Line Filter Image Reconstruction on One-Shot Color Cameras

Craig Stark, Ph.D.

Introduction

At one point or another, most amateur astrophotographers get bitten by the line filter bug. Causes of the infection vary. For some, it comes from seeing stunning black and white photographs taken through line filters or stunning tricolor composites. For others, it comes from realizing that, on the right target, line filters will let them peer through the urban skyglow halo that frustrates their attempts to produce better DSO pictures.

What are line filters?

Line filters, or narowband filters isolate and only allow to pass a very small area of the spectrum of light. For example, a Hydrogen-alpha (Ha or H-α) filter only allows a small window around 656.3nm to pass through the filter, blocking all other light. For this to be of any use, it assumes your target has a significant amount of signal at this wavelength and that your camera is sensitive to this wavelength. The first assumption pushes one to targets whose spectra show exceptionally strong signals at specific wavelengths and away from more broadband targets with their energy spread across a wide range of the spectrum. Emission nebulae are therefore very well-suited to line filter imaging, as the excited hydrogen gas resonates at specific frequencies corresponding to changes in its discrete energy level or state.

Since a large portion of the light we receive comes from a very narrow range, blocking all light not in this range can cut through light pollution while leaving the light from the target largely intact. Thus, even in the case of broadband light pollution, a line filter can effectively increase the signal-to-noise ratio in our image by hardly touching the signal (the nebula) but drastically hurting the noise (the light pollution). This same logic helps show why line filters are not useful for broadband targets like galaxies or star clusters. Since their light is spread out fairly broadly and evenly over the visible spectrum, isolating a specific frequency cuts down on the light one receives from them significantly – about as much as the reduction in light from skyglow.

Cameras for line filter work: Traditional wisdom

With the first assumption addressed by the choice of appropriate targets, we are faced with the second assumption – the sensitivity of our cameras to the emission line we wish to use. If you query the wisdom held on various Internet groups as to what camera to use for line filter work, you will get two broad suggestions. The first is to buy a monochrome camera and avoid one-shot color cameras. Once in the realm of monochrome cameras, you will be directed to those with better response in the deep red of H-α light. While other lines are very useful (e.g. O-III and H-β) these lie in the blue/green regions of the spectrum where all cameras generally have fine sensitivity. Thus, one looks at cameras with Sony ExView-HAD sensors or a number of the chips in the Kodak line.

What you will not hear anyone (or anyone sane) say is that if you want to do line filter work, you should buy a one-shot color camera. One-shot color cameras create full-color pictures in a single exposure. The sensor in these cameras is the same sensor as in monochrome cameras, but each pixel is covered by either a red, green, or blue color filter (some use he "subtractive" cyan, magenta, yellow, and green set). Through a bit of math called "Debayering" or "Demosaicing" (see my article Debayering Demystified in volume 2, issue 3) this is turned into a full-color image with red, green, and blue values at each pixel.
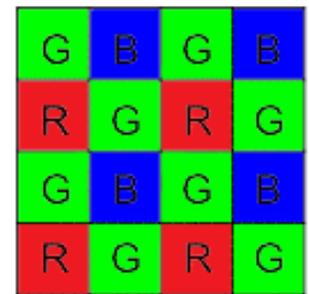

Figure 1: A typical Bayer Matrix for a small 4x4 area of the sensor. Each pixel has a single color filter letting each pixel code for red, green, or blue components of the image (during the debayer process, each pixel gets red, green, and blue components). When a line filter is also used, only some of the pixels will get any light (e.g., red pixels with a hydrogen-alpha filter).

It is this "Bayer pattern" or "Color Filter Array" (CFA) pattern (Figure 1) that is the source of the traditional wisdom. If we place an H-α filter over this array, the only pixels to get any light will be the red pixels. Thus, only 25% of the pixels get any light, their light is likely reduced by the imperfect filter (all filters have some loss), and the other 75% of the array is wasted and can only contribute noise into your image.

Side-Stepping Traditional Wisdom

There is no getting around the fact that one-shot color cameras will not be using all of their pixels to create an image when line filters are being used. The traditional wisdom is right and if you are purchasing a camera to do line filter work, you should look for a monochrome camera. But, what if you already have a one-shot color camera and want to do line filter work? Should you simply give up, not try it, and spend your nights window-shopping for a new monochrome camera? Or, are there things you can do to side-step at least part of the problem?

