Seeing Stars

Stars are "just" points of light.
Tricky little devils. They show up every slip we make.

It is easy to take stars for granted and yet they pose some of the most difficult objects to process. The problem is that we instinctively know what a star should look like... or do we? Theoretically, all stars should be a single illuminated pixel on account of their distance and yet we accept the concept that brighter stars appear larger than dimmer ones. The question remains, how much is enough? Pictorially, stars can visually get in the way of the purpose of an image: consider a dim nebula in the Milky Way, the eye is distracted by the numerous bright punctuations and as a result, it is harder to distinguish the gaseous clouds within. In this case, some photographers go to the extreme of removing the stars altogether, while others leave them to bloat naturally with image stretching. I aim somewhere in the middle, keeping true to nature but trying to avoid them detracting from the image. In other images they are the "star" of the show and processing is optimized to show their individuality: color, size, definition and symmetry.

Star processing then is a complex matter, designed for the purpose in mind. The tools at our disposal can reduce star sizes, improve star shape, increase color, accentuate faint stars, remove stars altogether or blend star images with nebulosity image data from parallel workflows. This chapter looks at complex techniques such as deconvolution, other star-shrinking techniques, removing stars (to assist image processing), restoring star color and look at the essential supporting act of creating and using star masks. In a typical processing sequence, deconvolution is the first challenge and is probably the trickiest to get just right.

Deconvolution

This process is surrounded by some considerable mystique. This mathematical function is used both in signal processing and imaging in many disciplines. For astrophotographers, its aim is to undo the effects of the optical limitations set by the laws of diffraction, refraction, dispersion and minor tracking errors. These limitations convolve light, or in simple terms, blur it, reducing local contrast and resolution. The aim of deconvolution is to reverse these (in the case of the initial Hubble Space Telescope, it was used to compensate for its initial flawed mirror alignment). Although it is instinctive to think about the benefit to stars, deconvolution's magical properties equally apply to all fine structures. Deconvolution is not a panacea though; it is most effective on over-sampled images, that is, those taken with long focal lengths and small pixel sizes. This is because the math requires the optical smearing to occur across a block of pixels. I typically do not use is below 400-mm focal length, with my 5.4μm pixel camera.

Deconvolution is applied to linear luminance image data. In the case of an LRGB image, to the unprocessed, integrated luminance channel, or in the case of a CFA RGB image, to the luminance information contained within. (In the case of deconvolving color images, PixInsight requires implicit instruction that the image is linear since RGB camera image data is assumed to have a gamma setting of 2.2. This is done with the RGBWorkingSpace tool. Make sure the settings are the same as those in fig.1, with equal weights for the RGB channels and in particular, a linear gamma setting of 1.0.)

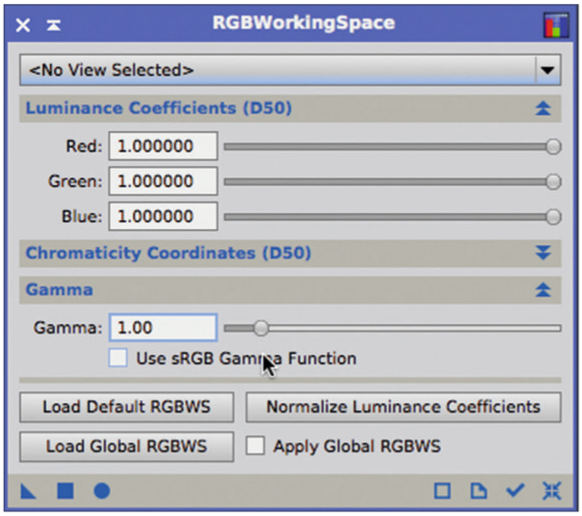


fig.1 For those of you who wish to deconvolve an RGB image, PI assumes it has a gamma of 2.2 until told otherwise. The settings above ensure the deconvolution works as predicted.

In the PixInsight implementation, as multiple variables affect the outcome, it can be quite tricky to find the best settings. That being said, my best results have been using the PI version, since it offers extensive facilities to tune or selectively apply deconvolution to different deep sky image types. In addition to my normal scouring of existing

fig.2–4 The Goldilocks dilemma: what is too soft, too sharp or just right? You may come to a different conclusion by viewing the page at different distances. These images differ in the deringing global dark setting, found in the deconvolution tool.

resources, I have approached this by considering several image scenarios and with a methodical approach; I have found an efficient way to deconvolve an image without extensive and cyclical experimentation. The clues lie in some forum tutorials and the layout of the tool itself.

Deconvolution Process Flow

The deconvolution tool has a number of settings laid out in the normal vertical arrangement. In practice, I found I could establish the settings in each section before proceeding to the next, with only the smallest amount of tweaking at the end. The trick is to know what to look for in the image at each stage to establish the correct setting. In essence, the setting flow is:

1 Set aside a duplicate luminance image and apply a medium stretch for use with the mask generation tools.
2 Measure the convolution effect with the point spreading function tool (DynamicPSF).
3 Disable Deringing and Wavelet Regularization options.
4 Establish a preview window that encompasses a range of image areas (background, different star brightness and nebulosity/galaxy).
5 Use this preview to experiment with tool settings, in this case, choose a value for Iterations to optimize the appearance of small dim stars, ignoring dark halos for the moment.
6 Enable Deringing and experiment with Global dark settings to almost entirely remove dark rings around dim stars.
7 Experiment with small values of Global bright (if necessary, to remove light artefacts around dark objects).
8 Create a mask for use with the Local deringing option.
9 Enable Local deringing, identify the Local deringing support file and experiment with Local amount to improve the appearance of bright stars and stars in bright regions.
10 Enable Wavelet Regularization and tune settings to establish the minimum noise reduction setting which removes the “curdling” of bright areas, such as bright nebula or galaxy cores.
11 Create a mask that protects stars and bright areas, invert it and apply to the main image.
12 Apply the Deconvolution tool to the main image and check the key areas of background noise, dim stars, bright stars and nebula/galaxy detail. Make small adjustments for fine-tuning.

Before starting, one needs to know what success looks like. Deconvolution is an imperfect process; it cannot reconstruct a perfect image but it can go a long way to improving it. In doing so, it will create other issues, increasing image noise, creating artefacts and unwanted halos around bright and dark objects. The settings are a compromise and although we may all agree on the more obvious issues, individual preferences define a wide selection of “acceptable” results that in addition, are also dependent upon the final reproduction scale and application. A range of potential candidates is shown in figs.2–4.

