

DSOMask

By Bradley Craig

Facilitates target mask creation for isolating image target for further processing. [\[more\]](#)

Keywords: mask star protection nebula mask

Contents

[\[hide\]](#)

[1 Description](#)

[1.1 Mask Generator](#)

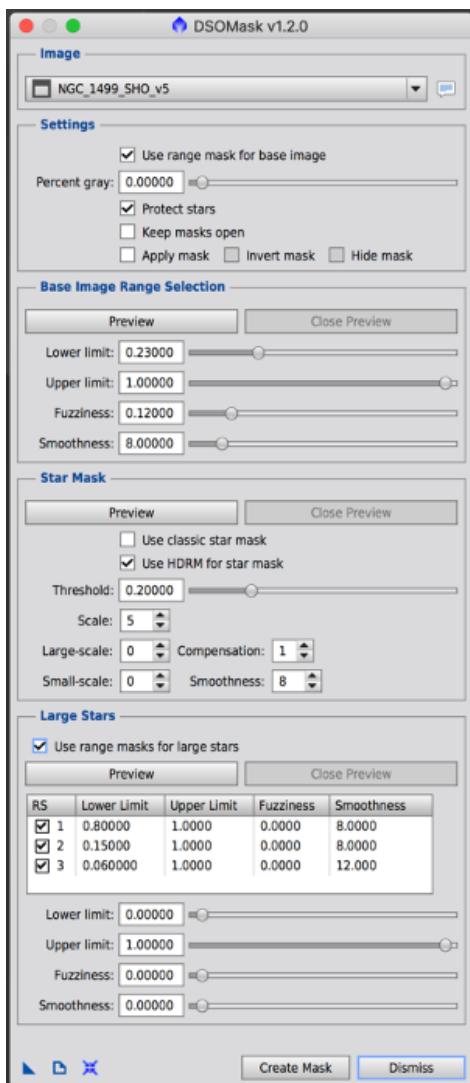
[2 Usage](#)

[2.1 Source Image](#)
[2.2 Settings](#)
[2.3 Base Mask](#)
[2.4 Star Mask](#)
[2.5 Large Star Mask](#)
[2.6 Examples](#)

[Related Tools](#)

[1 Description](#)

[\[hide\]](#)



The DSOMask script brings together core PixInsight processes to create a mask for protecting the target signal of an image for further processing.

DSOMask provides the following functions:

- [Settings](#) to control aspects of mask creation and final combination.
- Ability to define the [base mask](#) of the source image to protect the target.
- Ability to define a [star mask](#) to protect stars.
- Ability to define a [large star mask](#) to protect larger stars.
- Ability to [combine](#) them in multiple ways to [create](#) the desired mask.
- Ability to preview any of the supporting masks that will be used in creating the final mask.
- Ability to apply the final mask or a previewed mask to the source image and then invert and/or hide the mask on the source image.
- Ability to preserve script settings between sessions of the script using the PixInsight Core JavaScript Settings object.
- Ability to preserve script settings for a particular execution of the script using the PixInsight Core JavaScript Parameters object.

1.1 Mask Generator

Base Image

The base mask serves as the definition of the protection for the main structure of the image. There are two methods of base mask creation. One is using a standard [RangeSelection](#) process which is controlled by [settings](#) on the script dialog.

The other method, if not using a [range mask](#) for the base mask is a grayscale starless version of the source image. This image is created using e ATrousWaveletTransform with the first layer removed on a duplicate of the source image. The duplicate is then subtracted from the source image to create the base mask.

Call this image **BASE**.

Star Mask

Optionally, a [star mask](#) can be created and added to the final mask to allow for protection of the stars in the image. Two methods are available for star mask creation. A standard star mask can be a *StarMask* process controlled by [settings](#) on the script dialog.

The other method is to use the [scripts->ClassicStarMask](#) script.

Call this image **STARS**.

Larger Star Mask

Optionally, a [star mask](#) to protect larger stars not covered by the star mask can be created. This is done by using 1 to 3 *RangeSelection* processes controlled by [settings](#) on the script dialog.

Call this image **LSTARS**.

Gray Mask

Optionally, a gray mask can be used to add some level of protection for the signal of the main structure. This can be controlled by [settings](#) on the script dialog.

Call this image **GRAY**.

Mask Construction

If the **GRAY** mask is utilized then the final mask is created by

$\min(\text{GRAY} | - (\text{STARS} | \text{LSTARS} | \text{STARS+LSTARS}) |), \text{BASE}$

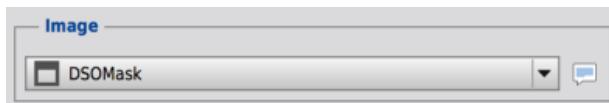
Otherwise the final mask is created by

$\text{BASE} | - (\text{STARS} | \text{LSTARS} | \text{STARS + LSTARS}) |$

2 Usage

[hide]

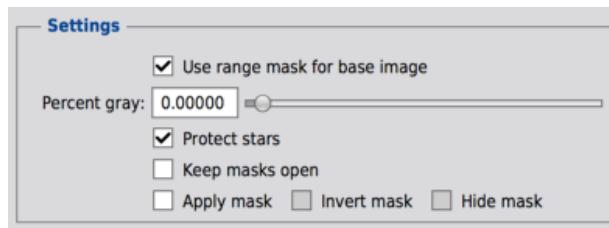
2.1 Source Image



Image

Select the source image from the drop down box. The image can be color or mono.

2.2 Settings



Use range mask for base image

If selected, the PixInsight *RangeSelection* process will be executed on the source image using the settings defined in the [base mask](#) section to create the base image for the mask.

Otherwise, create a starless version of the source image using the PixInsight *ATrousWaveletTransform* and *PixelMath* processes.

Percent gray

Controls the amount of grayness in the final mask. If this is set to 0 (zero) then the final mask will be that of the base mask.

If this is set to a non-zero amount, then that will be used to build a gray mask first and then use the *minimum* function on the gray mask and the base mask.

Protect stars

If selected then a [star mask](#) will be generated to apply star protection in the final mask.

Optionally, a [large star mask](#) for use in the final mask to protect larger stars not covered in the star mask is enabled.

Keep masks open

If selected, then in addition to the final mask, any supporting masks used will remain open.

If not selected, then only the final mask will remain open.

Apply mask

If selected, then the final mask will be applied to the source image.

If previewing any of the supporting masks, then they will be applied to the source image.

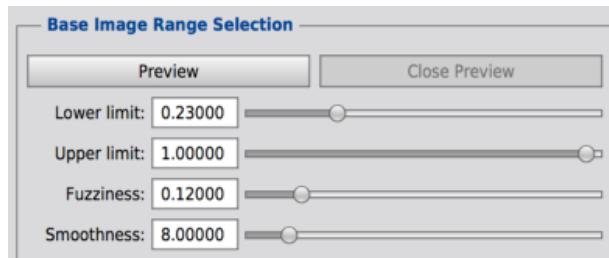
Invert mask

If selected, the currently applied mask on the source image will be inverted.