In truth, all is not lost when doing line-filter work on one-shot color cameras and often 75% isn't lost either. The case of H-α filters on RGB cameras is an example of the worst-case scenario. We'll return to this scenario later in the article, but first we turn to several brighter situations. Here, we see the responses of RGB and CMYG filters used in Sony one-shot color sensors (ICX285AQ and ICX429AK respectively) found in may color astronomy cameras (Figure 2).

In this RGB filter set, the O-III and H-beta lines (around 500 nm) are evenly covered by the blue and the green pixels. While neither is at their peak efficiency,
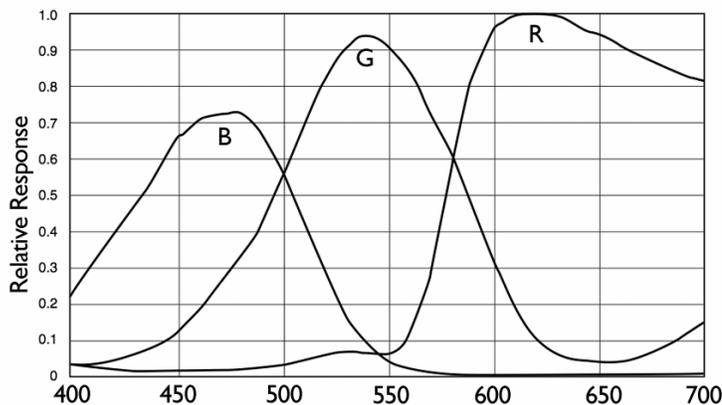
both are still passing significant amounts of light. This means that 75% of the pixels are getting valid data (recall that there are two green pixels for every red and blue pixel). A similar situation exists here for the CMYG arrays where all but the magenta pixels have valid data. The H-α filter situation for RGB arrays only activates 25% of the pixels, but for CMYG arrays, both the yellow and magenta filters are passing large amounts of light (50% of the pixels).

Specialized Reconstruction

Therefore, there are a number of situations in which the one-shot color cameras are operating at resolutions well-above the "half resolution" point (25% of pixels corresponds to half the resolution in each direction). If we ignore the pixels without valid data and just use the valid pixels as the basis for interpolation, we should be able to recover a very good image.

In Figure 3, we have the pattern of light and dark pixels we would expect to see on a CMYG array using a H-α filter. (the pattern of course repeats over the entire array) This is chosen here as it is a more difficult case than the 75% coverage of O-III or H-beta and because it is something typical of what users will face. These arrays are found in the Meade DSI color cameras, the Orion StarShoot camera, and in the Starlight Xpress M5C and M7C cameras.

If we consider Pixel 9, we see that there are 6 pixels (2, 4, 8, 10, 14, and 16) surrounding it that can be used as the basis for interpolation. Pixel 15 has 5 surrounding pixels and, in the worse case, pixel 22 has 4. Yet these 4 are those that are are used in the standard bilinear interpolation algorithm. Thus, we have a strong basis for interpolation of the missing values *continued* ⇨
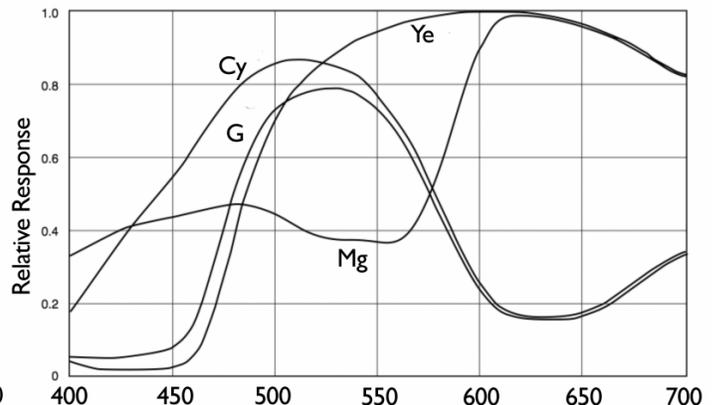

Figure 2: Response curves for each color-coded pixel type on common RGB (left; Sony ICX285AQ) and CMYG (right; Sony ICX429AK) one-shot color sensors used in many cameras. Note that at the 656nm wavelength of H-α is picked up by the red pixels on RGB arrays and the yellow and magenta pixels on CMYG arrays. The O-III and H-β lines around 500nm are picked up by the green and blue pixels on RGB arrays and the cyan, green, and yellow pixels on CMYG arrays.

