

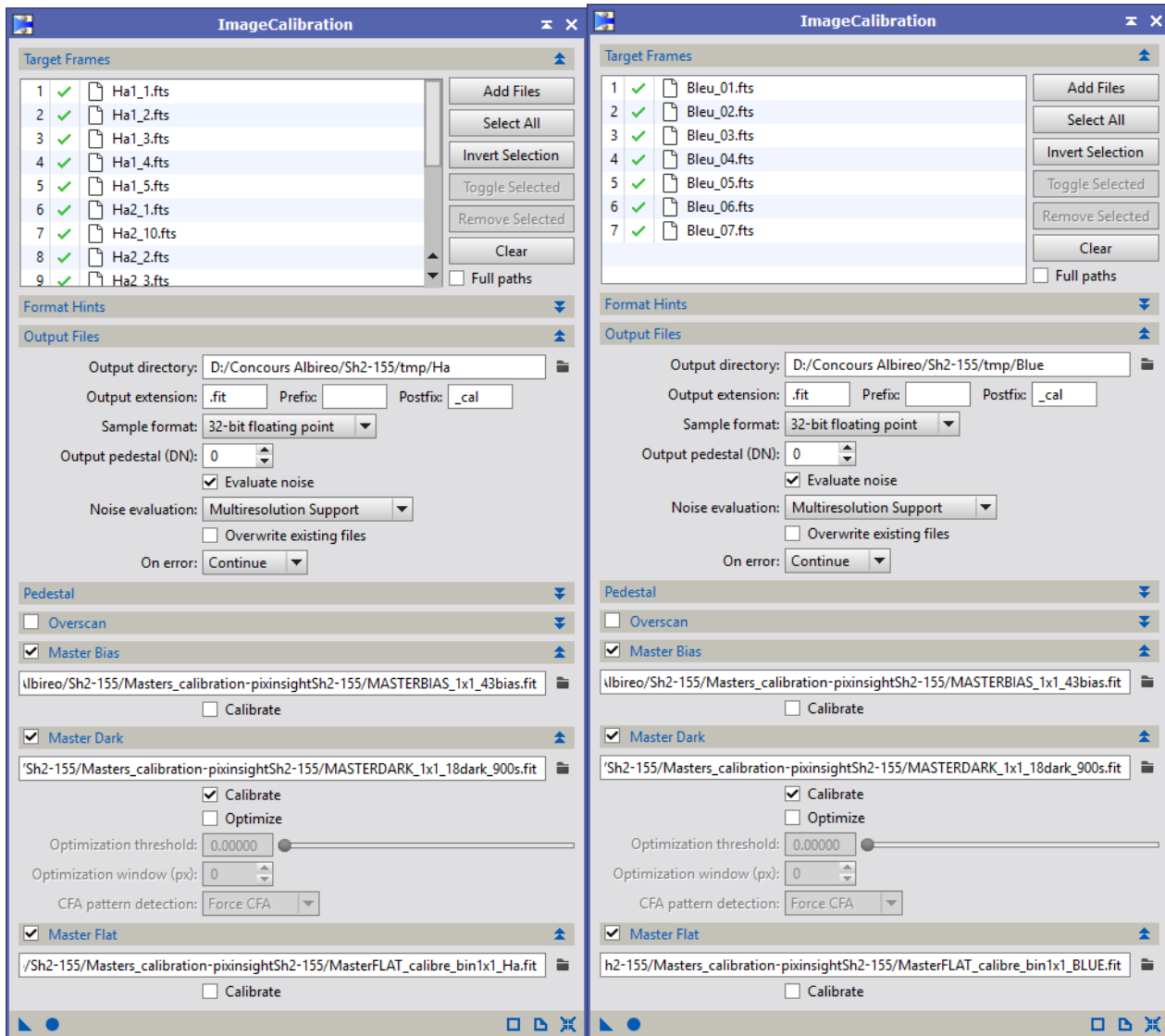
*Albiréo*<sup>78</sup>

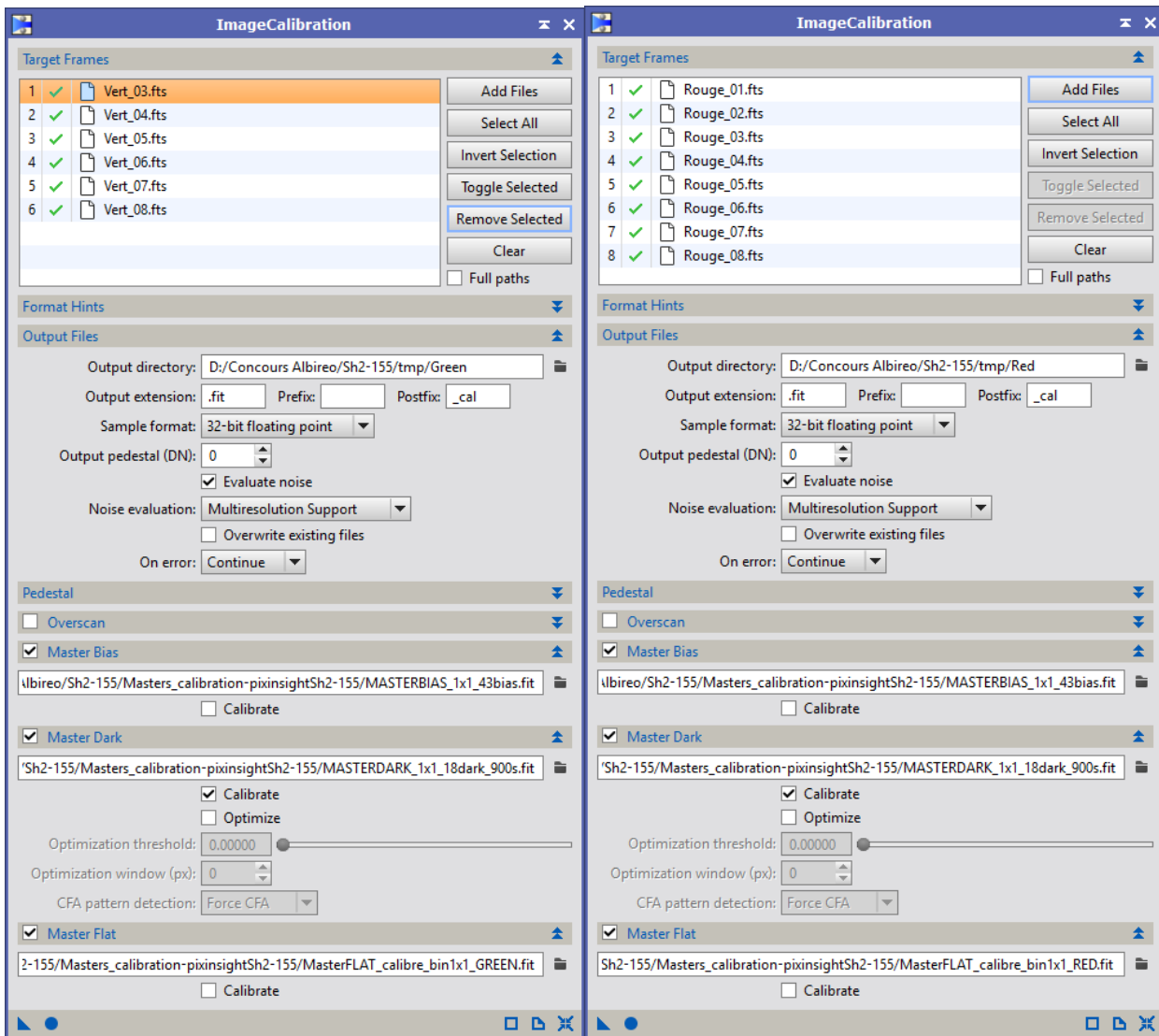
