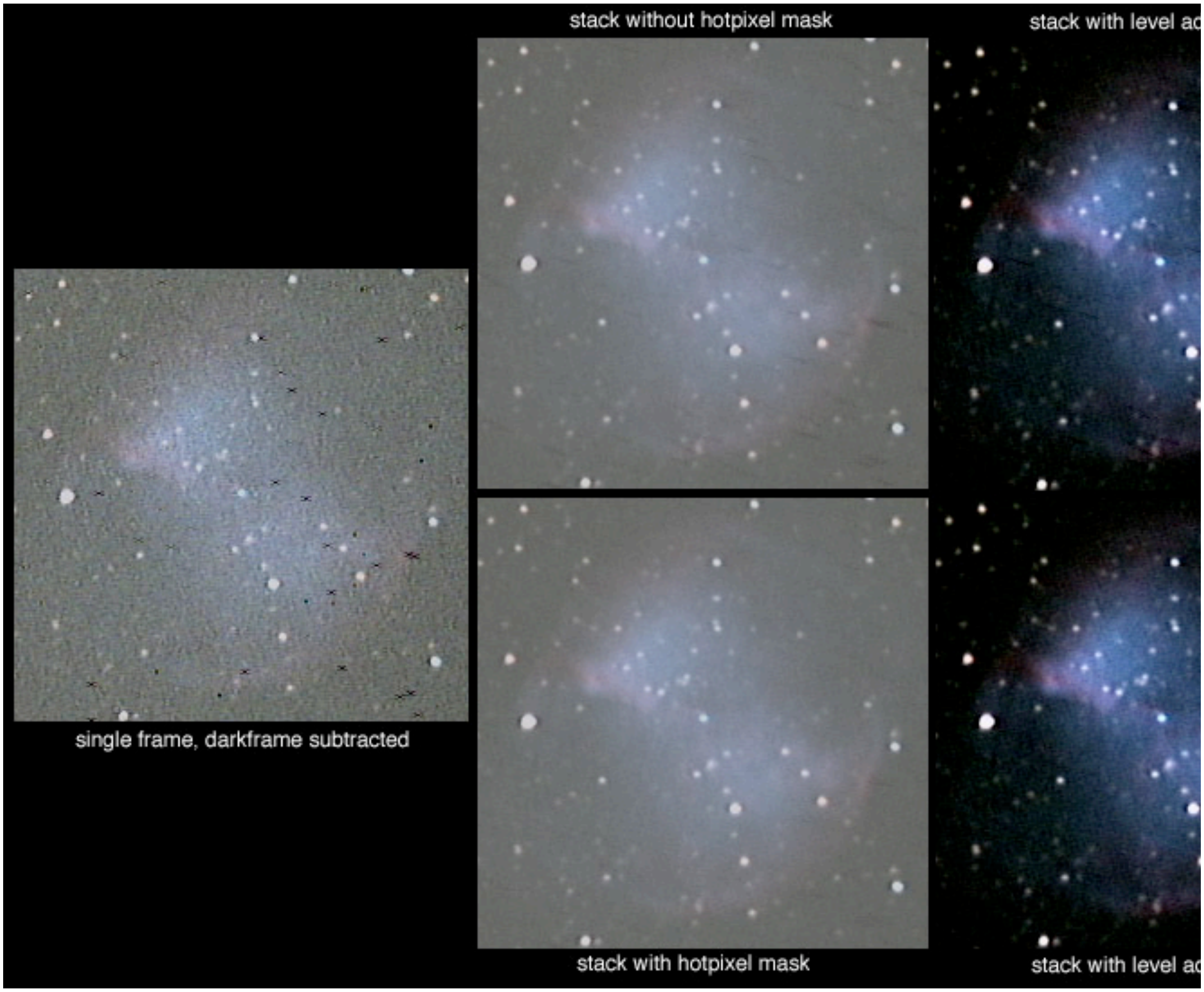
The first and preferable way is a command under the Operations Menu called Create Hot Pixel Mask which will let you define a mask that hi-lights the hotpixels while eliminating the rest of the image. Sometimes it is helpful to extend a radius around the hotpixels as well. Creating a hot pixel mask tells the program to not use hot pixel locations in an image at all during the stacking process. If all the frames are perfectly aligned this will result in holes anyway, not because of the hot pixel subtracted holes but because there will be no information at the hot pixel locations to accumulate in the stack. However, if your frames are offset from one another, then the likelihood is that any given location in the stack is represented by at least some individual frames and only the information-containing pixels in each frame will contribute to the stack.

Note that in dialog boxes like this, you can TAB between the fields.

The second way to avoid hot pixel holes is to median filter the individual frames prior to stacking. You can do this by hitting the 'M' key while over a single clip or by making a selection of clips and then hitting the 'M' key. This will bring up a dialog box in which you can specify a median filter radius and a value threshold. Only pixels that are darker than the threshold will be median filtered. All other pixels will remain unaltered. This allows you to fill in the hot pixel holes in the individual frames using nearby pixels as a reference, while leaving nonhot pixels uncorrupted.

The first method is prefereable, if it works. The reason for this is that the first method only accumulates real information in the final stack, information that was truly gathered during imaging in the individual frames. The second method "fakes it" by guessing what the hot pixel holes ought to be filled in with. There is no guarantee that a filled in hot pixel hole will have the true correct value, although hot pixels generally vanish completely using median filtering, which suggests that very good guesses are being made. Such guesses contribute corrupted, although believable, information to the final stack. Nevertheless, the final result is quite impressive. However, the first method is not always satisfactorally effective, since it is dependent on a variety of alignments across the set of individual frames. The second method can be used to supplement or replace the first method in such cases.

The following image shows an example of eliminating hot pixels using a hot pixel mask. The effect would be similar for using median filtering as well. The results speak for themselves.