Hide mask

If selected, the currently applied mask on the source image will be hidden.

2.3 Base Mask



Preview

Execute the *RangeSelection* process with the current settings and open a window with the results. The results can be applied to the source image using [apply mask](#) setting.

Close Preview

Close the open base mask preview.

Lower limit

The *RangeSelection* lower limit setting.

Upper limit

The *RangeSelection* upper limit setting.

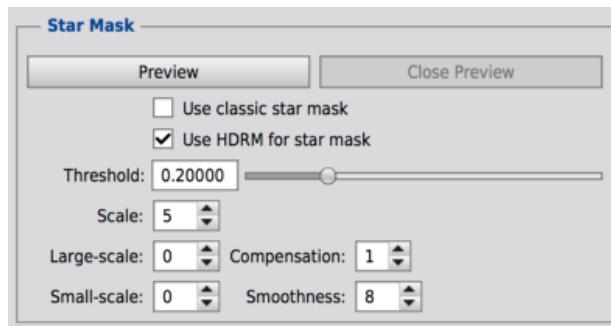
Fuzziness

The *RangeSelection* fuzziness limit setting.

Smoothness

The *RangeSelection* smoothness limit setting.

2.4 Star Mask



Preview

Execute the *StarMask* process with the current settings and open a window with the results. The results can be applied to the source image using [apply mask](#) setting.

Close Preview

Close the open star mask preview.

Use classic star mask

Create the star mask using the script *ClassicStarMask* approach.

Use HDRM for star mask

Applies *HDRMultiscaleTransform* to source image before star mask creation using *StarMask*.

Threshold

The *StarMask* threshold setting.

Scale

The *StarMask* scale setting.

Large Scale

The *StarMask* large scale setting.

Compensation

The *StarMask* compensation setting.

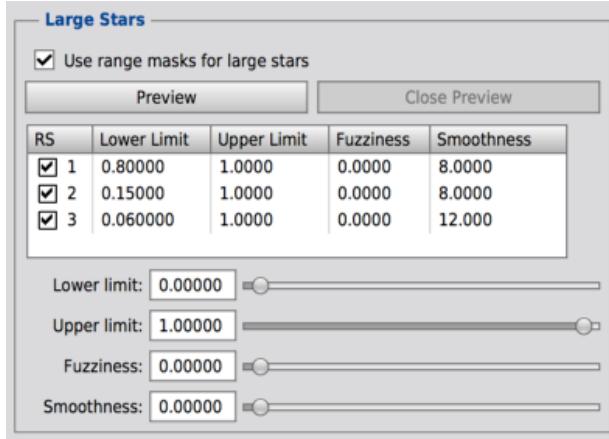
Small scale

The *StarMask* small scale setting.

Smoothness

The *StarMask* smoothness setting.

2.5 Large Star Mask



Use range mask for large stars

Enable up to 3 *RangeSelection* processes to isolate large stars for protection in the final mask.

Preview

Execute the *RangeSelection* process with the current settings and open a window with the results. The results can be applied to the source image using [apply mask](#) setting.

Close Preview

Close the open star mask preview.

Lower limit

The *RangeSelection* lower limit setting.

Upper limit

The *RangeSelection* upper limit setting.

Fuzziness

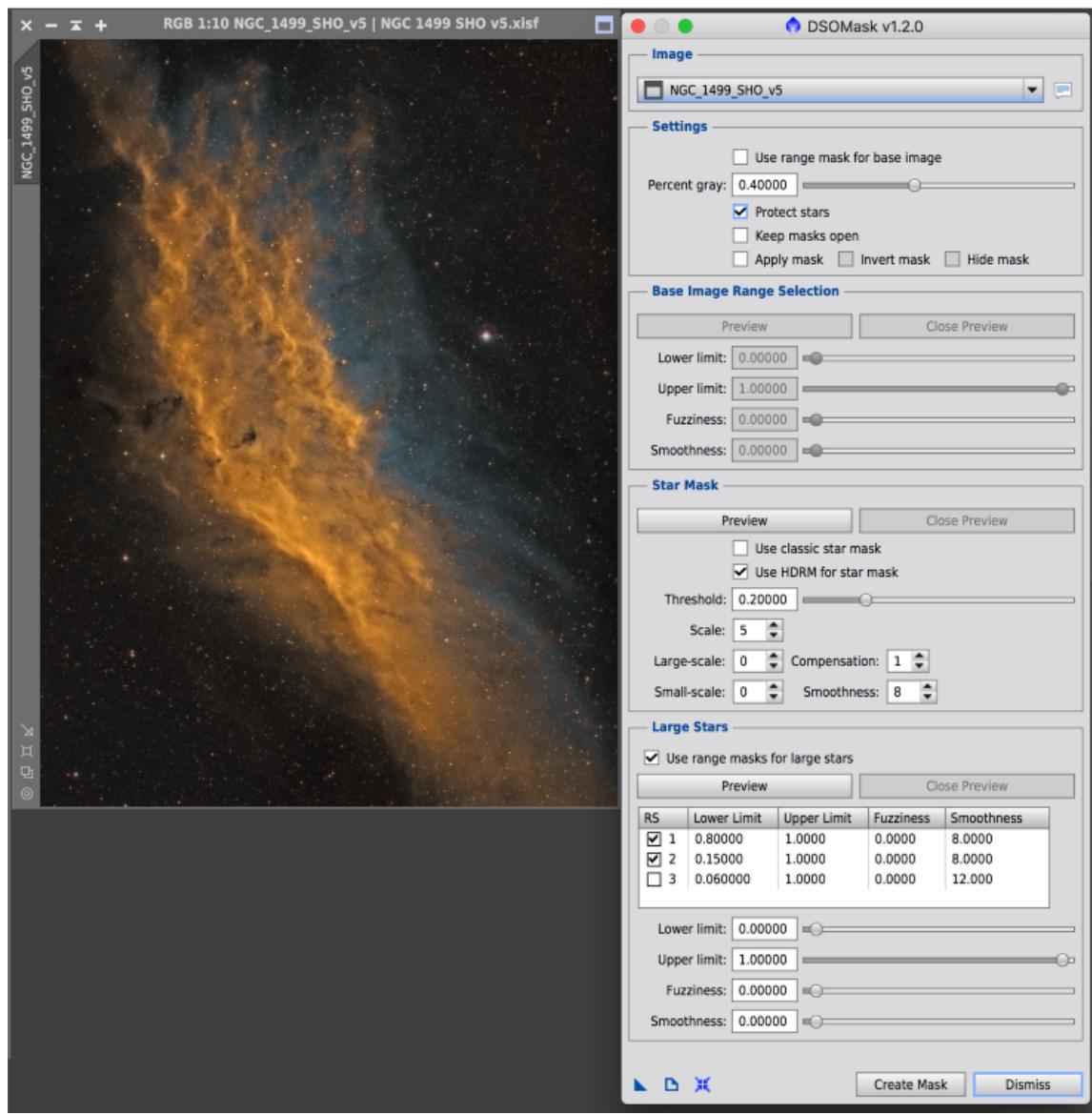
The *RangeSelection* fuzziness limit setting.