Preparation (1)

Before using the deconvolution function itself, it is necessary to complete some preparatory work for the deconvolution process, starting with the image. We stated at the beginning the deconvolution process is a benefit to stellar and non-stellar images. That being said, it is sometimes necessary to exclude it from operating on particular areas of the image that are otherwise featureless, but exhibit noise. Applying a deconvolution function to these areas makes matters worse. In other areas, differing amounts of deringing are required. Both cases require selectivity and these are achieved through the applications of masks at various points in the process. Forming a mask directly from an unprocessed linear image (star or range mask) is not an easy task. In both cases a mild image stretch increases local contrast where it is most needed by the mask tools. There are some additional ways to improve the robustness of star mask generation but for now, create two clones of the luminance image by dragging its tab onto the desktop. Next, open the HistogramTransformation tool and apply a mild stretch to both luminance clones, sufficient to see more stars and perhaps the first traces of a galaxy core or bright nebula. Give each clone a meaningful name; it helps when there are dozens of images on the desktop later on!

Table Parameter	Description
A	Amplitude of centroid, (0–1) Reject stars with A > 0.2 or A < 0.005
MAD	Mean Absolute Difference. Smaller is better. Reject those stars with big values.
r	Aspect ratio. A perfect circle = 1 Reject stars that have poor aspect ratio
theta	Angle of eccentric star axis. Check out the outliers and delete as required

fig.5 The DynamicPSF tool classifies stars with various parameters. The above parameters are the most useful to determine which of the sampled stars most reliably represent a PSF form.

Point Spread Function (2)

The deconvolution process starts in earnest with the supporting process of describing a Point Spread Function (PSF). This is a model of the effect of all those imperfections on a perfect point light source. Deconvolution is also used in microscopy and determining a PSF in this discipline is partly guesswork; astrophotographers, on the other hand, have the good fortune to routinely work with perfect light sources, stars, from which they can precisely measure the optical path characteristics rather than make educated guesses. PixInsight provides a specific tool, DynamicPSF, with which to measure selected stars to form a model.

The DynamicPSF process starts with a cropped linear image (before stretching) and before noise reduction too. After opening the DynamicPSF tool, apply a screen stretch to your image to show up the stars. Select up to 100 stars from all areas of the image, although it helps to avoid those in the extreme corners, where excessive field curvature may distort the results. At the same time, avoid saturated stars and the very tiny dim ones that just occupy a pixel or two. As you click on each star the tool analyses it for symmetry, amplitude and compares them statistically. Theoretically, each star should have the same PSF. Of course, this does not happen in practice and so the next step is to find a PSF that best describes them as a group. This is achieved with the Export synthetic PSF button (the little camera icon at the bottom).

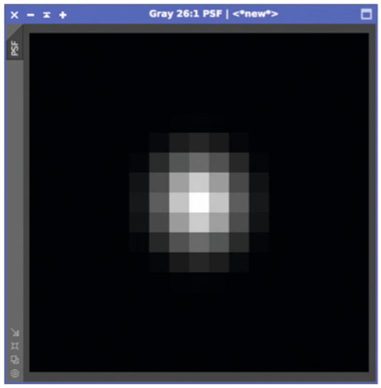


fig.6 The point spreading function describes the outcome of a point light source through the atmosphere, optics and sensor. It puts things into perspective, doesn't it?

Before hitting this button though, it is necessary to weed out those samples that do not fit in. To do this sort the DynamicPSF table using a few of the many criteria and remove those star entries that appear to be non-conforming. The table uses unusual acronyms for each criterion and the most useful are explained in fig.5. In turn select the Mean Absolute Deviation (MAD), Amplitude (A) and then eccentricity or aspect ratio (r). In the first case remove those stars that seem to have an excessively high MAD value and then remove the outliers for amplitude. I have seen some tutorials that propose to keep stars in the region 0.2–0.8. I found I had better results using dimmer stars and rejecting anything above 0.2. It is certainly worth trying out both approaches. Finally, remove any stars whose eccentricity is very different to the norm (which may be the result of double stars). If your tracking is known to be good, reject anything that shows poor eccentricity (for example r<0.80). Select the remaining stars and hit the Export synthetic PSF button. The result is an image of a blob, named PSF (fig.6). Although this appears particularly underwhelming, this blob describes what a singular point light source transforms into after it has passed through countless light years of interstellar matter, our atmosphere, your optics and

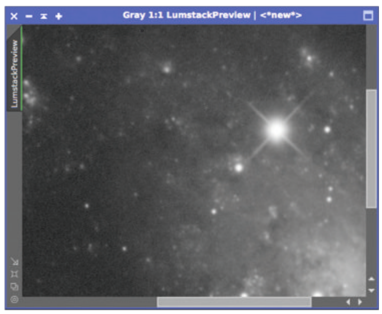


fig.7 A close up of the image preview showing a selection of stars for reference (with respect to the subsequent processing settings).

fig.8–10 From left to right, these three close-ups show the characteristic dark halos of a plain deconvolution, with dark side deringing and lastly deringing with local support. Note the background is "curdled" with respect to the starting image (fig.7)

triggered electrons on the CCD. It is pretty blurry… which explains a lot! Keep this image and close the DynamicPSF tool.

First Iteration (3–5)

Perform an automatic screen stretch to the luminance image and drag a preview that covers a range of sins, including background, faint stars, bright stars and brighter areas (preferably with some bright stars too). The initial trial runs will be on this preview. Some of the deconvolution processes use image properties to alter their behavior and if the preview is not representative of the image as a whole, the final application to the full image will produce a different result. Open the Deconvolution tool and disable the Deringing and Wavelet Regularization options. Choose External PSF and select the PSF file created earlier. For the algorithm, choose Regularized Richardson-Lucy and start with the default 20 iterations. Apply Deconvolution to the preview and compare the results for 10–50 iterations.

As the iterations accumulate so does the sharpening affect increase and the accumulation of artefacts. In addition to the halos around bright stars, those areas with the lowest SNR start to "curdle" and then progressively the brighter regions with better SNRs do too. This curdling is objectionable and requires further treatment (simply put, blurring) to remove the effect. That is fine when it is an area of blank sky but is a problem when it is a galaxy core in which one wants to preserve detail. I normally increase the number of iterations to the onset of this curdling in the bright areas (fig.8). At the same time, check the Process Console for warning messages on divergence (going the wrong way). This may be indicative of a poor PSF description or too many iterations.

The results are messy at first but for the moment the aim is to improve the smaller stars, checking they are tighter and more distinct (even if they have a dark halo) and in addition, that any small-scale patterns within bright areas are more distinct. At this point in the process, it establishes a general setting that can be revisited later on in the final round-up.

Deringing (6–7)

Almost inevitably, dark rings, as shown in the fig.8 will surround most stars. These artefacts are an unavoidable consequence of the sharpening process. Fortunately, they can be removed by progressively replacing ringing artefacts with original pixel values. These artefacts are associated with dark and bright object boundaries and are tuned out by changing the values for Global dark and light in the Deconvolution tool. The sliders are sensitive and I type in the values I need, to two significant figures. Of the two artefacts, dark rings are the most obvious and many images may only require an adjustment to the Global dark setting. Each image is unique, however, but I often find optimum values in the range of 0.02–0.06. For the Global light setting, I may use an even smaller amount, if at all, around 0.01, to remove bright artefacts.