1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30
31	32	33	34	35	36

Figure 3: Pattern of pixels that will get signal (white) and pixels that would not get signal (black) when a H-α filter is placed over a CMYG array. Specialized reconstruction techniques can be used to interpolate the missing values based on signal from nearby pixels.

and either normal, or slightly modified versions of normal interpolation techniques can be used.

In Figure 4 we have the result of four attempts to reconstruct a single frame of the Horsehead nebula taken on an Orion Starshoot camera (image courtesy of Michael Garvin). Each processing method was applied to the same individual raw frame and the same stretch was applied to each output. In the first panel, the fact that binning an image 2x2 will always remove the CFA mo-
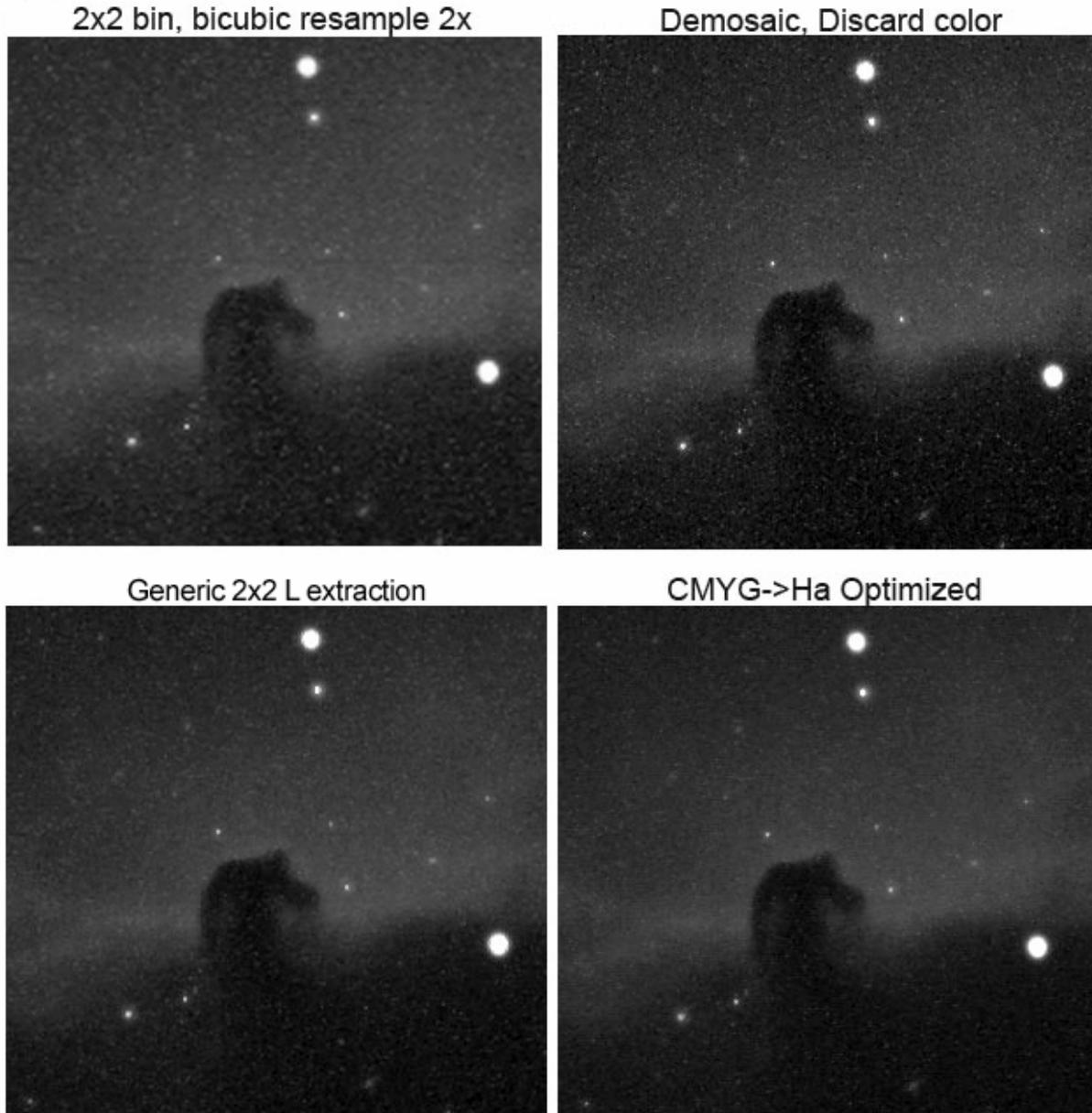
Figure 4 (CMYG_Ha_reconstruction.png): Results from various methods of reconstructing a single CMYG image of the Horsehead Nebula taken through a H-α filter. (upper left) Image binned 2x2 to remove the CFA (Color Filter Array) pattern and resampled 2x using bicubic interpolation. (upper right) Image debayered and color information discarded. Note the enhanced noise, made worse by the CMYG debayer process. (lower left) Image reconstructed using a generic technique to strip the CFA information while maintaining the full image size. (lower right) Image reconstructed using a technique designed specifically for CMYG + H-α reconstruction. (Image courtesy of Michael Garvin).