Optionally blurring the stack

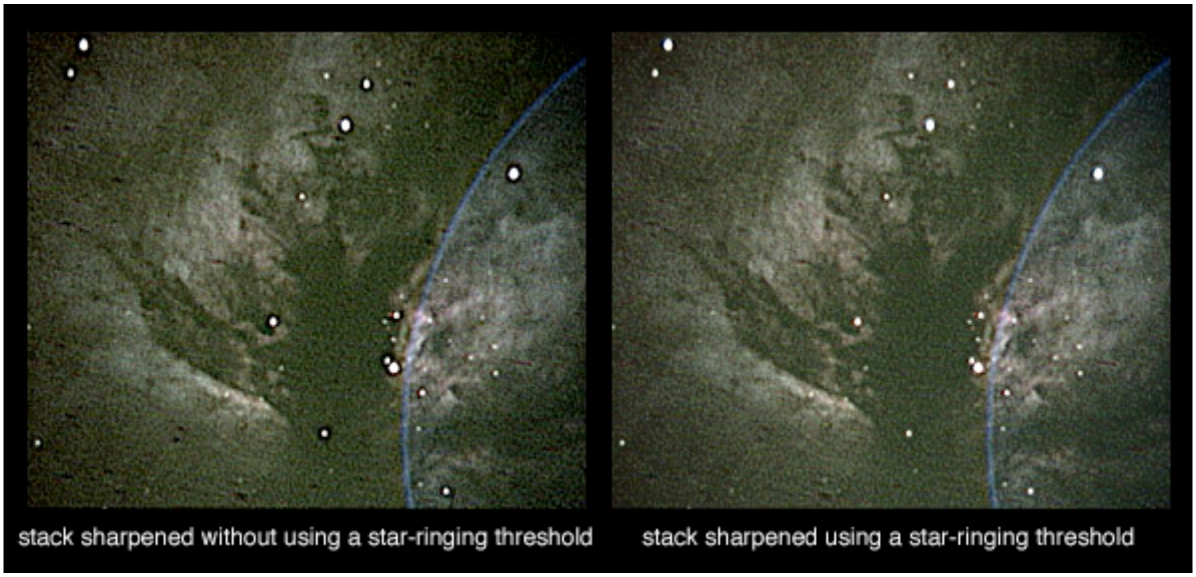
The reasons why this might be useful and how to do it are described in the section on planetary processing.

Optionally sharpening the stack

While planetary images usually benefit from a copious amount of sharpening, this is usually unncessary to sharpen deep sky images. However, some deep sky objects have a substantial degree of sharp detail in them, generally nebulas, and some face on spiral galaxies. The process for sharpening is exactly the same as in sharpening planetary images.

Deep sky images often have stars in them. Sharpening such images using techniques similar to Unsharp Masking and Laplacian Sharpening generaly produce an undesirable artifact in such images in which stars end up being surrounded by dark halos or rings. To understand why this happens, one needs to understand how Unsharp Masking and Laplacian Sharpening work in the first place. Although it takes a lengthy description to fully understand it, it is sufficient to realize that these methods sharpen a blurry edge by making the bright side of the edge brighter and the dark side of the edge darker, which yields dark rings around stars. In order to prevent this effect from happening, there is a textbox in the Interface Dialogs for the sharpening methods which allows you to specify a value cutoff. The default is 0, which means the program will make no effort to prevent "star ringing". Increasing the value in the checkbox increases a cutoff for the sharpening process. Pixels with values greater than or equal to the specified cutoff are sharpened normally, being brightened or darkened as necessary for the sharpening process. Pixels with values below the cutoff are only sharpened if they are to be brightened. If the sharpening process requires such pixels to be darkened they are simply left alone, unsharpened. The edge of the star will still be sharpened because the other side of the edge (inside the star's disk) contains pixels that will be brightened in the usual manner.

The following image shows an example of eliminating star-ringing using a value threshold during sharpening. Notice that there is a secondary benefit of cleaning some residual hot pixel holes that although weak after attempting to eliminate them them, can be strengthened by the sharpening process.


stack sharpened without using a star-ringing threshold
stack sharpened using a star-ringing threshold

Level adusting the stack

One of the simplest steps is level adjusting the stack. This is performed exactly as in planetary processing. However, note that while the process itself is identical, you will usually adjust the levels differently than you would for planetary images. Since deep sky images are often quite dim, you can brighten them up considerably. Also make good use of the gamma controls to bring out the fainter portions of deep sky objects.