To decide upon the setting, change the dark setting so that small stars just lose their dark halo, as in fig.9, or can just be perceived (to improve apparent sharpness). If you overdo the Global light setting, it negates out the effect of the deconvolution. Flip the preview back and forth to check the overall change. In some cases, this level of deringing will suffice for the entire image. Bright stars and stars over brighter areas may need more help though. You can see from the figures that they have a hard core and alternating rings. These are addressed by using the Local deringing option in the tool and for this it needs to be selective, using a form of mask.

Local Deringing and Star Masks (8–9)

The Local deringing option addresses the ringing around the brighter stars (fig.10). It does this by limiting the growth of artefacts at each iteration of the deconvolution

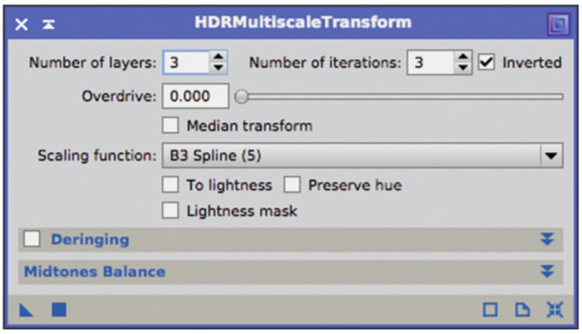


fig.11 The above settings in the HDRMT tool do a good job of evening out background levels to help with reliable star detection.

fig.12 The entire preview image after HDRMT application on the non-linear (stretched) image effectively removes background levels to improve threshold star detection.

algorithm. In the case of a deep sky image, it does this selectively using something resembling an optimized star mask for Local support. Mask generation appears straightforward enough but generating a good Local support file requires care. There is no one do-it-all star mask and since there are a number of ways of creating a star mask it is worth comparing some common methods:

Star Masks (An Aside)

There is a world beyond the StarMask tool to produce a decent star mask. By itself, it can be tricky to use on some images, on account of altering background levels and a wide difference in star intensity and sizes. Mask-building skills are worth acquiring though; they come in handy during many processes as well as deconvolution. Although the tool can be used on unmolested linear and non-linear images, in practice, it is easier to use on a stretched image. Some were produced earlier in step 1 and there is nothing to prevent one applying the StarMask tool to these images. There are some things, however, that help the StarMask tool achieve a better result. Star images are small blobs that are lighter than their surroundings. Two techniques help discriminate stars, even-out fluctuating background levels and distinguish star-sized objects from noise (at a smaller scale) and bright objects (at a larger scale). There are several ways to do this; two common techniques use the HDRMultiscaleTransform (HDRMT) or MultiscaleMedianTransform (MMT) tools.

In the first case, apply an HDRMT to the stretched clone image to flatten the background (large scale areas) and leave the stars alone. For this, start with the default settings and experiment with it set to several layers and iterations (fig.11–12). Carefully measure the background level and use this for the StarMask tool's Noise threshold value.

In the second, the stars are isolated by using scale, rather than brightness as the key, by applying the MMT tool to the stretched duplicate image. In this case, from the default setting, increase the layers value to 5 or 6 and disable the first and residual scales (fig.13–14).

Both isolate star-like objects and yet in both cases the resulting image may contain elements of non-stellar material that can be interpreted as stars. In most cases a slight adjustment to the black levels with the HistogramTransformation tool will clip faint traces and as a last resort, the CloneStamp tool may be applied too. (Sometimes this is the most expedient way to deal with a bright galaxy or comet core.)

The StarMask tool has a number of parameters that require some explanation. The simplest is the Noise threshold. This defines a baseline between the background level and the faintest stars you wish to mask. If this is set too high, some stars will not be detected, too low and noise may be interpreted as stars. Select a dim star with the mouse and press the left mouse button. A magnified cursor appears with a convenient readout. The working mode is normally left at the default (Star Mask) and the Scale parameter set to an upper star-size limit. Too small and it will miss the big bright stars altogether, too large and it may include non-stellar objects. There are cases when one scale does not fit all and it is then necessary to create several star masks, using different Scale and Noise threshold settings optimized for small stars and heavyweight ones, and then combine the masks with PixelMath using an equation of the form:

max(mask1,mask2)

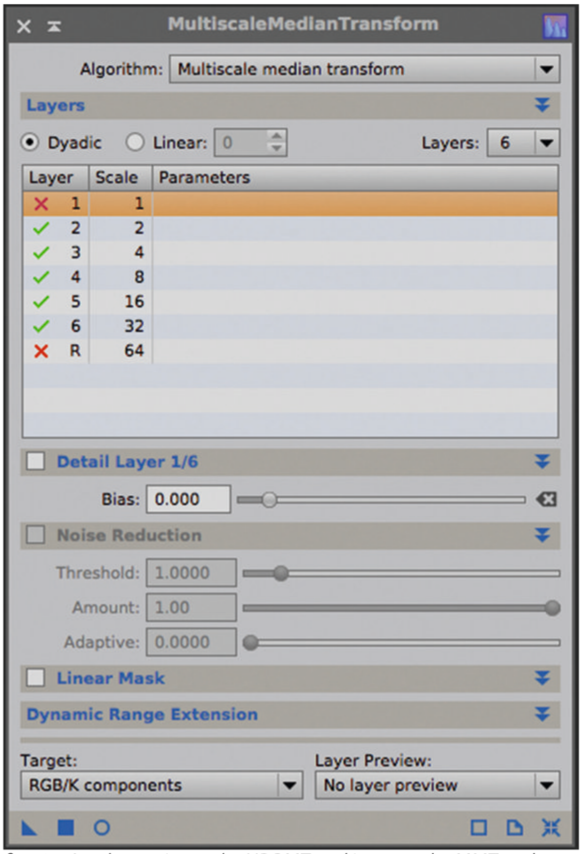


fig.13 An alternative to the HDRMT tool is to use the MMT tool, to remove noise and large scale objects, to help the StarMask tool to discriminate more effectively.

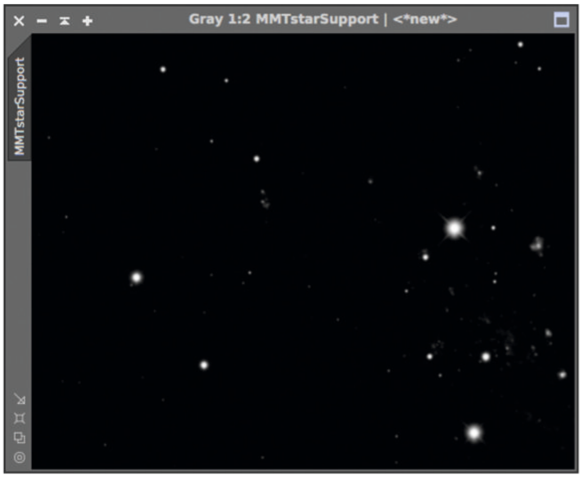


fig.14 The resulting image from the MMT application prior to using the StarMask tool on it, shown here with a mild stretch and shadow clipping for printing purposes.