saic pattern was exploited. The image was binned 2x2 in Nebulosity (http://www.stark-labs.com) and interpolated in PhotoShop (http://www.adobe.com) using their bicubic algorithm to restore it to the original size. The image is quite soft. In the second panel, a common approach is taken by letting a demosaic algorithm attempt to deal with the image and then stripping off the color information. The image is sharper, but also noisier. The noise from adjacent pixels has entered into the image during the reconstruction process (this is especially problematic on CMYG arrays as red, green, and blue must each be calculated from the cyan, yellow, and magenta values).

The third and fourth panels represent algorithms designed for interpolation of one-shot color data with filters in place. On the bottom-left shows the results of a generic algorithm found in Nebulosity that can be used for any filter setup (UHC filters or specific line filters) on any color array. It is less noisy and more sharp than either of the top two panels. The bottom right is using the specific information about the color array and filter to reconstruct the image and, as a result, is slightly sharper and cleaner than the generic algorithm.

Worst-Case Scenario: 25% coverage

The worst-case scenario is unfortunately a common scenario. If an H-α filter is placed over an RGB array, only 25% of the pixels will get any light. To add insult to injury, the other 75% of the pixels will not be blank, but contain read noise, thermal noise, etc. This noise should not be allowed to enter the image if at all possible.

Here, we have three approaches that we can take. The first approach is to demosaic the image as if it were a normal color image and then to either discard all color information or extract only the red channel. When RGB arrays are used and only the red channel is kept, the results can be satisfactory if the demosaic algorithm is tolerant of having this much missing data. In particular, it must be tolerant of missing the green data. Constant hue algorithms that base everything on a relationship to the green intensity would be poor choices here as noise in the green channel (which is all noise, after all) will enter into the reconstruction of the red channel.

A second approach would be to use the "generic" algorithm noted above. While keeping a full-resolution image on the output, no attempt is made here to eliminate the noise from the unused channels and the image quality may suffer as a result.

The third approach is to face the fact that only every other pixel in each direction has data and to eliminate the unused pixels. One way to do this is to simply bin

the image 2x2. One pixel will dominate the average (the red pixel here) and effectively eliminate the other three pixels. Ideally, however, the unused pixels would be entirely ignored so that their noise would not be included in the output. Nebulosity provides a generalized version of this tool ("Low Noise Bin") that determines the noise-only elements of the image and performs a weighted-bin of the pixels with valid signal, ignoring these noise-only pixels. Similarly Iris (http://www.astrosurf.com/buil/us/iris) provides a SPLIT_CFA command for this purpose.