Denoising the stack

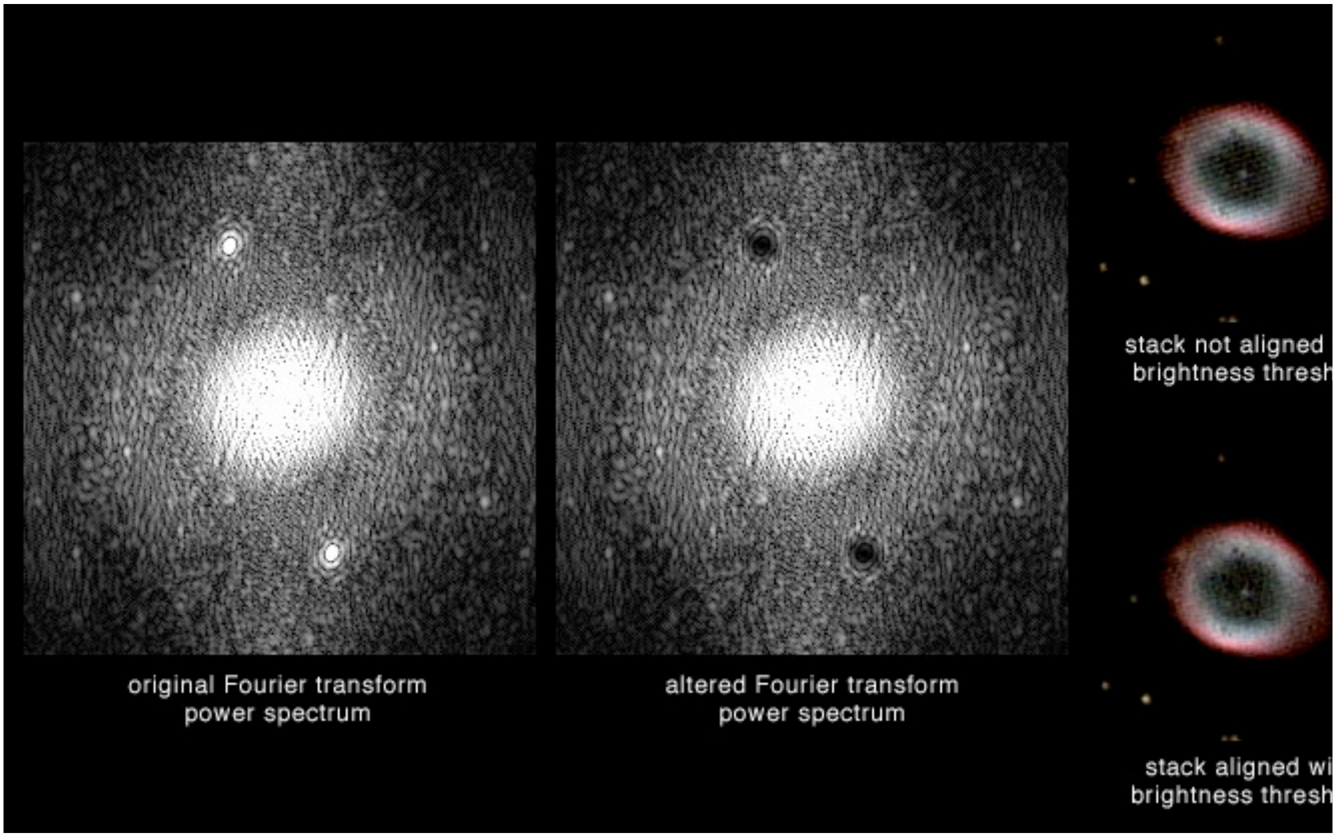
You may want to attempt to remove some of the noise from the final image. This is performed exactly as in planetary processing.

Removing electrical interference patterns from the stack

Webcams that have been modified for long exposures are often subject to a variety of forms of electrical interference which can introduce artifacts into the raw images. One common pattern is called the "herring bone" pattern in which a series of fine diagonal lines permeate the entire image. This problem can be dealt with to some degree by using a brightness threshold during image alignment, as described above in the section on aligning the frames. That section has an example which shows the herring bone pattern. Notice that even after properly aligning, the herring bone pattern is still slightly visible, although greatly reduced.

One possible solution is to capture a larger number of images for stacking, which will help smooth and average out the interference pattern, but another method is to directly remove the pattern from the final stacked image. This is readily achieved by use of the Fourier Transform Editor command under the Operations menu.

The following image show the example from above again, this time comparing the final results from above with the results that can be achieved with subsequent Fourier transform editing. The interference pattern is easily identifed in the power spectrum as a matching pair of bright dots. Simply paint them out with the editor and the results speak for themselves. Note that the combination of both brightness threshold alignment and Fourier transform editing is the sharpest (look at the center star and the patterns in the ring). Proper alignment is necessary to get the sharpest alignment of the frames, so don't abuse the Fourier transform editor as a means by which to solve problems like this outright. Instead, use it to patch up the final image.



Optionally beautifying stars

Stars often suffer from two ailments in astrophotos. The first is bloat, in which bright stars expand into a large disks in the raw exposures, when in theory all stars should be a single point regardless of brightness. The second is ringing, which results from various methods of post-processing such as Unsharp-Masking, Laplacian Sharpening, and Wavelet Denoising. Ringing can also occur in raw exposures as a result of quirks in the way CCDs gather light from bright point-sources such as stars. Ringing can be handled at two stages in this program. The first is in the sharpening operations, in which a threshold can be specified which will help alleviate ringing. However, that method is not always entirely effective and similar methods in the Denoising interface don't work very well. Therefore, it is sometimes necessary to perform further deringing using the Star Beautification tool, as well as possible debloating.

The first step is to help the program actually locate the stars. Stars are recognized as contiguous "paint-fills" of bright pixels starting from an arbitrary bright pixel. The first thing to do is to set the star brightness threshold to detect the stars you are interested in. Follow that by adjusting the star spread to fill out most of the area of the stars. By pushing the spread down but leaving the search brightness high, you can more easily avoid picking up nonstars. You can see the detected stars at any given time in two ways. The first is to check the Overlay Star Mask option and the second is to check the Overlay Dering Radius option. Both methods will show the locations of detected stars, but only the Mask method will show the specific pixels that are detected as stars. The Beautification tool will work best if you thoroughly cover the star, so use the Mask to make sure you do this.

As you decrease the star brightness threshold and spread threshold you might start picking up a lot of small dim stars that you don't want to operate on because you are pleased with their present appearance. Use the Minimum Star Surface Area control to eliminate small stars.

Likewise, in pictures with bright elements that are not stars, such as the cores of nebulas and galaxies, the star detection will probably detect these areas as stars as well, which you obviously don't want. There are two ways to combat this problem. The first is to use the Maximum Star Surface Area to eliminate really large detected "stars", which can help rule out nebula and galaxy cores. The second is to drag a rectangle in the thumbnail image to remove areas of the image from any attempts at star detection. Click-dragging without the option key will add areas to the total area to be searched while Click-dragging with the option key will remove areas from the total area to be searched. The default search area is the entire image.