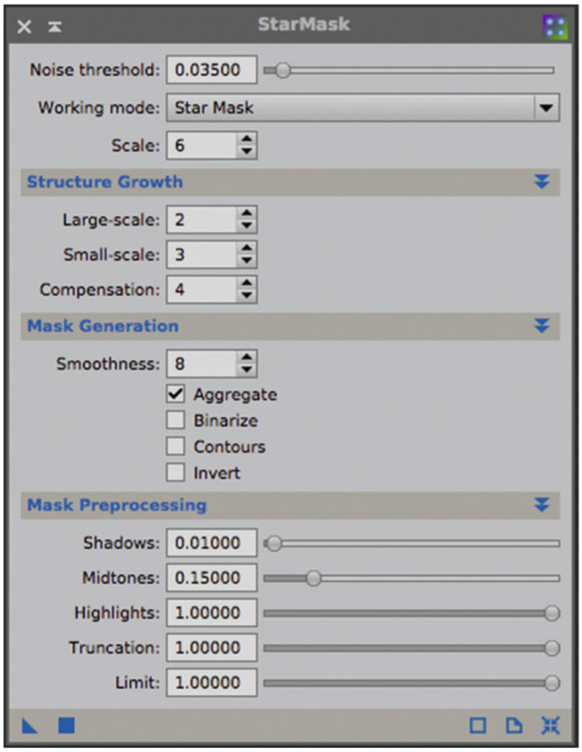


fig.15 These StarMask settings were used on the HDRMT processed image to form the Local support file (fig.16).

The output of the StarMask tool is a non-linear stretched image and it is often the case that it appears to have missed many of the smaller stars at first glance. This may not be the case; it is just they are not white in the mask but dark grey. Jumping ahead, the Mask Preprocessing section has a series of mask stretch and clipping tools. The Mid-tones slider performs a basic non-linear stretch. Decreasing its value boosts faint detail in the mask. Values around 0.01–0.1 will significantly boost protection on the fainter stars.

The Structure Growth section can be confusing at first since it appears to have two controls for small stars and interact with the Mask Generation settings too. There are several benefits to growing structures; the first being one often needs to process a star as well as its diffuse periphery beyond its distinct core. Growing a structure extends the boundary to encompass more star flux. Another reason is to do with the smoothness option; this blurs the mask and lowers protection on the star side of the mask boundary. A growth of the mask before smoothing ensures this erosion does not encroach into the star flux. The two top controls change the mask boundaries for Large and Small stars and in practice, I use the Compensation setting as a fine tune to the Small-scale adjustment.

In the Mask generation section there are four controls that alter the mask appearance: Aggregate's pop-up description is not the easiest to work out. When it is

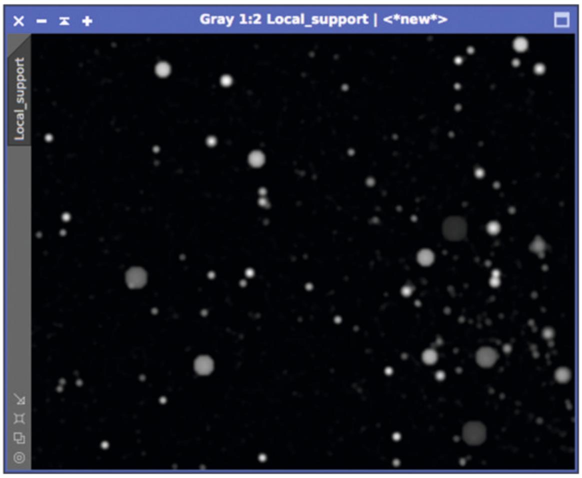


fig.16 The Local support file. Note that the largest stars in this case are only partially masked. That can be changed by enabling the Binarize option in the StarMask settings.

enabled, big bright stars do not appear in the mask as a uniform mid-grey blob but with less intensity and shading towards the edges. This gives a natural feathering of the mask and can be useful during the deconvolution process on big bright stars. The Binarize option is the opposite and creates a black and white mask, with hard edges, no mid-tones and is not suitable. It has its uses though in other processes but can throw unexpected results. If the Noise threshold is set too low every noisy pixel turns white in the mask. To avoid this, increase the Noise threshold (to about 10x the background value) and optimize the mask with small adjustments to the Noise threshold. When Aggregate and Binarize are used together, fewer "stars" are detected and the larger stars are rendered smaller in the mask, on account of the shading. If I do enable Binarize, I enable Aggregate too as I find the combination is less susceptible to small-scale noise.

After a little experimentation on the preview, I chose the settings in fig.15, which produced my Local support image (fig.16). Select this file in the deconvolution Deringing settings and move the Local amount slider fine to tune the correction. This slider blends the original image with the deconvoluted one. Choose a setting that leaves behind the faintest dark ring around bright stars (fig.10) as the next step also reduces ringing to some extent.

Wavelet Regularization (10)

The controls in this section of the deconvolution tool look suspiciously like those in the noise reduction tools, and for good reason; their inclusion is to counter the curdling effect caused by the deconvolution process trying its best on noisy pixels. The noise reduction level for each scale is set by a noise threshold and reduction amount and just as with noise reduction settings, the strongest settings are at the lowest scale. The Wavelet layers setting determines the number of scales. I normally set it to 3 or 4 and proportionally scale back the larger scale noise thresholds and reduction amount. I choose a setting that restores the appearance of the brighter areas of the image, in this example, the main part of the galaxy.

The brighter areas of the image have a high signal to noise ratio and the wavelet regularization settings required to remedy these areas are less severe than those required to smooth the dark sky background. Conversely, I find that the noise reduction settings to fix the background appearance soften the appearance of the brighter areas. For this reason, I optimize for the bright areas and mask off the darkest regions of the image before applying the final deconvolution settings.

Background Combination Masking (11)

The mask for the background is a combination of a range and star mask. In this example, I experimented using the same star-based mask used for Local support and some other derivations. I increased the scale one notch to identify the biggest stars and stretched the mask to boost small star protection.