Recovering from Binning

Once binned, it is up to various resolution enhancement techniques to attempt to recover the lost resolution. One technique, Drizzle alignment was covered in a recent article (Stacking Images with Drizzle, volume 2, issue 2). Given that the images are now more undersampled, Drizzle can provide an excellent way to recover the lost resolution by using the inevitable fractional-pixel movement of the stars between your frames to accurately recover the resolution.

Drizzle requires a number of individual images to operate properly and therefore may not always be a viable option. A second technique is to simply resample the binned image (or stack of images) to be twice as large, restoring the original size. Image resampling is frowned on by some as "making up" data. Resampling techniques are all forms of interpolation and are attempting to come up with the best guess as to what an unknown pixel should look like. When doubling the image size, this equates to effectively shrinking the size of each pixel in the image and filling in the gaps then formed between these shrunken pixels. But, if you already have a one-shot color camera, you should already be used to the idea of interpolating your data (that's how you get red, green, and blue data at each pixel from red, green, or blue data at each pixel), so we can put this concern aside an evaluate just how well this can be done.

Many image resampling methods exist and each has its strengths and weaknesses. One technique may work well on some images but poorly on others while these may be well-handled by another technique. That said, there are several that I have found to be quite useful in up-sampling astrophotographs. They are the traditional Bicubic algorithm (found in many packages, including PhotoShop), the Catmull-Rom spline algorithm, and the Lanczos sinc algorithm. Catmull-Rom is in the same family as the typical bicubic filter and gives a nice balance of sharpness without serious ringing. Lanczos sinc interpolation is quite sharp but can be prone to ringing artifacts in very high contrast regions (and thus should not be used on images until they are stretched).
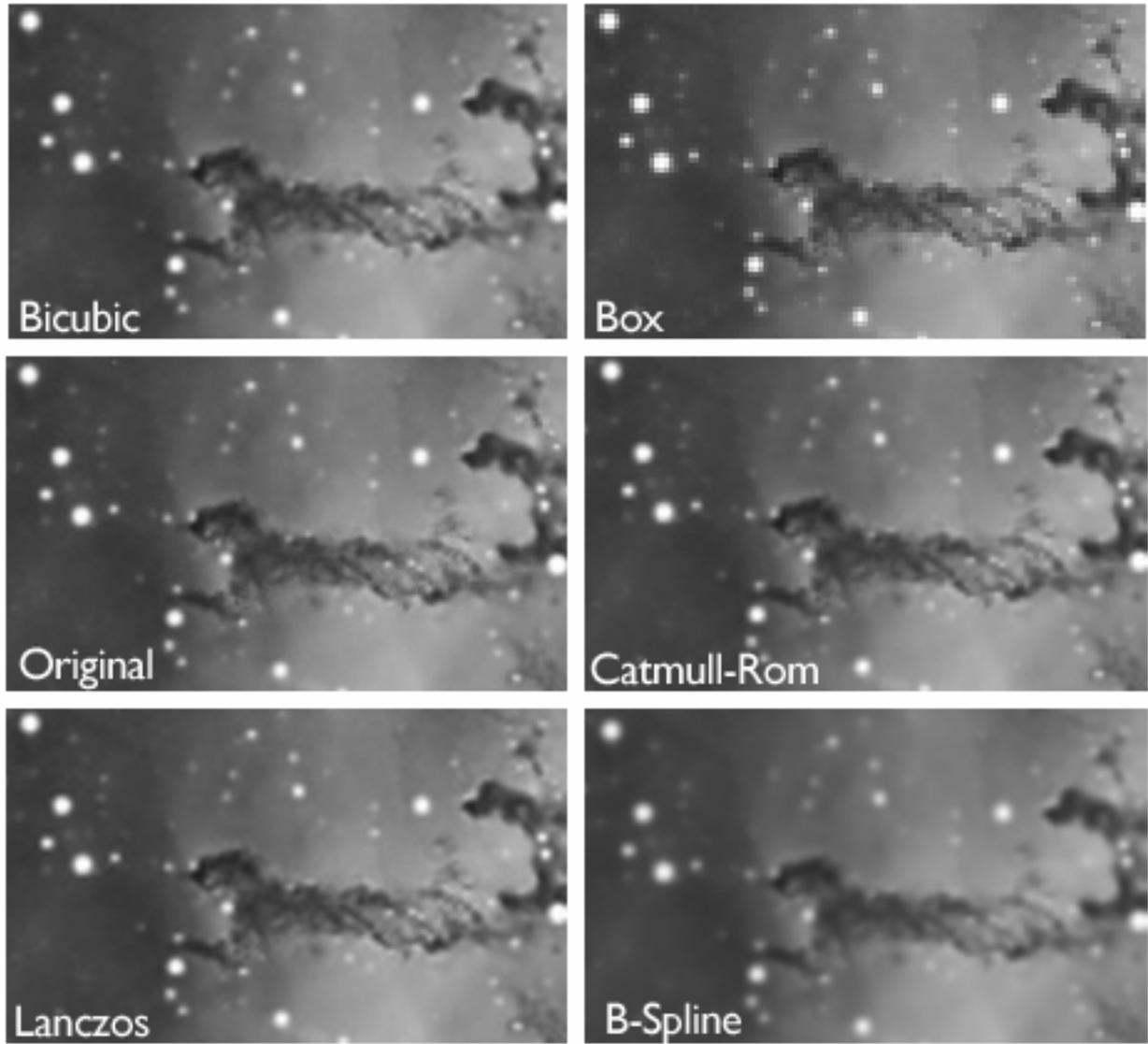
Figure 5: Small crops of an image of the Rosette nebula (see Figure 6; Image courtesy of Alan Chen) shown in the original form (middle left) and after 2x2 binning and re-interpolation to the original size. The ability of each algorithm to re-create the original image from the half-sized image can thus be directly compared. Bicubic interpolation was performed in PhotoShop and others were performed in Nebulosity.