Once you are relatively pleased with the star detection, you can perform either deringing, debloating, or both. Deringing is a simple matter of setting the Dering radius large enough to encompass the worst ringing in the image, and setting the Dering Brightness Threshold to remove the rings.

Debloating is accomplished through truncation of the star outside a specified radius. Simply pick the desired radius and smooth it out some. That's it.

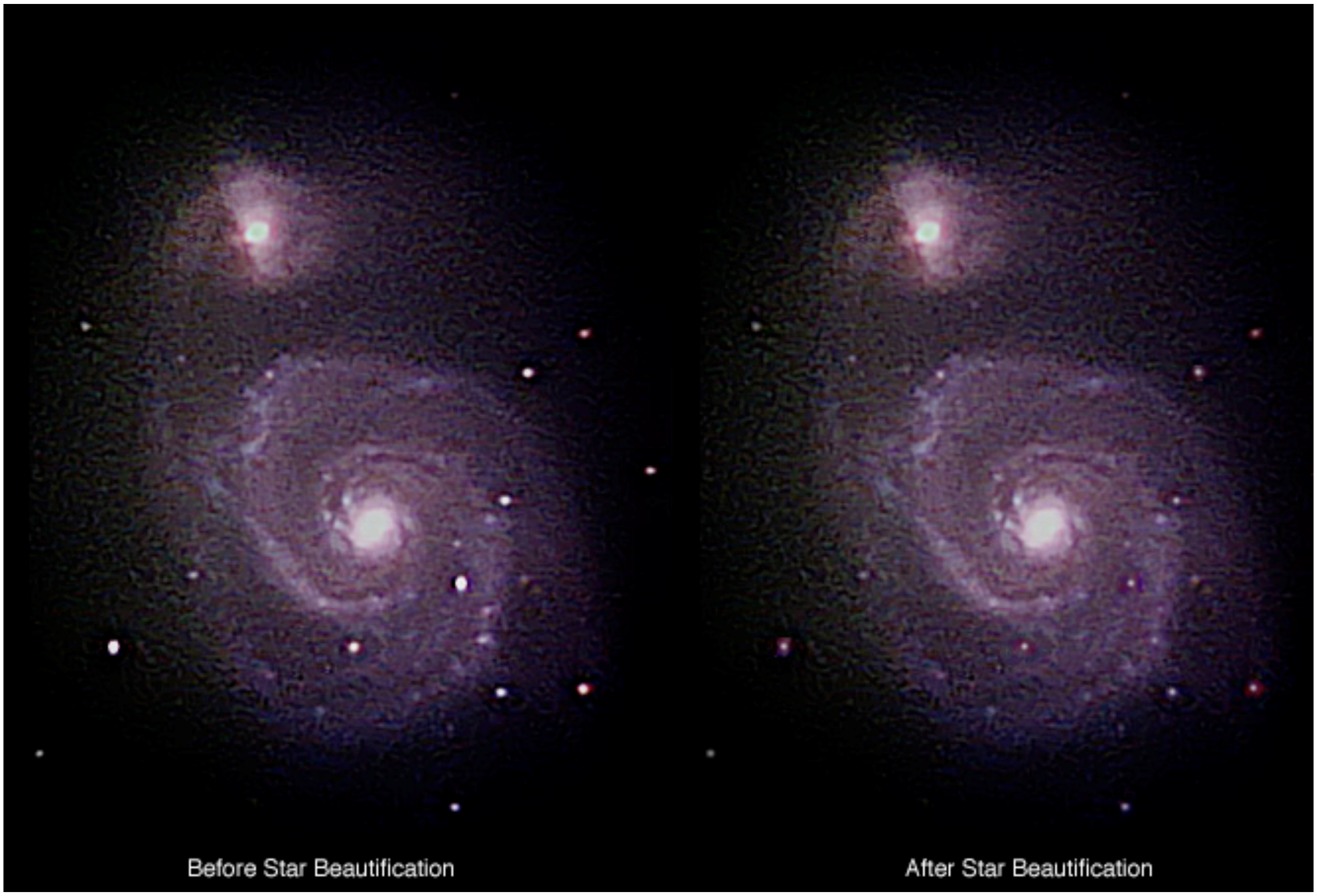
Be aware that the order of operations performs Deringing prior to Debloating, so it is best to pick the settings for these operations in that order as well.

Now that you understand the basics of using this tool, it is useful to understand a few specifics as well. Both the Deringing and Debloating can be finicky and it can be difficult to get really superb results. The first and most important thing to realize is that different stars in different areas of the image might require different degrees of processing. If this is so, do not try to operate on such stars simultaneously. Use the rectangle-dragging trick to eliminate areas of the image and perform Beautifaction in stages by exiting the dialog and opening it again for further processing.

If you discover that your truncated (debloated) stars, have rings around them that are brighter than the surrounding area of the image (easy to detect by using the realtime RGB and HSV readout that is present when the cursor is over the FrameStack window) then you have not set the Star Brightness Threshold or the Star Spread Threshold low enough. This leaves pixels that are brighter than the background outside the star mask which will then be used to approximate the brightness of the truncated area around the star.

Do not be alarmed by the initial appearance of truncated stars before you adjust the Smoothness control. They will look very artificial until you smooth them over a fair degree. The Smoothness control is simply a Gaussian blur that is limited to the bounding rectangle of the star mask instead of the entire image.

Although this tool does not produce truly ideal results, it does quite a nice job of cleaning up stars, as shown in the example below:



RGB channel alignment on the stack

You can manually align the red, green, and blue channels of the final FrameStack separately if you with. The method for doing this is described in the section on planetary processing.

That's it. You can save the stacked image and you're done.