Smoothness

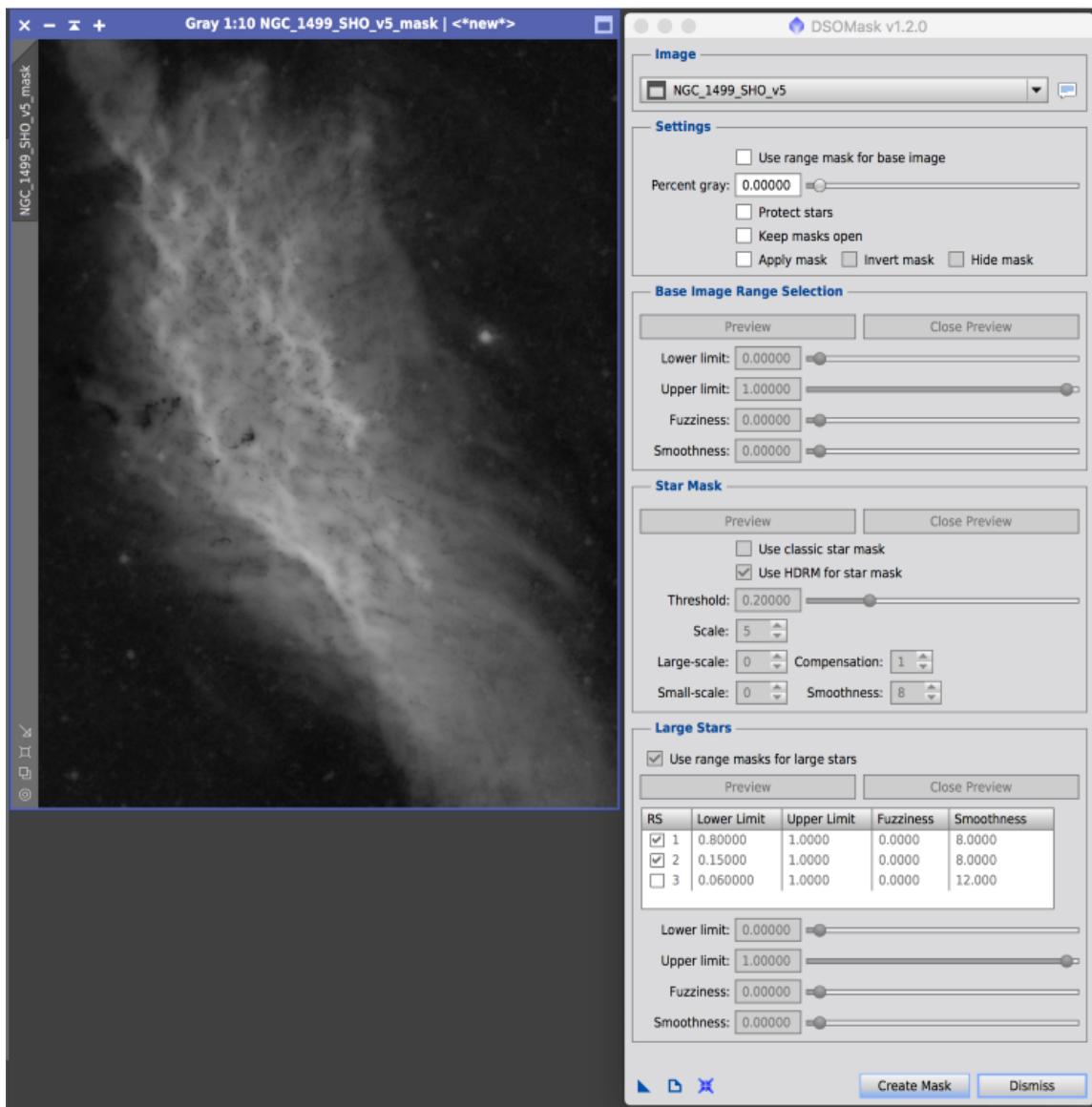
The *RangeSelection* smoothness limit setting.

2.6 Examples

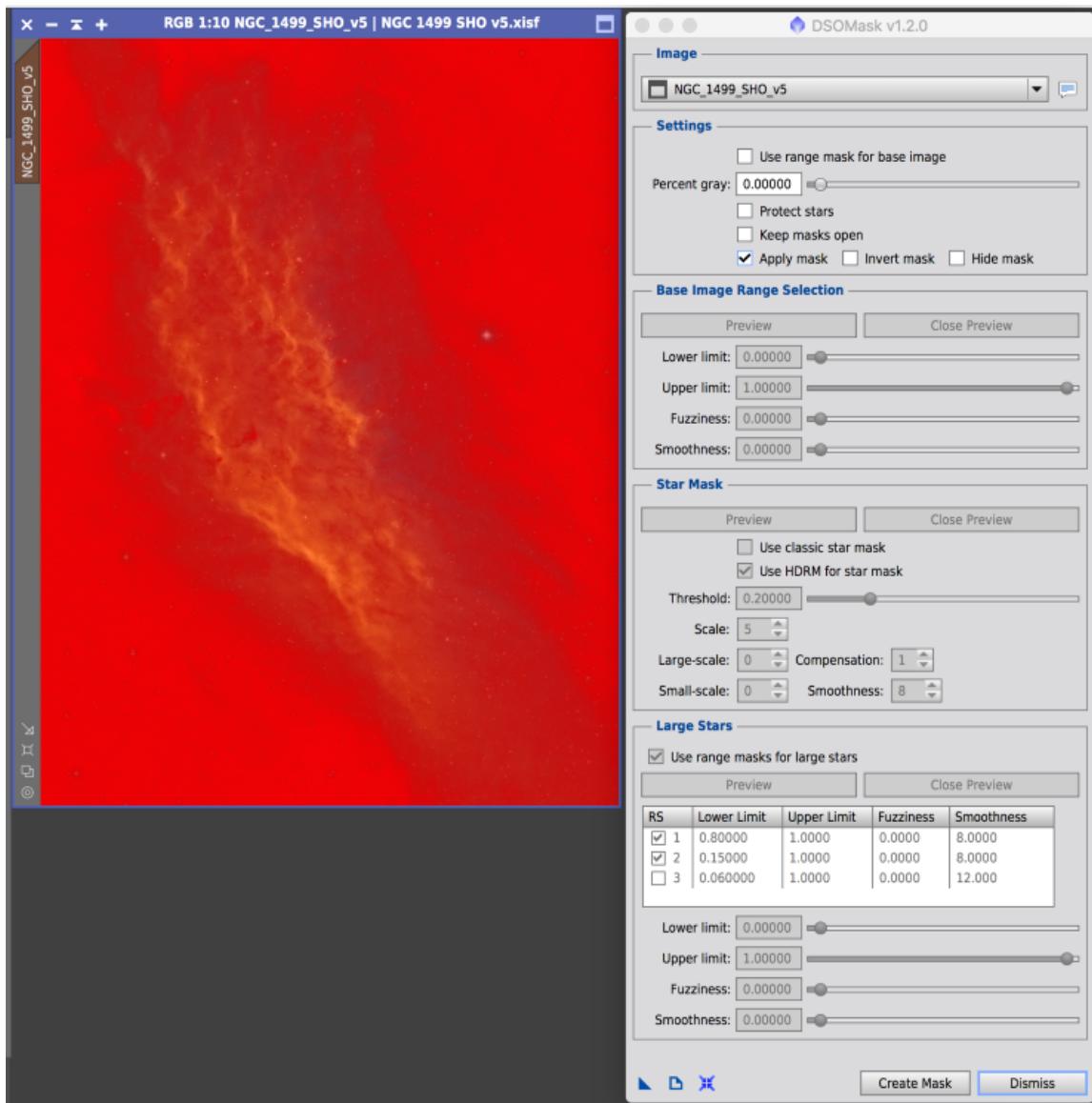
Let's look at some examples of what DSOMask can do using NGC 1499.