The range mask tool seems easy enough with its preview tool. Well, yes and no. If the background has a sky gradient, it may prove troublesome. In this case duplicate the luminance channel and use the DynamicBackgroundExtraction tool to flatten the background before using the RangeSelection tool. Deselect Invert and choose a threshold that excludes featureless background but leaves behind interesting structures. You can feather and smooth the selection too, to alter the boundaries. Feather and smooth have different effects; with both at zero, a simple black/white mask is generated according to the limit sliders. Increasing the feather slider selects pixels proportionally based on their value, whereas the smooth slider blurs the mask. Even so, this mask will exclude some stars that reside in an otherwise featureless sky and it is necessary to unprotect these areas. The common method is to create a star mask and then combine it with the range mask to create a mask that protects the background, using a simple PixelMath equation in the form:

max(rangemask, starmask)

Ironically, the very brightest stars may not appreciate being deconvoluted and may exhibit weird artefacts. In

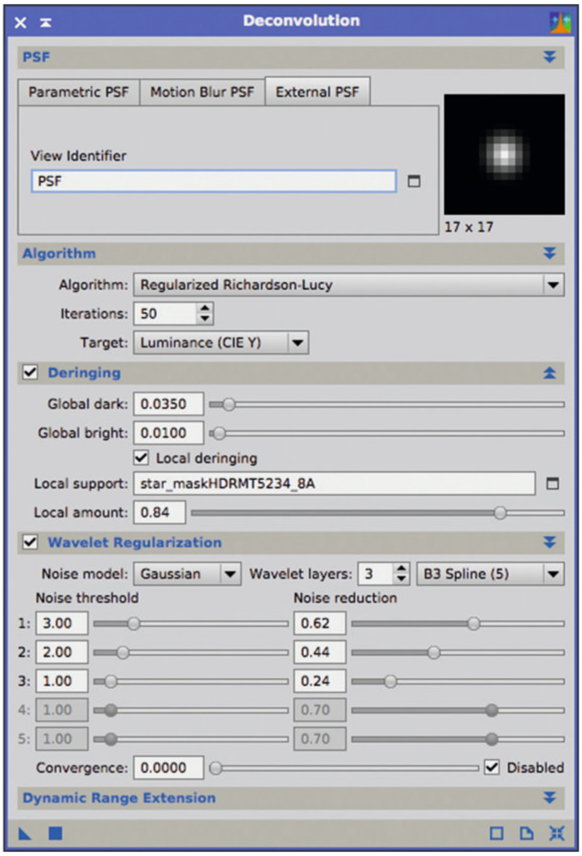


fig.17 The final deconvolution settings that were applied to fig.18, in combination with a mask protecting empty background, produced the final result in fig.19.

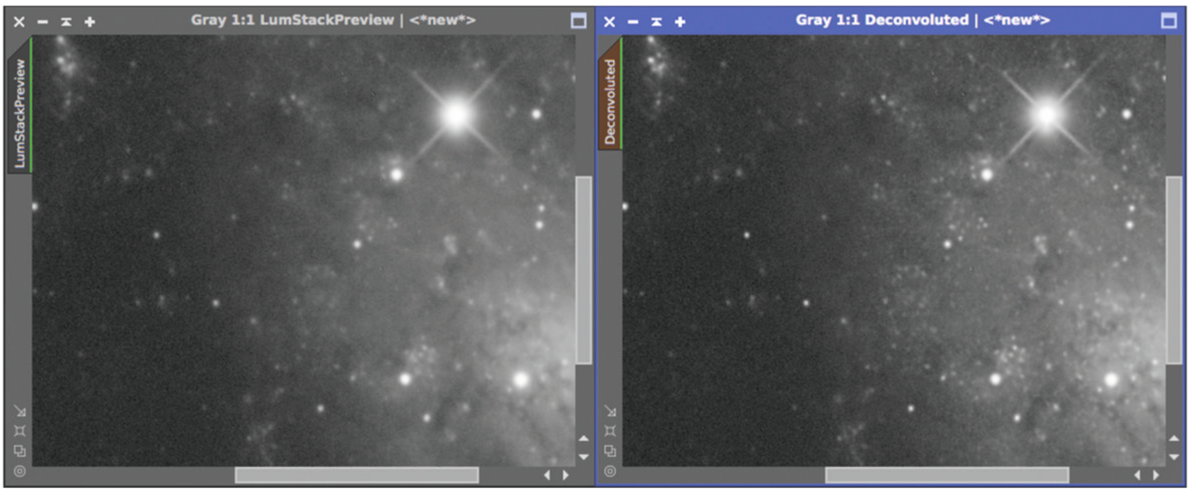
these cases, it may be necessary to further alter this combination mask by creating a unique star mask with a very high noise threshold that just selects the clipped stars and then subtract this from the combination mask using Pixelmath. The combination mask will look peculiar but it will do its job.

Final Tuning (12)

Generate the range mask and Local support image from the full frame image using the settings you settled upon in the preview trials. Generate the full frame star mask and combination mask too and apply this to the luminance image. Select your full frame Local support image in the Deconvolution tool and keeping all the other settings as they were, apply to the full frame. The result may differ from that of the preview (depending on how well you chose the preview area) and so a small amount of tuning is in order. Knowing what each tool setting does makes it much easier to make those small adjustments. I found my deringing tools and noise reduction settings were still good and I just increased the number of iterations to see how far I could go. In this case I increased the iterations from 20 to 50. This control has diminishing returns and with 50 iterations only mild further sharpening was apparent and without any artefacts.

Life After Deconvolution

Deconvolution is not the last word in star processing; although it yields a modest effect on small structures it struggles with large bloated stars and even nicely sharpened stars can be mutilated by extreme stretching (for example, those encountered during narrowband image processing).



figs.18, 19 The original image stack is shown on the left (magnified) and the deconvoluted version on the right. The differences are subtle and not "obvious", which is a sign of good editing judgement. A more heavy-handed approach can cause unsightly artefacts that outweigh the benefits of deconvolution.

In these cases, there are techniques that can shrink all star sizes, even large ones, or remove them altogether. In the latter case, some find it helpful to process nebulous clouds in the absence of stars and then add them back in later on. The star processing is done separately and has a less aggressive non-linear stretch. This avoids highlight clipping and preserves color saturation.

Morphological Transformation

An alternative to deconvolution is morphological transformation (MT). This tool can appear to change the size and emphasis of stars within an image or remove them altogether. Its tool settings alter the amount and shape of the transformation by iteratively replacing pixels with a statistical combination of its neighbors and blending them with the original image. (The tool is very flexible and it can potentially make an asymmetrical transformation to compensate for elongated stars.) The tool is applied to an image in combination with a mask to confine the effect. To remove stars altogether, apply iteratively until the stars have shrunk to a few pixels and then blend these pixels with its neighbors within the star mask's holes. I use this tool on the stretched (non-linear) image and after it has received some noise reduction. Excessive noise interferes with star mask generation and reacts to any image sharpening too.