Multithreading Options

If you are new to this program and are trying to familiarize yourself with the basics or are otherwise trying to simply get up and running, then I implore you, skip this section entirely.

Most modern computers have multiple processors and/or have processors which have multiple cores. For our purposes we need not concern ourselves with the distinction between these two concepts. The important point is that both of these features enable computers to perform true parallel processing, *i.e.*, they can perform multiple operations at once. One obvious advantage of parallel processing is that time-consuming tasks can be accomplished more quickly.

Keith's Image Stacker exploits parallel processing through multi-threaded task division for many of its basic operations. Specifically, most operations which are performed on a set of frames in the Clips Window can be performed in parallel by dividing the set of frames up into multiple subsets, one per thread. The GUI provides a fairly simple interface for controlling how threads are used. When used properly, multiple threads should enable you to vastly speed up the program. The menubar has a Threads menu, which in turn has two sections. In the main section you can specify how many threads the program should use for operations which can be performed in parallel. The question then arises, how many threads should you specify? The basic rule of thumb is simply to assign about one thread per core on your machine. This assumes that you do not have any other processor-heavy programs running at the same time. So if you have a single processor dual-core machine, the rule of thumb would prescribe two threads. If you have a dual processor, quad-core machine, then eight threads would be the suggestion, and so on. The basic logic here is that each core should have a thread to work on so it is kept busy instead of sitting idle without any work to do.

However, in some instances, there is a further benefit to be realized by assigning as many as two threads per core. In other words, some further speedup may be achieved by assigning up to two threads per core. The reason why this can be beneficial is that a thread might stall while it is waiting for data to be loaded from RAM into a low-level cache. While waiting, the core in question can load the second thread and do useful work instead of sitting idle. This same benefit can, on occasion, reach past two threads per core, but just as likely, the overhead of handling numerous threads can actually cause a *decrease* in overall performance if threads are abused (a task that takes sixteen seconds with four threads might take twenty seconds with eight threads on the wrong machine!). Therefore, it is recommended that you not specify more than two threads per core. It should be noted that the benefit of going from one thread per core to two is far less significant than from going from one thread overall to one per core. That is where the main speedup will be achieved.

NOTE: The number of threads you wish to use will be saved when you quit and restart the program, so you shouldn't need to reset this setting every time the progam starts up.

A second setting is also provided from the Threads menu. This setting indicates whether the program should make sure the display stays synchronized to the progress of the various threads during a long multi-threaded operation. This synchronization mainly pertains to the thick cyan border that is used to highlight the current working clips and to redrawing the updated clip of a frame in the Clips Window immediately after it has been modified. These methods of synchronization require some threads to pause briefly and this can slow down the overall operation slightly. If you do not find such synchronization to be crucial and are more interested in the fastest possible throughput, you might consider disabling display synchronization.

FAQ (Frequently Asked Questions)

> *Blessed art those who read thy documentation and thy FAQ to completion, followed by dutiful attention to thy tutorials.*

Topics

- I prefer my documentation in HTML with hyperlinks and all that sci-fi stuff.
- The program quits immediately after it is launched.
- How do I open images?
- Why can't I open my digital camera's RAW images?
- How do I align images?
- Why does my stacked image look really dark after sharpening or denoising?
- Oye! What are all the various uses of the number keys?
- Should I save my stack as PICT, FITS, or RAW?
- I see fancy technical sounding options in the menus. They scare me. What are they?
- Which sorting method should I use?
- I want to sort by power spectrum distribution but I get an error that complains about power-of-two bounds something-or-other instead.
- Threads menu. Yeah, give me the dish on that.
- I think I found a bug. Yeah, definitely a bug.
- Where's the answer to my perfectly valid question that hasn't been addressed yet?
- Would you add such-and-such new feature please?
- Yeah, so, I just downloaded your program and it, like, looks really cool. Could you please explain how to use it? That'd be great, thanks.

I prefer my documentation in HTML with hyperlinks and all that sci-fi stuff.

The documentation can be downloaded in HTML format on the program's website (see the documentation front page for the URL).

The program quits immediately after it is launched.

There are multiple versions of the program corresponding to various architectures and Mac OS versions. Note whether you are attempting to run on a PPC or Intel mac and whether you are attempting to run on OS 10.4 and below or on OS 10.5 and above. Then check the version of the program you are attempting to run by selecting the application in the Finder, choosing Get Info under the File menu, and looking at the *Version* string of the info box. If the program does not specify the correct architecture/OS combination, download the correct version of the program (see the documentation front page for the URL) and try again.

If the problem persists with the correct version of the program, contact me. Maybe we can sort it out together. My contact information is on the documentation front page.

How do I open images?

Images come in two forms, individual image files and Quicktime movies. Either kind can be drag-and-dropped onto the application's icon in the Finder or in the Dock, and either kind can be opened via the relevant menu option under the File menu.

Why can't I open my digital camera's RAW images?

Have you read the RAW image documentation yet, as referenced from the documentation front page?

How do I align images?

There are two forms of image alignment: translational (sliding around) and rotational. Translational alignment can be done in three different ways: difference, centroid, and cross-correlation. If you want to keep things simple, just use the difference method and don't sweat the details. Rotational alignment can only be done using the difference method.

The first thing to do is define an Operation Bounds Rect. This is a rectangle that specifies an area where the frame being aligned will be compared to the reference frame (the first frame in the Clips Window). Start by making sure no clips are selected (Deselect All from the Edit menu). Then shift-click one clip. Conventionally one would use the reference frame, so shift-click the first clip in the Clips Window. Move to the Frame Inspector Window. If it is blank, you don't have exactly one frame selected. Try again. Otherwise, click-drag a rectangle in the Frame Inspector Window to define the Operation Bounds Rect. I recommended that you keep the Operator Bounds Rect small (to speed the alignment process up) and include edges and other interesting areas of the image in the Operation Bounds Rect (this will make alignment perform better).