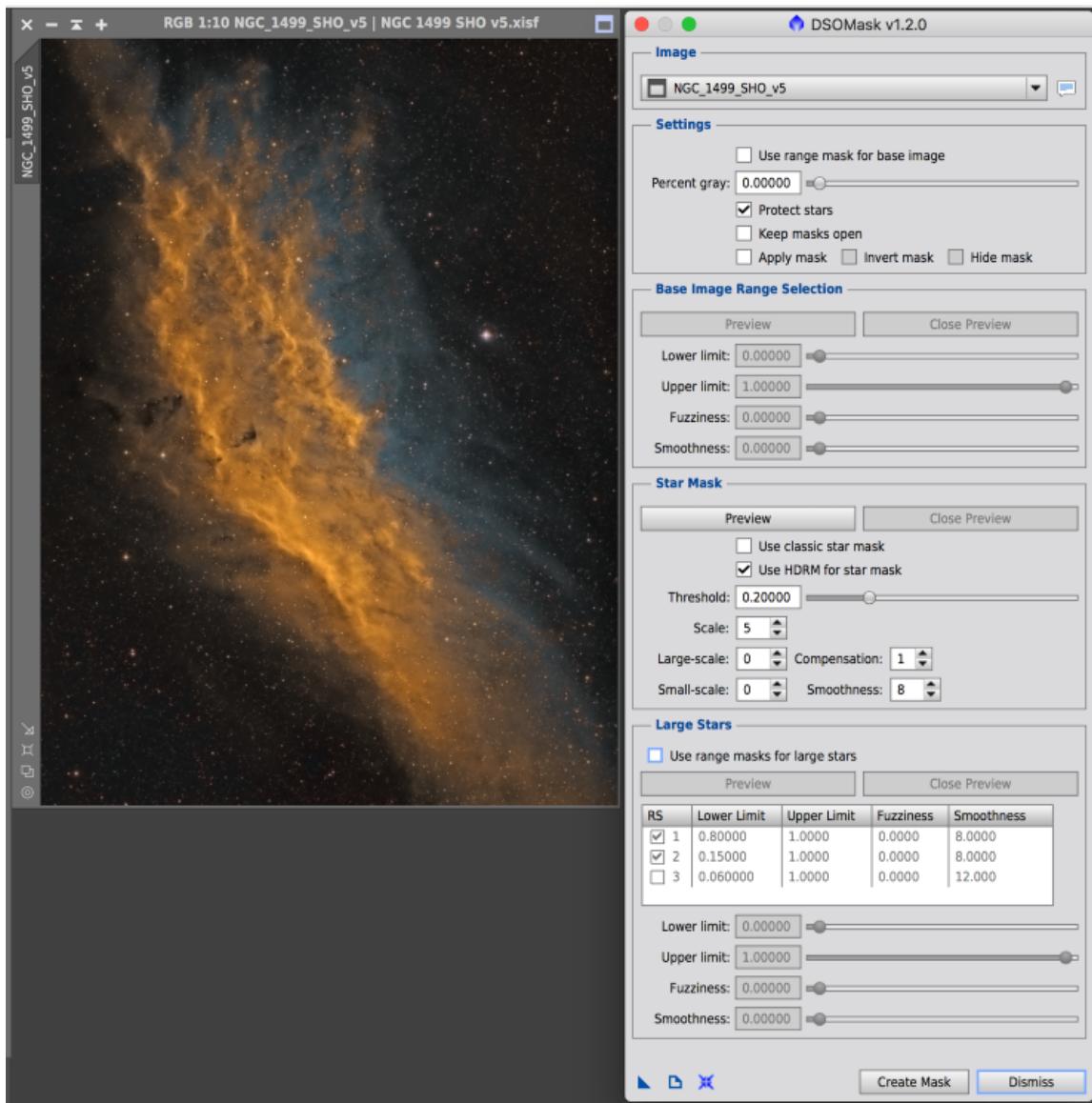
Let's start with the simplest settings. No star protection and using the source image as the mask. The following shows the resulting mask, which is just the gray scale starless version of the image.