Reducing Star Sizes using MT

Star masks have already been discussed at some length. As before, use HDRMultiscaleTransform on a duplicate stretched image to even out the background. In the StarMask tool, set the background level so it only identifies stars and the scale to identify the stars you wish to shrink. The MT tool blends each pixel with the median of the pixels within its defined boundary (the Structuring Element). If one uses a simple star mask, this will also cause the central core pixel value to lower too. To just shrink the star edges select the star peripheries with the mask, using the StarMask tool, but this time, with very different settings. In fig.20 the Structure Growth is reduced to minimal levels, as is the Smoothness parameter. This confines the mask and prevents even small stars being fully selected. In the Mask Generation section select the Contours option. Finally, change the Mid-tones setting to about 0.1 to boost the mask's contrast. When applied to our prepared image, it produces the star mask shown in Fig.21. On closer inspection, each star mask is a tiny donut that marks the stars diffuse boundary. If during this process it is impossible to create a perfect mask from one application, create a range of star masks, optimized for stars of different scales and intensities and then combine

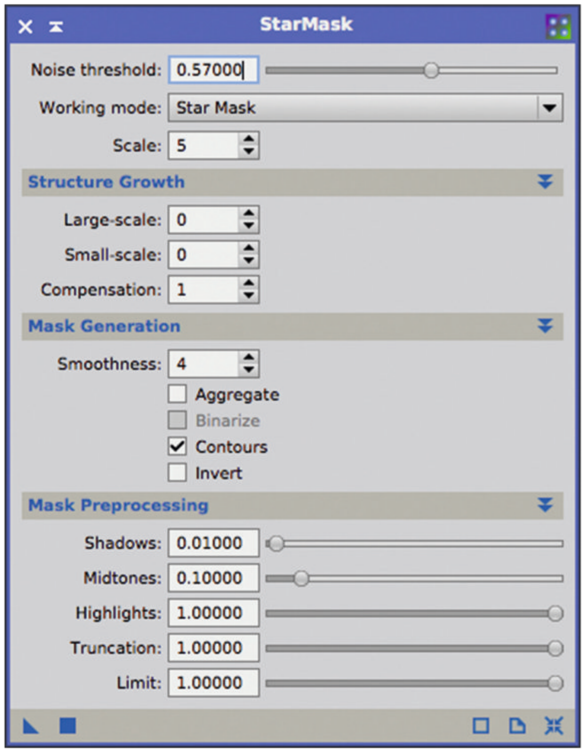


fig.20 These StarMask settings identify a star's periphery, rather than the whole. The low growth settings ensure a thin annulus mask is generated around each (fig.21). Higher growth settings would "fill-in" the donuts, with the risk of star removal during the MT application.

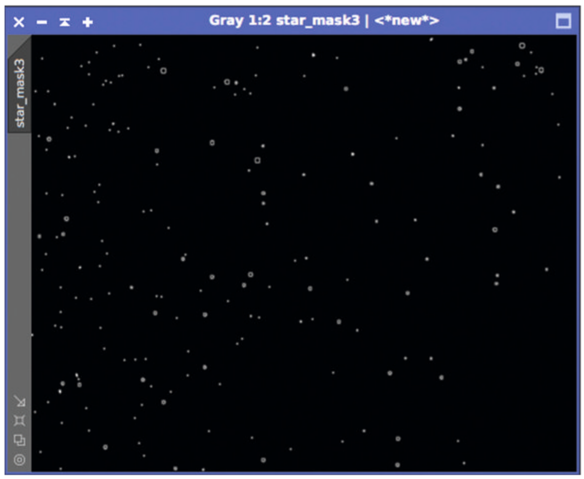


fig.21 This star mask, using the Contours option in StarMask (as shown in fig.20), protects star cores and restricts manipulations to the diffuse star boundaries.

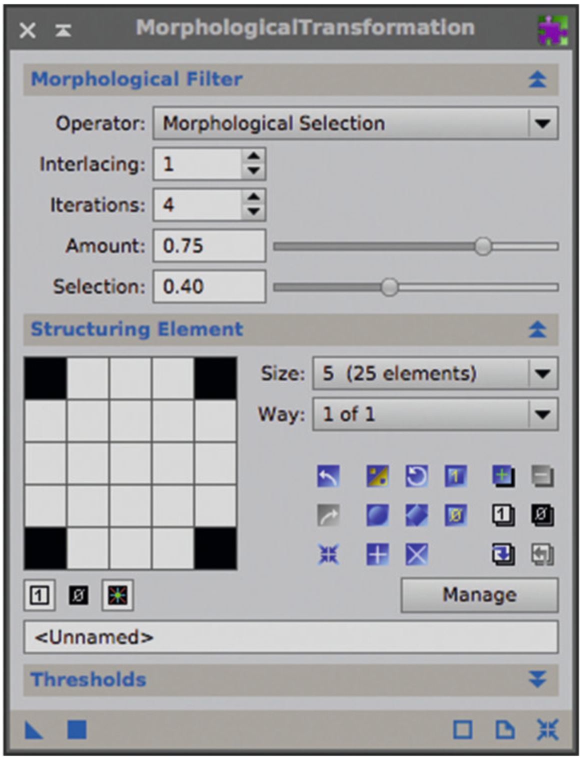


fig.22 Typical MorphologicalTransformation (MT) tool settings for reducing star sizes, identified using the star mask in fig.21.

these using a simple PixelMath equation (as above) to generate a single mask. Apply the final star mask to the image.

The MT tool has a number of modes (operations): erosion shrinks a star, dilation expands a star and Morphological Selection combines both erosion and dilation. This last mode produces a smoother overall result than erosion on its own. The Selection parameter defines the ratio of the two operations. Low values (<0.5) shrink the star and high values (>0.5) enlarge it. The Amount parameter blends the transformed image with the original image; when set to 1 there is no blending and for a more natural result try a modest blend in the region of 30–10% (0.7–0.9). The MT tool is more effective when a mild setting is applied iteratively; try 2–5 iterations with a mild erosion setting. The last group of settings concerns the Structuring element. This defines the scope of the median calculation for each pixel. In this case, for small and medium stars, choose a circular pattern with 3x3 or 5x5 elements.

Apply the MT tool to the image or preview and evaluate the result. If the mask and settings are correct, the smallest stars are unaffected but the larger stars are smaller and less intense. In an image of a diffuse nebula, this may be desirable as it places more emphasis on the cloud structure. If, however, you only wish to reduce star sizes and not their intensity, applying some image sharpening restores normality. This again employs a mask to select the image's small-scale structures and then these are emphasized by applying the MultiscaleMedianTransform (MMT) tool.

In practice, take the prepared stretched image that has HDRMT applied to it and use it to create star mask (with the scale set to 2 or 3) or apply the MMT tool to it, (disable all but the two smallest scales). With the mask in place, emphasize the star intensity by applying the MMT tool, only this time using its default settings and

fig.23 The image above is a magnified portion of the stretched deconvoluted image.

fig.24 The image above has had further star reduction applied to it, using the MT tool and some mild sharpening (using the MMT tool) to recover the peak star values.