Now you are ready to align the frames. Deselect all the frames (Edit menu). To align a single frame, hold down the '1' key (large distance difference alignment) and click one clip in the Clips Window. It will autoalign. If the alignment isn't very good, give a second or third try, it may converge on the proper alignment over a series of repeated attempts. Alternatively, you might consider defining a different Operation Bounds Rect or manually nudging the clip closer into alignment (sloppily and quickly) before running autoalignment again.

To align multiple frames at once, shift-click or shift-click-drag a selection of clips in the Clips Window. To align all the frames choose Select All from the Edit menu. In either case, multiple or all, hold down the '1' key and click on any selected clip. They will all autoalign.

To perform rotational autoalignment, use the '6' key. Rotational autoalignment will align minor rotational misalignments better if the Operation Bounds Rect is ***not*** near the center of the image. Think about it, it makes sense.

Why does my stacked image look really dark after sharpening or denoising?

Have you read the dynamic range documentation yet, as referenced from the documentation front page?

Oye! What are all the various uses of the number keys?

It is true, the number key commands are rather confusing. My apologies. Let's go through it slowly.

All number key commands use the same basic interaction: you press and hold down a number key. Then, while the number key is down, you perform a second operation (mouse click or tapping a second key for example). This behavior is similar for some other keys as well. For example, to delete a clip you hold down the 'D' key and then click a clip with the mouse. So the question is, what are the commands that can be performed with the number keys?

Let's consider autoalignment first. Make sure no clips are selected (Deselect All from the Edit menu). Hold down the '1' key and then click a clip in the Clips Window. A translational autoalignment using the difference method is performed. This is also true for the '2' and '3' keys. The only difference is the distance over which the three numbers autoalign the fastest. '1' is fastest on 15+ pixel misalignments, '2' is fastest on 5-15 pixel misalignments, and '3' is fastest on 0-5 pixel alignments.

To autoalign multiple clips at once, select them by shift-clicking or shift-click-dragging a set of clips. Hold down the '1' key and click any selected clip. All selected clips will then be autoaligned.

The '4' and '5' keys also perform translational autoalignments and work exactly like the '1', '2' and '3' keys: hold '4' down and click a clip or any one of a selected group of clips. '4' uses the centroid method and '5' uses the cross-correlation method. If having five options for translational autoalignment confuses and bothers you then simply ignore all of them except the '1' key method. It is simple and staightforward and will handle almost any situation adequately.

The '6', '7', and '8' keys perform rotational autoalignment and work just like the three difference-based translational alignments, with '6' being fastest for large misalignments and '8' being fastest for small misalignments. Again, if this is too complicated, just use '6' all the time.

The number keys are also used for red/green/blue individual channel alignment when manually translationally nudging clips. To nudge just the red channel of a clip, select Align RGB Channels Separately from the Operations menu. Deselect all clips. Then hover the mouse over any clip in the Clips Window. It is not necessary to press the mouse button. Now, hold down the '1' key and tap an arrow key. The red channel will be nudged. Hold down the arrow key to continuously nudge the red channel. '2' nudges green and '3' nudges blue. This works with a selection of clips just as you would expect. By the way, separate RGB nudging also works on the final stack, not just the clips prior to stacking. Try it.

Observe that these number/key mappings (1:red, 2:green, 3:blue) also match the options in the View menu for viewing channels of the stack individually (this only works on the stack, not the clips). This similar number/key mapping is intentional and should assist comprehension.

Should I save my stack as PICT, FITS, or RAW?

To see your stack in other environments such as other image processing programs or on the web, you will need to convert it to a conventional image format such as tiff or jpeg (there are literally thousands of image formats). To do this, save the stack as PICT. Then open the PICT with Preview, which is a free program provided by Apple. You can find it in your Applications folder. Once the PICT is opened in Preview, choose Save As and convert it to the format of your choice.

The problem is, all conventional image formats are 32 bit, 8 bits per red/green/blue component and 8 bits unused (or used for the alpha channel). Your stack will probably have a much greater dynamic range than this and saving to PICT destroys that information. To save your stack with all of its information perfectly intact you should save it as FITS or RAW, both of which can perfectly preserve your stack without any loss. The cost, of course, is that not very many programs can open FITS or RAW files, and even programs that can open some FITS or RAW files may not open them all. For example, some programs can open 16-bit-per-component RAW files, but *Keith's Image Stacker* can read and write both 16- and 32-bit-per-component files.

Here's what you should do. Save your precious processed stack as FITS or RAW and archive it forever so you can come back to it with *Keith's Image Stacker* anytime you want without suffering any loss of quality. Then, additionally, save your stack as PICT and convert it to jpeg for viewing on the web.

I see fancy technical sounding options in the menus. They scare me. What are they?

The menus provide commands for Dark Frames, Hot Pixel masks, Fourier thingies, Wavelet something-or-others, and generally lots of fancy technical sounding stuff. If you want to know what they're about you must read the documentation (and probably need to learn more about astrophotography in general). On the other hand, if you don't already know what these commands are for then you probably don't need to understand them. They are rarely required, instead offering improved results at the power-user level, so just ignore them. When you become a totally leet user and are ready to expand your horizons, they will be there waiting for you, with associated documentation.

Figure out how to align a set of images and generate a stack. That's 90% of image stacking as a whole. Then learn about dark frames, and maybe, just maybe, about flat fields. Then, when you're ready, read about all the other stuff.

Which sorting method should I use?