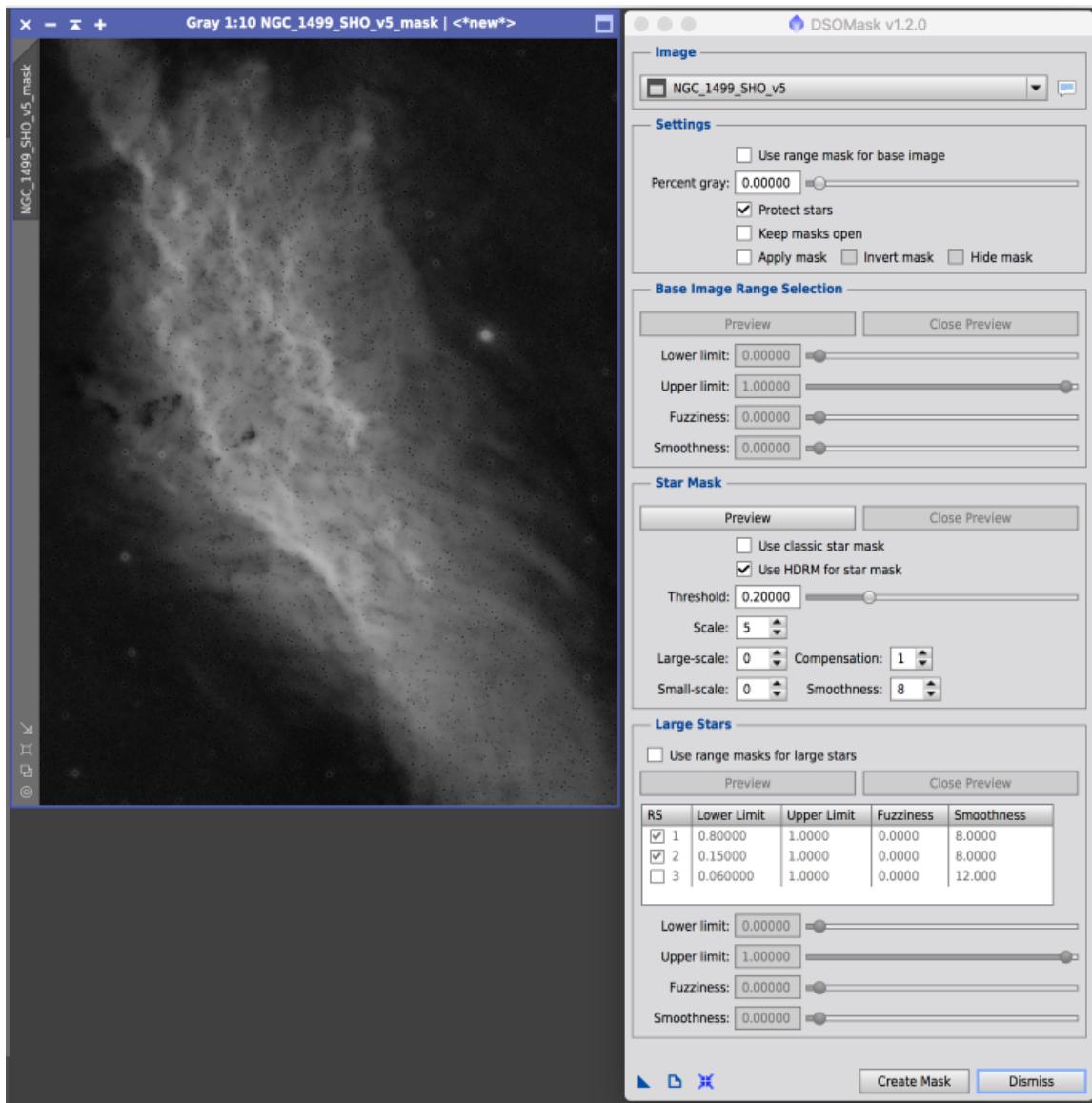
And with the mask applied to the source image.



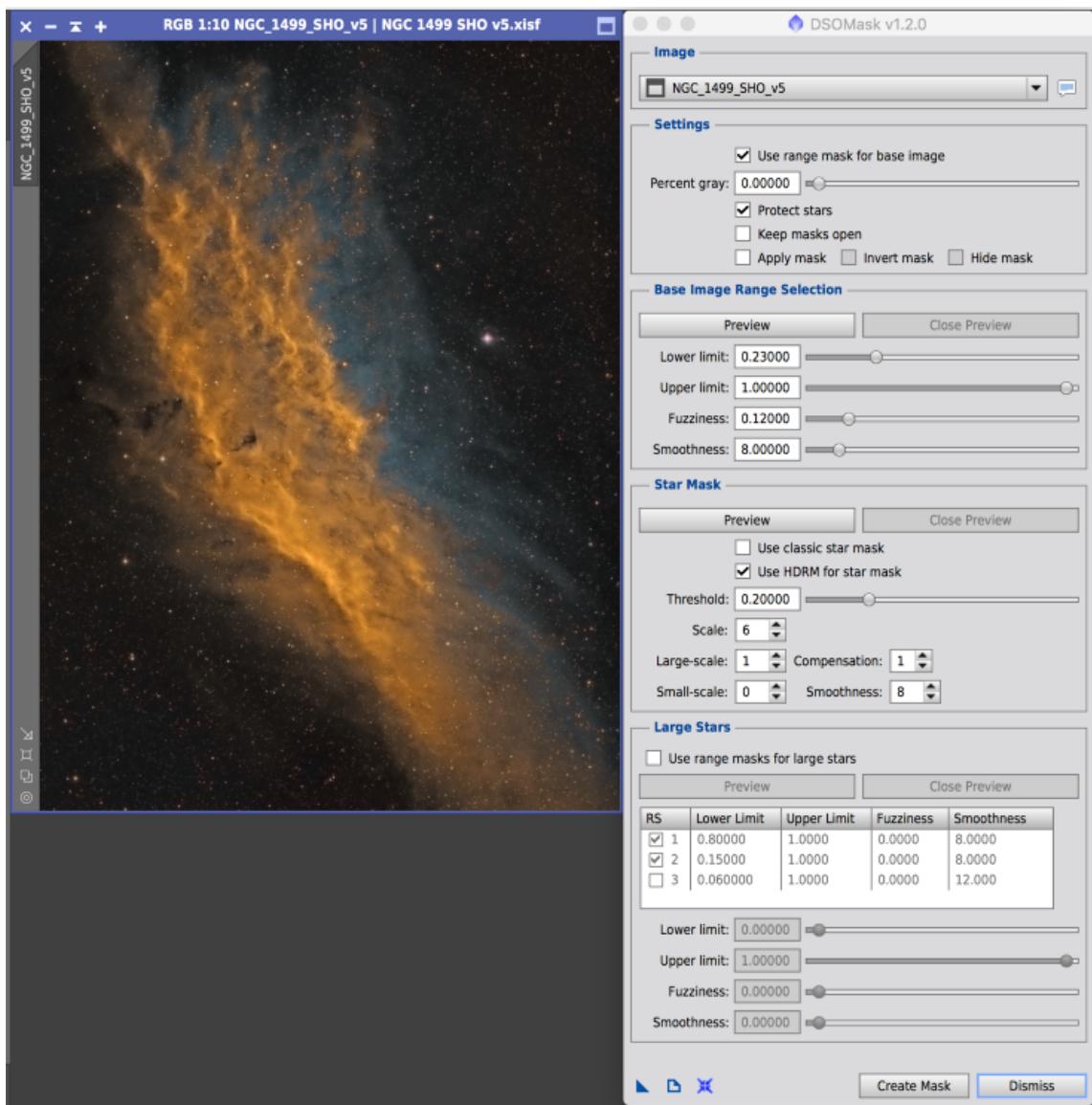
Now let's add in star protection.