**Sh2-155**  
**Nicolas**



# 1. Prétraitement

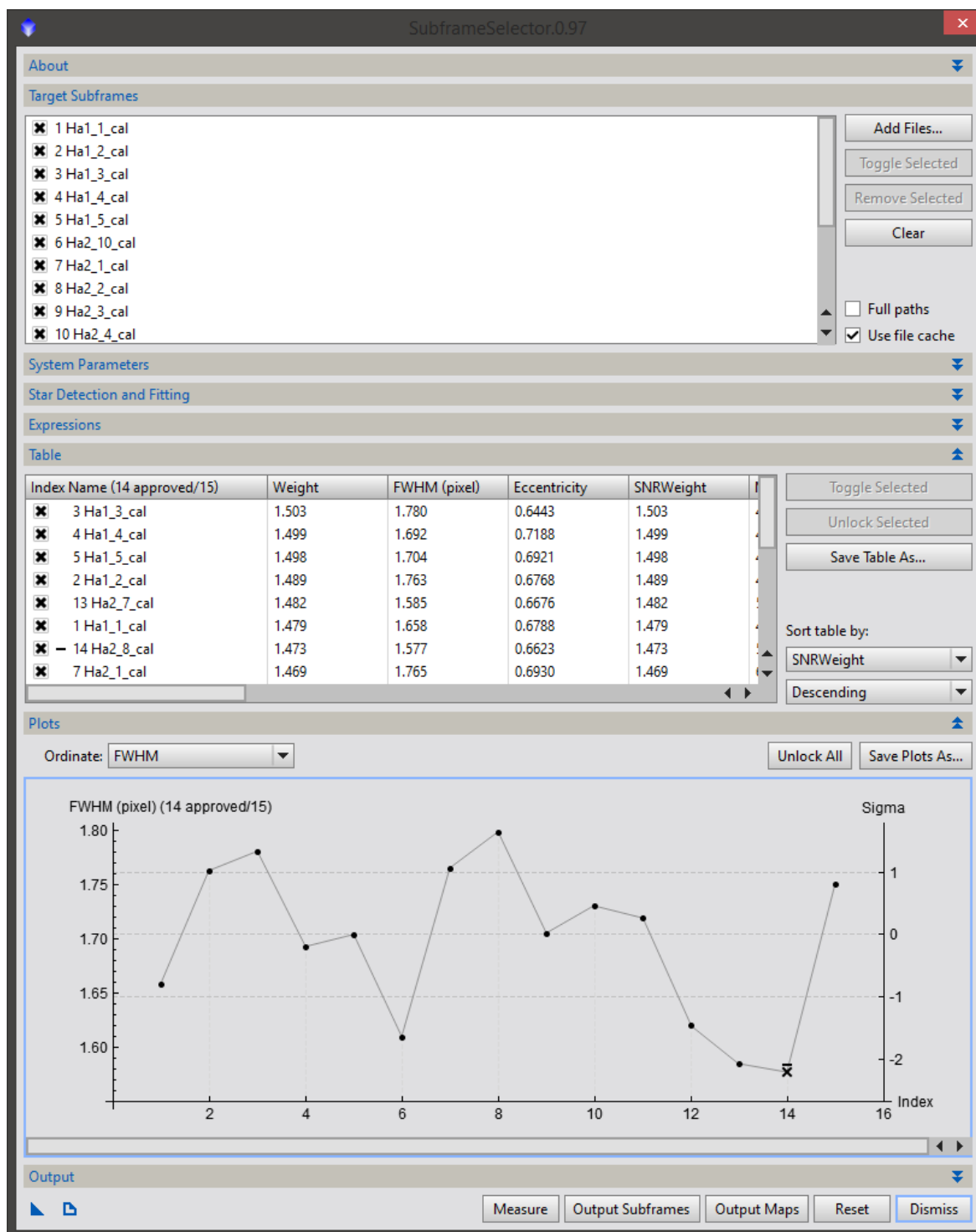
→ J'ai calibré chaque couche en utilisant les masterdark, masterbias et masterflat correspondants





La première image de la couche verte semble corrompue, je fera donc sans...