Four sorting methods are provided: *difference*, *value*, *value range*, and *power spectrum distribution*. Although the menu options provide a description of each, they don't actually prescribe one over the others. Bottom line, under most circumstances, sorting by power spectrum distribution is the best idea since it favors in-focus images over out-of-focus images. However, it is also the most complicated since it involves a user interface which must be used properly to get good results. To prevent noisy images from being preferred, pull the high frequency cutoff down to remove high frequency noisy grain from ruining the sorting results.

I want to sort by power spectrum distribution but I get an error that complains about power-of-two bounds something-or-other instead.

Sorting by Power Spectrum distribution requires an Operation Bounds Rect that is a perfect square with sides of length 32, 64, 128, or 256 pixels. No other Operations Bounds Rect is permitted for sorting by this method. To create such an Operation Bounds Rect, set the desired size from the corresponding submenu of the Operations menu. Then, while defining an Operation Bounds Rect, hold down the Control key. You will get a perfect pow-2 square that you can place anywhere you want in the frame. Note that if you move too close to the edge of the image, the square will disappear until you move back toward the center.

Threads menu. Yeah, give me the dish on that.

If you want to keep things simple, ignore this menu forever, although learning a little about it may help you speed the program up significantly.

If you are ready to learn the way of the thread, then read the relevant documentation, as referenced from the documentation front page. You will want to understand concepts like *multi-processor* and *multi-core* before attempting to use threads. Otherwise save yourself the trouble and leave it alone.

I think I found a bug. Yeah, definitely a bug.

Grrr, my apologies. Please contact me and let me know about the problem. My contact information is on the documentation front page. It would greatly assist me if you could characterize the bug as precisely as possible. What is the minimal circumstance under which it occurs (minimum number of images brought into the workspace, minimum number of operations performed to trigger the bug, that kind of thing). If you can generate a perfectly reproducible, yet relatively small, example, and wouldn't mind sending me the input images and a description of how to reproduce the bug, I would greatly appreciate it. Lastly, please tell me your system configuration (PPC/Intel, OS X version, Keith's Image Stacker version). Thanks.

Where's the answer to my perfectly valid question that hasn't been addressed yet?

First, have you read the documentation and worked through the tutorials? Yes? That's a great start, I am truly sorry you are having trouble. I am always welcome to questions by email and will attempt to help in any way I can, but I reserve the right to respond in a condescending tone if the answer is in the docs. You can find my contact information on the front page of the documentation.

Would you add such-and-such new feature please?

I would ***consider*** it, although I don't make much money off this program so my time is hardly justified...In fact, I probably won't do it. The program is getting pretty old and mere maintenance is justified at this point. A complete rewrite is in order, but I'm doubtful that I will ever invest the time and effort. Sorry.

Yeah, so, I just downloaded your program and it, like, looks really cool. Could you please explain how to use it? That'd be great, thanks.

Ask your question on one of the many astrophotography discussion groups. I don't have time for this kind of nonsense.

Command Reference

MENU COMMANDS

How to Perform the Command	Description of the Command
File: Open and Add Images/Movies	Brings images and quicktime movies into the workspace. Details of the interface for opening movies are described in the section on planetary processing.
File: Open Dark Frame	Selects a dark frame image.
File: Open Flat Field Frame	Selects a flat field frame image.
File: Open FITS File as Stack	Opens one FITS file as a red, green, or blue channel of the stack.
File: Open Raw File as Stack	Opens one Raw file as the stack.
File: Save Stack as 32-bit PICT	Saves the stack as a 32 bit PICT file.
File: Save Stack as 48/96-bit FITS	Saves the stack as three FITS files, one each for red, green, and blue, at 16 or 32 bits per component per file.
File: Save Stack as 48/96-bit Raw	Saves the stack as a Raw file at 16 or 32 bits per pixel component.
File: Open Workspace	Opens a workspace file (the state of the Clips Window).
File: Save Workspace	Saves the workspace (the state of the Clips Window) to a single file for further processing.
File: Dump Workspace	Dumps the workspace (the state of the Clips Window) to a set of PICT files.
File: Close Window	Close the current window.
Edit: Select All	Selects all the frames.

Edit: Deselect All	Deselects all the frames.
Edit: Invert Selection	Inverts the frame selection.
Edit: Select Included	Selects only the frames that are to be included in the stack.

Operations: Create Hot Pixel Mask	Creates a mask that excludes hot pixels during stacking.
Operations: Set Alignment Brightness Threshold	AssignS a brightness threshold for use on subsequent difference-alignment operations.
Operations: Align RGB Channels Separately	Toggles whether or not to align the red, green, and blue channels separately.
Operations: Power^2 Bounds Dim	Selects the dimensions of the Operation Bounds Rectangle when using the power-of-2 option.
Operations: Crop Clips	Crops the frames prior to stacking.
Operations: Resample Clips	Resamples the frames prior to stacking.
Operations: Rotation Method	Selects an interpolation method for rotation.
Operations: Sort clips	Sorts the frames based on the method selected. Details are given in the section on planetary processing.
Operations: Sum/Stack all included frames	Produces a 96 bit (32 bits per red, green, blue channel) mean-based (average) stack of the included frames.

Operations: Gaussian Blur	Gaussian blurs the stacked image.
Operations: Unsharp Mask	Unsharp masks the stacked image.
Operations: Laplacian Pyramid Sharpen	Laplacian sharpens the stacked image.
Operations: Fourier Transform Editor	Presents an interface in which you may paint out areas of the power spectrum that correspond to periodic noise artifacts.

Operations: Wavelet Shrinkage Denoise	Denoises the stacked image.
Operations: Adjust Histograms/Levels	Adjusts the levels of the stacked image.
Operations: Star Beautification	Performs deringing and debloating on stars.
Operations: Transform	Rotates and/or flips the stacked image.
Operations: Resample	Bins the stacked image 4 to 1 or 9 to 1. Increases the dynamic range by 4 or 9.