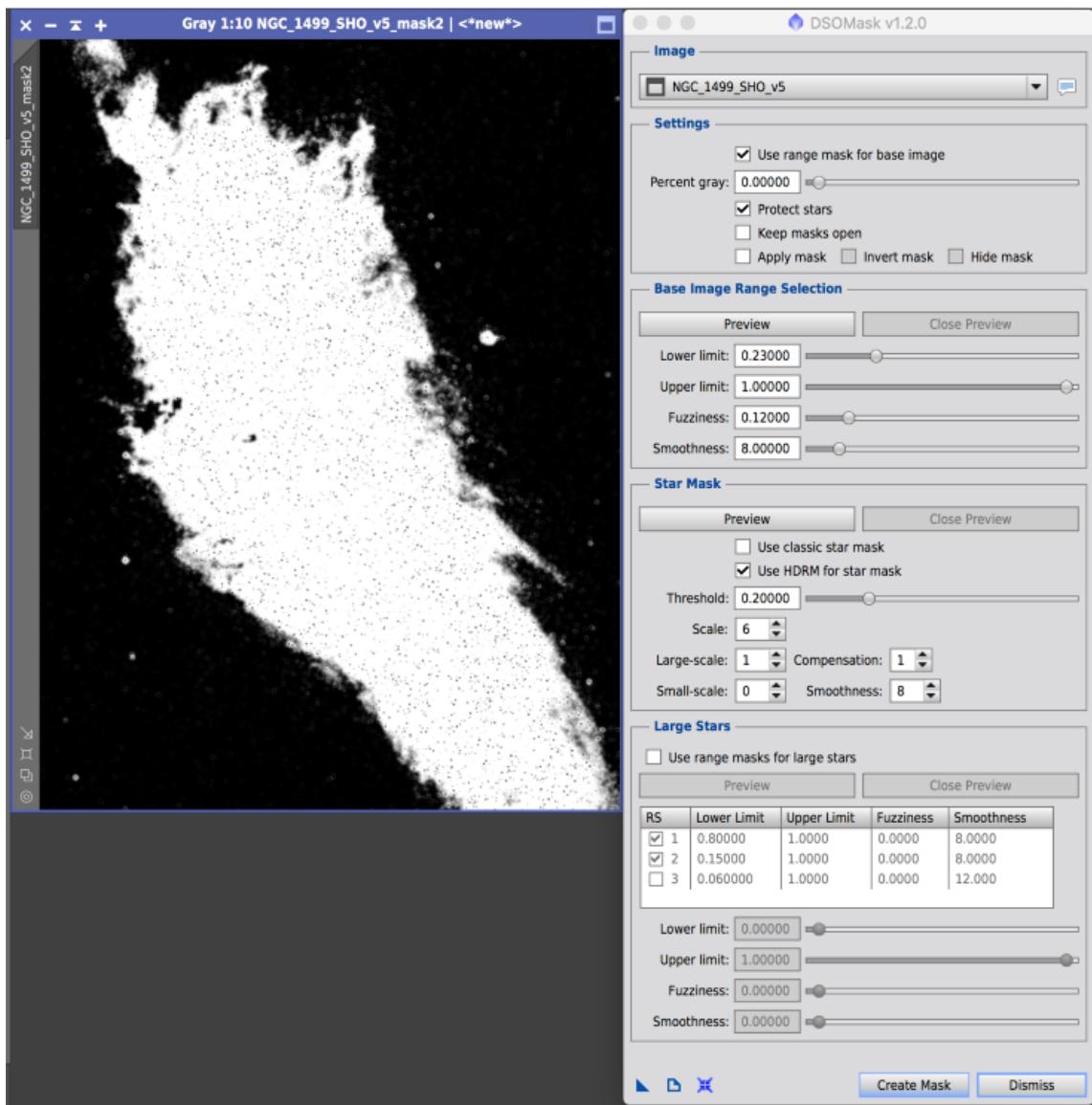
And the resulting mask.



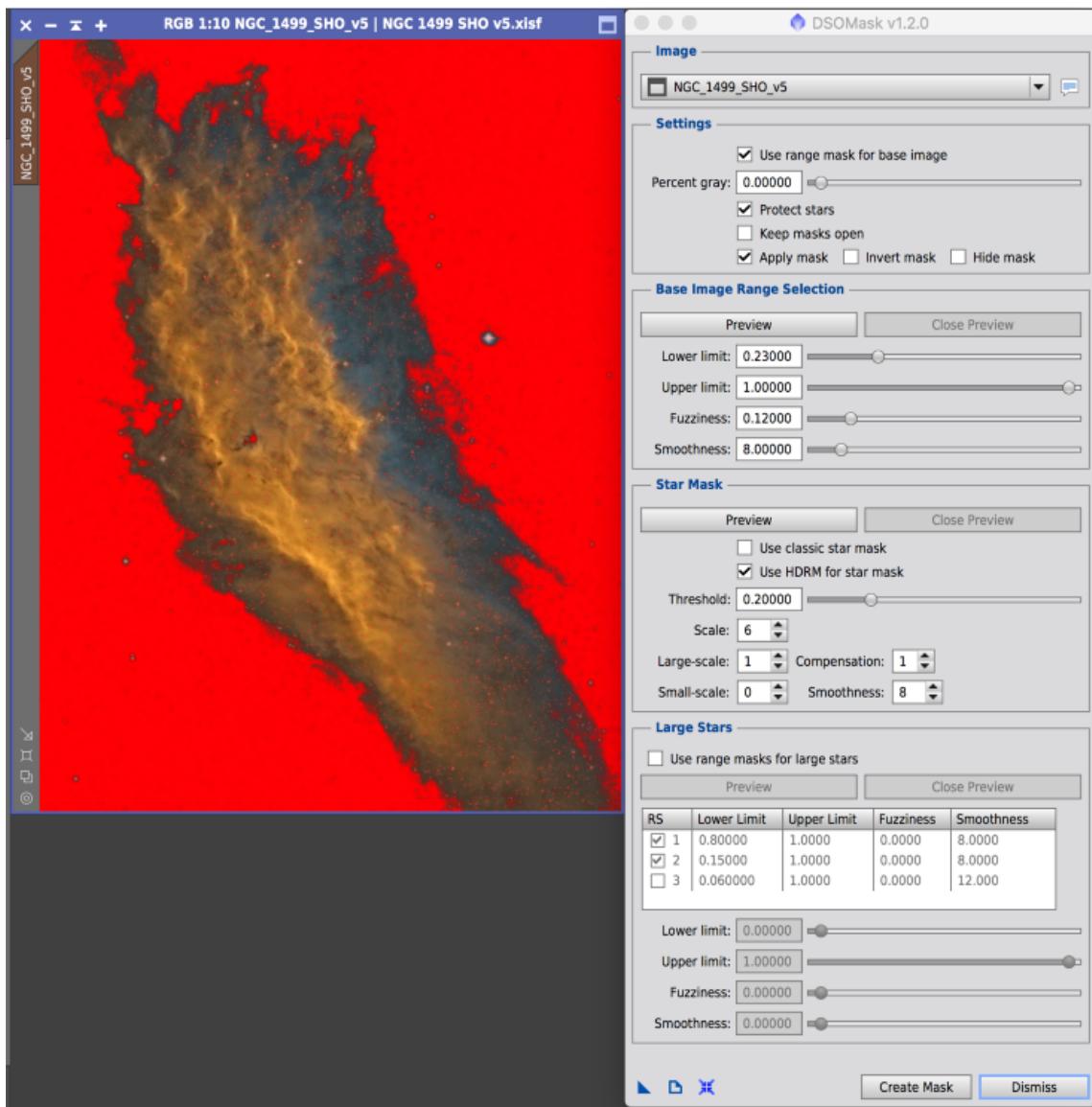
For another example, let's use a range mask as the base and add star protection.