Where ringing is a real concern, B-Spline algorithms can be used, but in practice I have yet to see results with B-Splines more accurately recreate the original image than one of the above.

Here, we have a test of how well the various algorithms can recover from a 2x2 bin of a stacked and processed image. The original image (a small crop of an image of the Rosette Nebula, courtesy of Alan Chen) is shown (middle, left) and was then binned 2x2. Each algorithm was used to resample the image at 200% in an attempt to recover the original image (performed in PhotoShop and Nebulosity). The de facto standard of PhotoShop's Bicubic algorithm is presented first (upper left) and does an admirable job at re-creating the original image from this half-sized input. Next is the Box (aka "Nearest neighbor") algorithm (upper-right), showing just how bad image resampling can be. Likewise, the B-Spline algorithm (lower-right) results in an overly smooth image. However, like the Bicubic, the Catmull-Rom (middle-right) and Lanczos (lower-left) both do excellent jobs at recreating the original image from the half-sized input.

Thus, the resolution lost by the 2x2 bin can be largely recovered. Note, this technique will work better on the processed and stretched image than on the un-processed image (as in Figure 4 above). Once processed and stretched, the contrast is greatly reduced, resulting in better performance of the resampling algorithms.

Conclusions

If you're looking to buy a camera and want to focus on line filter work, should you put one-shot color cameras at the top of your list? No. While one-shot color cameras have their real strengths, for the reasons laid out at the start of this article, their strengths run counter to the demands of line-filter imaging. If given a choice between line-filter imaging on the color and mono-chrome versions of the same camera, you would be foolish to choose the color version.

Rarely do we have such choices however and many astrophotographers have one-shot color cameras. I hope to have shown that line filter work is indeed possible on one-shot color cameras and that, with the proper techniques, the image degradation is not as bad as one might have assumed. If the article here does not do so, perhaps Figure 6, the full shot of the Rosette from Alan Chen will. As someone who lives beneath the urban skyglow halo and owns a modified Canon DSLR, a CCD Labs Q8-HR, and a Baader H-α filter, I certainly am glad that this is possible. ♦

By day, Craig Stark, Ph.D. is a professor of Cognitive Neuroscience at Johns Hopkins University. By night, he is an amateur astrophotographer and developer of software for astrophotography http://www.stark-labs.com

Figure 6: Image of the Rosette Nebula by Alan Chen captured on a one-shot color camera (Starlight Xpress M25C) using a H-α filter from mag 3.7 skies (http://www.heavenlyview.com). Alan's full color image was used for this issue's cover.