Threads: Synchronize feedback to thread progress	Specifies whether the cyan border and the Frame Inspector window reliably reflect the progress of multi-threaded tasks. With this option disabled, these forms of feedback may lag behind the actual progress of the task (the task may end before the cyan border has hilighted all selected clips). However, with this option enabled there may be a slight increase of overall time required to complete the task.
Threads: Number of threads	Sets the number of threads to use for parallel operations. Also lets you set whether or not to keep the display synchronized to thread progress during a threaded operation.

View: Small/Medium /Large Clips	Sets the size of the clips in the Clips Window.
View: Minimal Visual Feedback	Disables the cyan border around the clips and updates to the Frame Inspector Window during long, multi-clip tasks. This may speed the task up considerably, but has the cost of making it more difficult to monitor the progress of the task. However, the Working Dialog and associated progress bar will still be presented.
View: Channels	Specifies which channels are shown in the Stack Window.

CLIPS WINDOW COMMANDS

How to Perform the Command	Description of the Command
Mouse click/drag with shift key	Makes a selection of the clips for a subsequent operation. Selected clips are marked with a thin yellow outline.
Mouse click	Toggles stack inclusion on all selected clips (or the clip under the mouse if the selection is empty) (excluded clips have a red X over them).
Mouse click with option key	Completely removes any alignment on all selected clips (or the clip under the mouse if the selection is empty)
Mouse click with control key	Toggles darkframe subtraction on all selected clips (or the clip under the mouse if the selection is empty) (darkframe subtracted clips have a 'D' in the upper right corner).
Mouse click with control and shift keys	Toggles flat field division on all selected clips (or the clip under the mouse if the selection is empty) (flat field frame divided clips have an 'F' in the upper right corner).
Mouse click with command key	Performs any other command such that all clips are treated like the clip under the mouse. Otherwise each clip is processed separately (can have different effects if a selection of clips includes a mixture of settings). For example, if a selection has some clips stack-included, and others not, the default behavior when clicking on the selection is to toggle each clip individually. Holding down the command key will make all the selected clips match the clip that is clicked on.
Mouse click with '1', '2', or '3' keys	Autoaligns by difference the translation of all selected clips to the reference frame (the first clip). '1' autoaligns fastest if the translation is off by more than 15 pixels, '2' autoaligns fastest if the translation is off by 5 to 15 pixels, and '3' autoaligns fastest if the translation is off by fewer than 5 pixels. It doesn't matter too much which method you use, it's only a speed issue.
Mouse click with '4' key	Autaligns by centroid the translation of all selected clips. This method is less precise than the difference method and should be followed by a difference alignment, but might it might be faster to use this method first. It's up to you.
Mouse click with '5' key	Autaligns by cross-correlation the translation of all selected clips. Slower than centroid alignment, but it's there if you want it.
Mouse click with '6', '7', or '8' keys	Autoaligns by difference the rotation of all selected clips to the reference frame (the first clip). '6' autoaligns fastest if the alignment is off by more than 4 degrees, '7' autoaligns fastest if the rotation is off by 2 to 4 degrees, and '8' autoaligns fastest if the rotation is off by fewer than 1 or 2 degrees. It doesn't matter too much which method you use, it's only a speed issue.

'[', ']', '{', or '}' keys

If there are more clips than can fit in the Clips Window at its present size, these keys move back forth through the pages of clips. The square brackets, '[' and ']', move in half pages, and the curly brackets, '{' and '}' move in whole pages.

'R' key

Sets the clip under the mouse to the reference frame by switching places with the first frame.

'D' key

Permanantly deletes all selected clips (or the clip under the mouse if the selection is empty) from the workspace.

'N' key

Normalizes all selected clips (or the clip under the mouse if the selection is empty). Details are given in the section on planetary processing.

'M' key

Median filters all selected clips (or the clip under the mouse if the selection is empty). Should be used sparingly, with a low threshold. Primarily intended as a method to fill in hot pixel holes, not as a denoising technique (for which median filtering is commonly used). Denoising is better accomplished through stacking.

'B' key (uppercase)

Debayers all selected clips (or the clip under the mouse if the selection is empty). This is for use with RAW modded cameras that receive the Bayer matrix pattern from the camera. Different clips may require different Bayer patterns, so it is recommended to deBayer the reference frame manually, and then deBayer the rest to match it.

'b' key (lowercase)

Removes deBayering from all selected clips (or the clip under the mouse if the selection is empty).

arrow keys

Manually nudges all selected clips (or the clip under the mouse if the selection is empty) by one pixel.

arrow keys with shift and/or control and/or option keys

Manually nudges all selected clips (or the clip under the mouse if the selection is empty) by 5, 25, or 125 pixels. The number of modifier keys held down determines how large the nudge is. It doesn't matter which of the three modifier keys are used though (holding down shift, control, or option alone, will nudge by 5 pixels, any two of those keys will nudge by 25, and all three will nudge by 125).

'z', 'x' (lowercase)

Manually rotates all selected clips (or the clip under the mouse if the selection is empty) by .36 degrees.

'Z', 'X' (uppercase)

Manually rotates all selected clips (or the clip under the mouse if the selection is empty) by 3.6 degrees.

FRAME INSPECTOR WINDOW COMMANDS

How to Perform the Command	Description of the Command
Space key	Forces the Frame Inspector Window to display the reference frame. Otherwise the Frame Inspector Window displays the clip under the mouse.
Mouse click/drag	Defines an Operation Bounds Rectangle.
Mouse click/drag with control key	Defines an Operation Bounds Rectangle that is forced to be of power-of-2 dimensions.

FRAME STACK WINDOW COMMANDS

How to Perform the Command	Description of the Command
arrow keys with '1', '2', or '3' keys	Manually nudges the red, green and blue channels of the Frame Stack by one pixel.