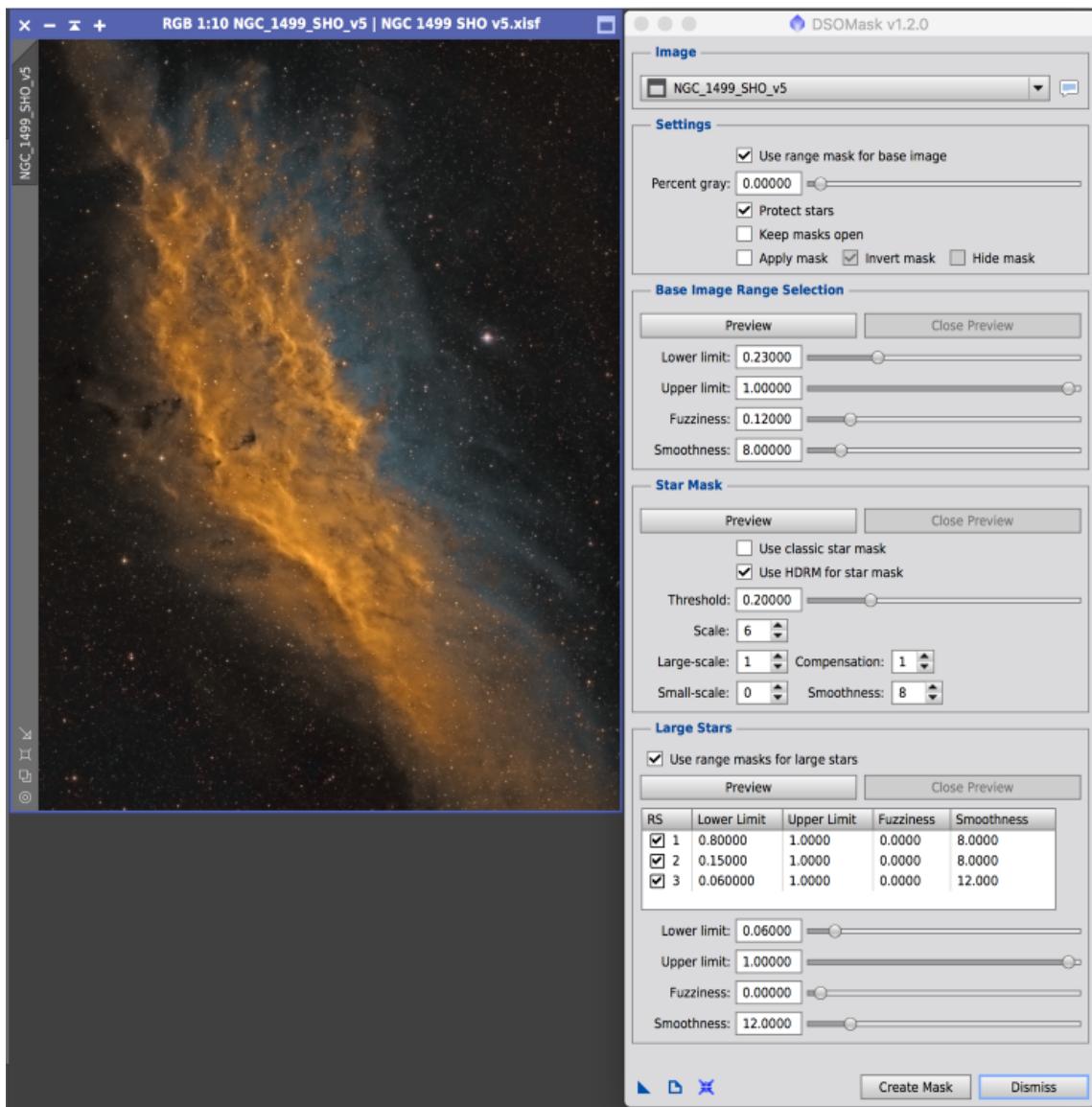
And the resulting mask.