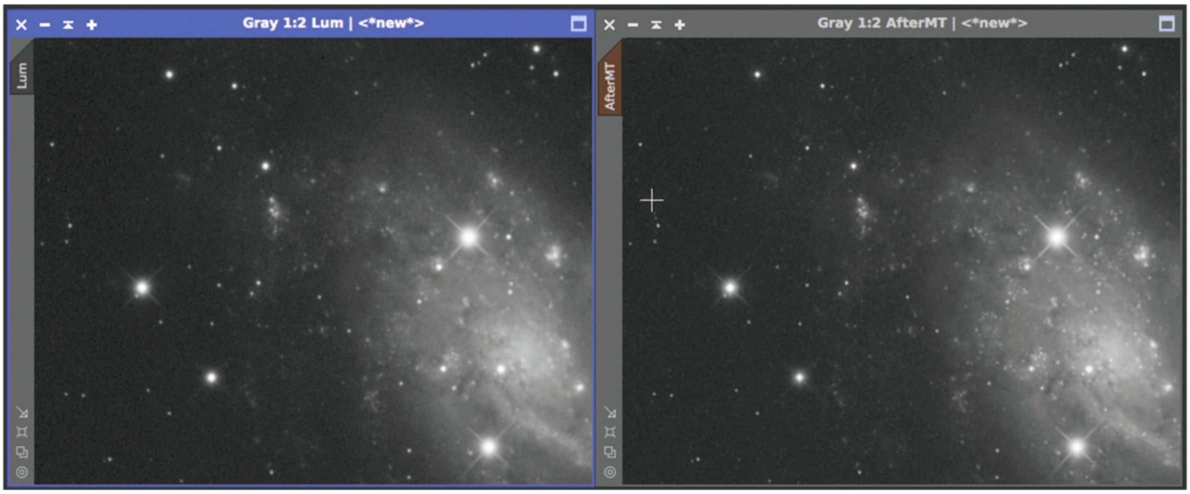


fig.25 A final comparison of a stretched luminance image with that of one that has been deconvoluted and had further star-size reduction. The difference is subtle but worthwhile.

with an increase in the Bias setting of the two smallest scales. Use the real-time preview function to try out different bias settings; depending on the mask intensity, it may require a large bias value to make a noticeable difference. Watch out for noise; increasing the bias of small scales is the opposite of noise reduction. If the mask does not obscure hot pixels and high noise levels, the MMT application will create havoc. The net outcome of all this, using the MT tool and some mild sharpening, is shown in fig.22 and fig.23. The deconvoluted image on the left has had a standard non-linear stretch applied to it and the one on the right has had further star size reduction with MT and MMT treatment.

Removing Stars using MT

The same set of tools can be used to shrink stars to oblivion (the Vogans would be impressed). In this case, the MT tool is repeatedly applied until the stars disappear, though with different settings. In the first step, ensure the star mask is only selecting stars and remove any remaining large-scale elements from the preliminary star mask. One effective method applies the MultiscaleMedianTransform tool to the star mask, with its residual layer setting disabled. Stretch the mask using the HistogramTransformation tool and at the same time, gently clip the shadow slider to remove faint traces of non-stellar imagery. Repeat to discriminate and boost the mask in favor of the stars.

To remove rather than shrink stars uses more brutal settings in the MT tool. Select the erosion operation with the Iterations and Amount set to 1. Even so, it will take several applications to remove the stars and even then, it may leave behind curious diffuse blobs. In the case of working on colored images, the process sometimes produces colored artefacts too. If this occurs, undo the last MT application, apply a modest MT dose of dilation

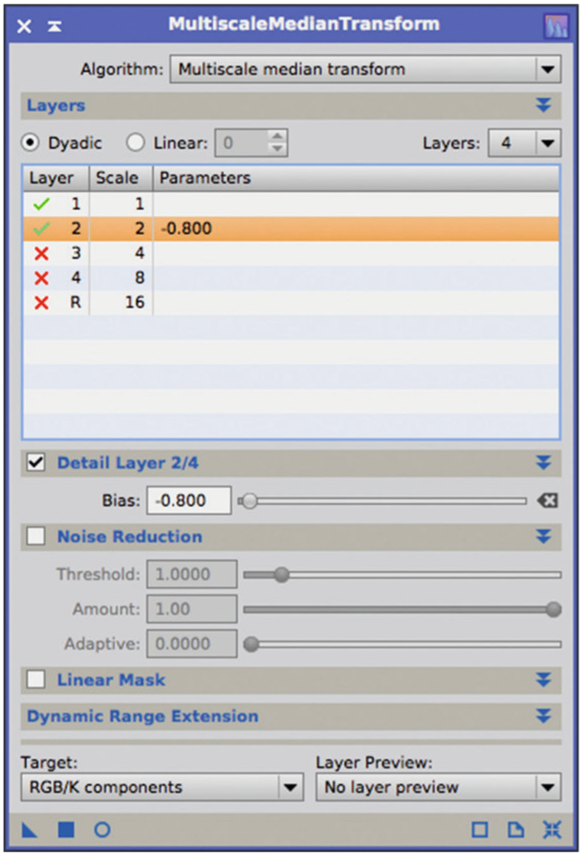


fig.26 These MMT tool settings generate a mask to select small structures and is an alternative to using the StarMask tool to select small stars for sharpening after shrinking.

and then try the MT tool (set to erosion) once more. Even so, some stars may stubbornly refuse to be scrubbed out, even after several applications of the MT tool. One method to disguise the remaining blip is to smooth the image (with the star mask still in place) using the MMT tool, only this time, enable the residual and larger scales and disable the smaller ones. The largest stars will still leave their mark, however, especially if they have diffraction spikes and as a last resort, even though it is heresy, use the CloneStamp tool to blend out the offending blobs.

Improving Star Color

Generating good star color is deceptively simple and in reality is a significant challenge in its own right. It appears that almost every action conspires to destroy it and to create and keep it requires special attention from image capture through to both luminance and RGB image processing. It can evaporate in a single step; for instance, if one has a pure red star and mix it with a high luminance value (>90%), the result is a white star. Similarly, if the RGB channel values are at maximum and mixed with a mid-tone luminance, you will get grey. Good color, therefore, requires two things: differentiation between the RGB values, coupled with a modest luminance value.

Image Capture Strategies for Star Color

Sub-frame exposures try to satisfy two opposing demands: sufficiently long to achieve a good SNR and capture faint detail, yet short enough to avoid clipping highlight values. It is rare to find a single setting that satisfies both. The solution is most easily met by taking a long and short exposure set; each optimized for a singular purpose and then combine the integrated images later on. This is easily done in the acquisition software's sequence settings by creating two distinct image subframe events for a filter. I normally create an LRGB sequence with long and short luminance exposures designed for nebulous clouds and bright stars/galaxy cores respectively. I choose a subframe exposure for the color channels that does not clip the bright stars or a few at most. Some subjects will never cooperate; Alnitak close to the Horsehead nebula is a beast that will not be tamed.

The concept of exposing a unique set of color subframes will also put natural star color into a narrowband image; a narrowband sequence typically consists of 10- to 20-minute subframe exposures and on their own produce oddly colored or clipped white stars. By including a few hours of RGB data into the imaging sequence (using short subframe exposures) the separate colorful star image is overlaid to good effect. This technique is explained in the C27 Crescent Nebula practical assignment.

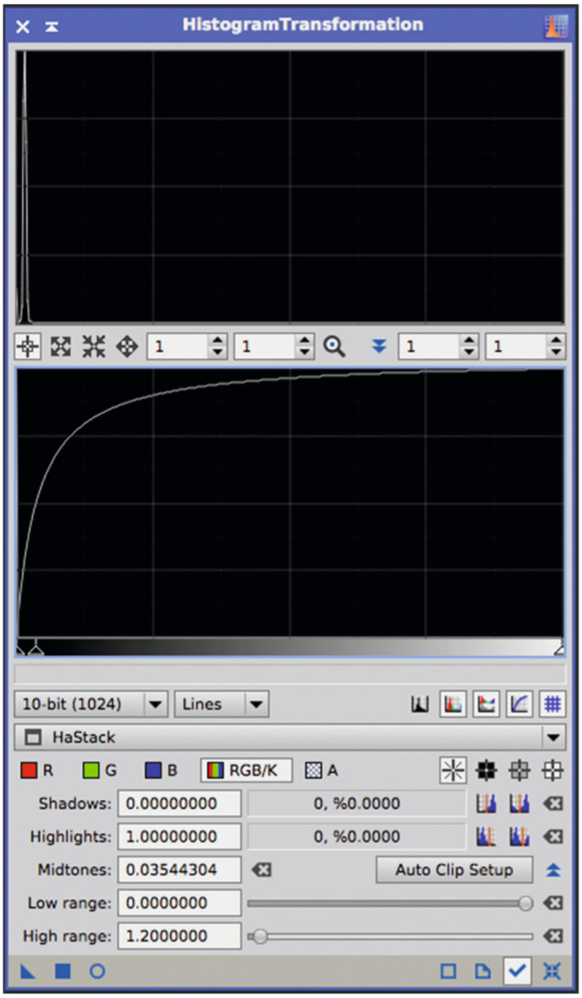


fig.27 A mild stretch before applying the MaskedStretch tool, using an extended high range setting, helps keep star peak intensities in the range of 0.8–0.95 and looking natural.

These are all excellent starting points but even so, a few stars may still clip. Really bright stars become large diffuse blobs when stretched, on account of diffusion and diffraction along the optical path. It does not really help to combine subframes of different exposure length either, as the lower-intensity diffuse boundary picks up color but the central core stubbornly remains near-white. This remaining obstacle, creating a realistic appearance to really bright stars, is possibly the largest challenge in an image and more drastic means are needed during image processing to tame these bloaters. (You didn't hear me say it, but Photoshop or Gimp is also quite useful for isolated edits, post PI editing.)

Image Processing Strategies for Star Color

Our two mantras during image processing are to stretch the image without clipping and to maintain RGB differentiation. The first applies to both luminance and color processing work-streams and the second solely to RGB processes. If one considers a deconvoluted linear luminance image, the next step after a little selective noise reduction is to stretch the image. By its very nature, everything becomes brighter. A couple of medium stretches using the traditional HistogramTransformation HT) tool soon boosts the brighter star cores into the danger zone and at the same time, extends their diffuse boundary.

The idea of a variable strength stretch, based on image intensity comes to mind; a simple image mask that protects the brightest areas may be a partial solution. The MaskedStretch tool does precisely this but in a more sophisticated progressive way. This tool stretches stars to form a small but pronounced central peak with an extended faint periphery. Used on its own it can cause stars to take on a surreal appearance. If you apply it to an image that has already received a modest non-linear stretch, the effect is more acceptable. First apply a medium stretch to the image, using typical settings as the ones in fig.27, followed by the MaskedStretch tool, set to 1,000 iterations and a clipping point set as a compromise between background noise and feature brightness. To avoid either stretching operation proliferating clipped highlights, the highlight slider on the HT tool is increased to 1.2–1.3, which provides some headroom for the stretching outcome (fig.27).

Another technique for retrospectively reducing star intensity during processing is to use the star shrinking properties of the MorphologicalTransformation tool. As seen before, the act of shrinking stars also dims them as it replaces each pixel with the median of its neighbors. In this case, create a star mask solely for the bright stars and apply an erosion or morphological selection to the bloaters. The same logic applies to the various blurring techniques, that blends a sharp centrally and clipped peak with its immediate surroundings. To blend the clipped star core apply a tight star mask and apply the convolution tool, or one of the multi-scale tools, with its bias setting reduced for the first two scales. A good non-linear luminance channel may have peak intensities below 0.95 and most star peak intensities below 0.8. If the image looks dull, one visual trick is to increase the apparent contrast by lowering the background level from the nominal 0.125 and mounting the image with a dark surrounding and with no nearby white reference. It is amazing how much you can fool the brain with simple visual tricks such as these.

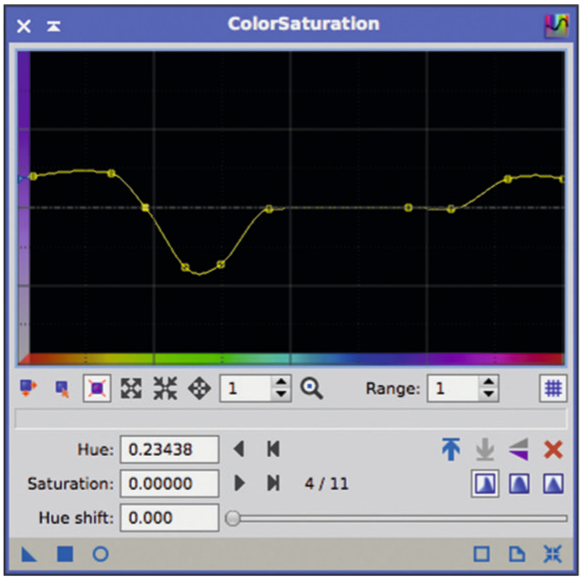


fig.28 The ColorSaturation tool applies selective boosts and reductions to various colors. Here, reducing green saturation and boosting yellow–magenta.

Having processed the luminance channel, it is now the turn of the color channels. It is important to remember that this only concerns the color information. Stretching a color image has two effects on saturation. It accentuates the differences (increases saturation) between color channels in the area of maximum local contrast increase (typically shadow areas) and conversely decreases the differences in the highlight regions (reducing saturation). At the extreme, if a color image is stretched too far, the individual RGB levels clip and once more bright stars become white blobs. (Once this occurs, there is no method to recover the original color information and an inverse intensity transform simply creates grey blobs.)

There are a few more tools at our disposal to improve color differentiation: The ColorSaturation tool selectively increases color saturation. I often use this to balance red and blue star saturation in an image and at the same time, suppress anything green (fig.28). Overall color saturation appears as one of the settings in the CurvesTransformation tool. Select the "S" for saturation and drag the curve to boost color saturation. This changes the color saturation as a function of its present saturation (not color or brightness). A typical curve predominantly boosts areas of low saturation (and for that reason may require a mask to protect featureless sky, to avoid increasing chroma noise).

Lastly, the color balance tools or individual RGB channels in the HistogramTransformation tool manipulate individual