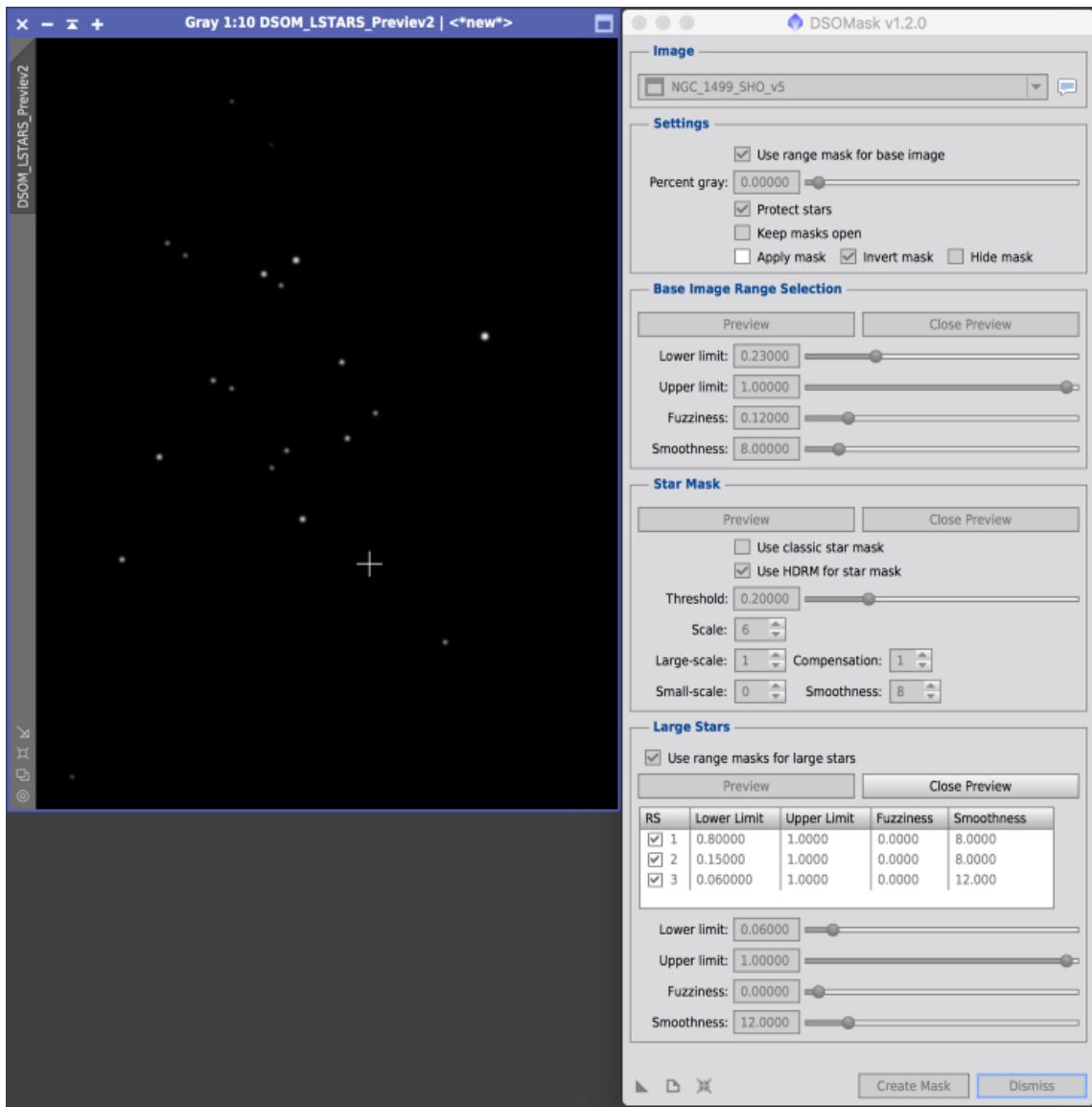
And the mask applied to the source image.



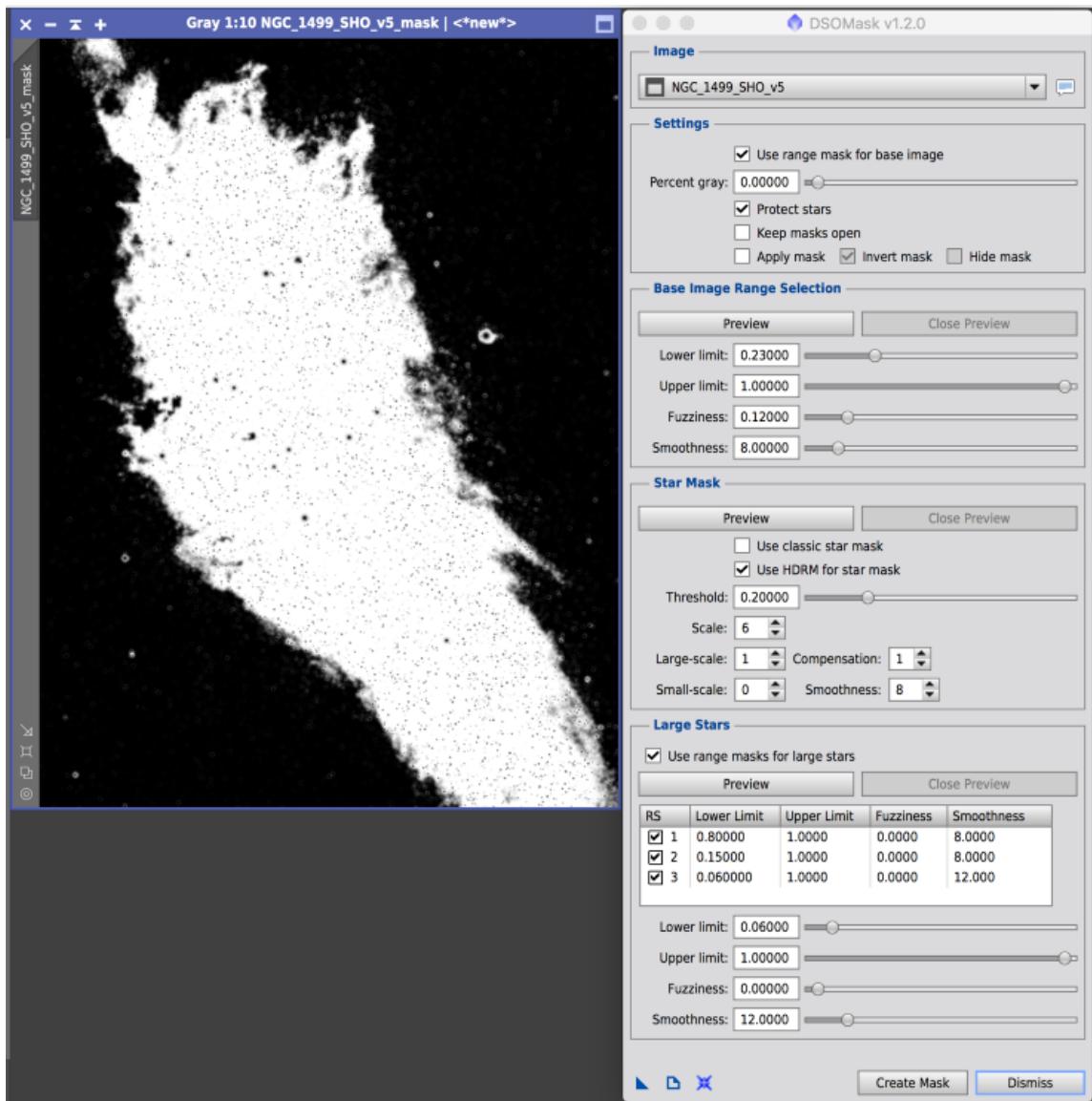
Now let's add some protection for the bigger stars. We'll use 3 successive *RangeSelection* processes. The first will be on the source image, and the next 2 on the previous range mask.



First let's preview what the large star mask looks like with the current settings for the 3 *RangeSelection* processes.



Now let's see the final mask.



Related Tools

[RangeSelection](#), [StarMask](#), [PixelMath](#), [ATrousWaveletTransform](#)

Copyright © 2018 Bradley Craig. All Rights Reserved.

Generated by the PixInsight Documentation Compiler script version 1.6.2 on 2018-02-23 01:11:16 UTC