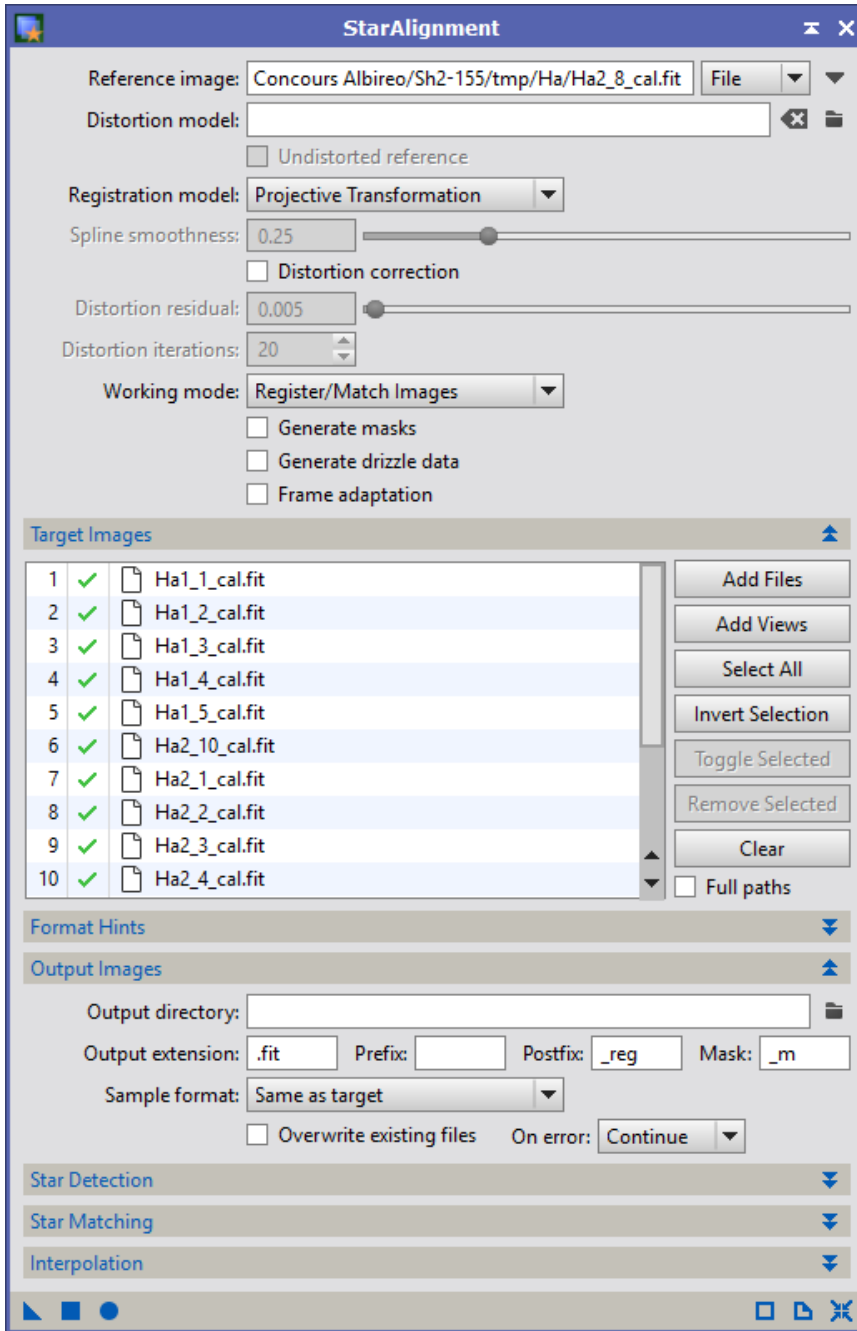
→ Le script *SubFrameSelector* me permet de choisir l'image qui a la meilleure FWHM



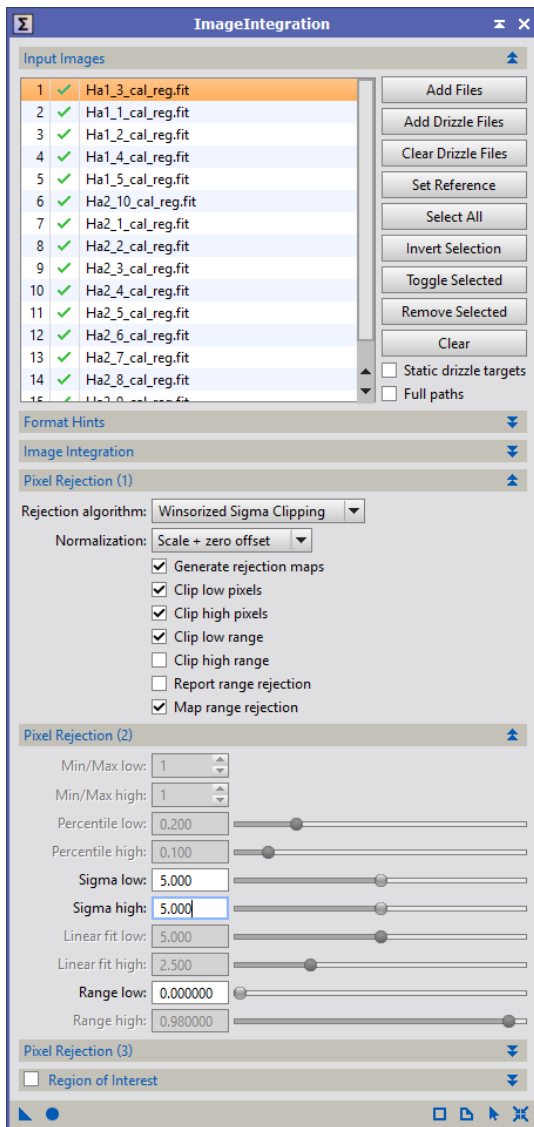
Il s'agit de l'image Ha2\_8

J'en profite pour noter l'image qui a le meilleur SNR (Ha1\_3).

→ J'aligne ensuite la totalité des images à partir de la même référence



→ J'empile les images avec *ImageIntegration*. Je répète cette opération sur chacune des couches.



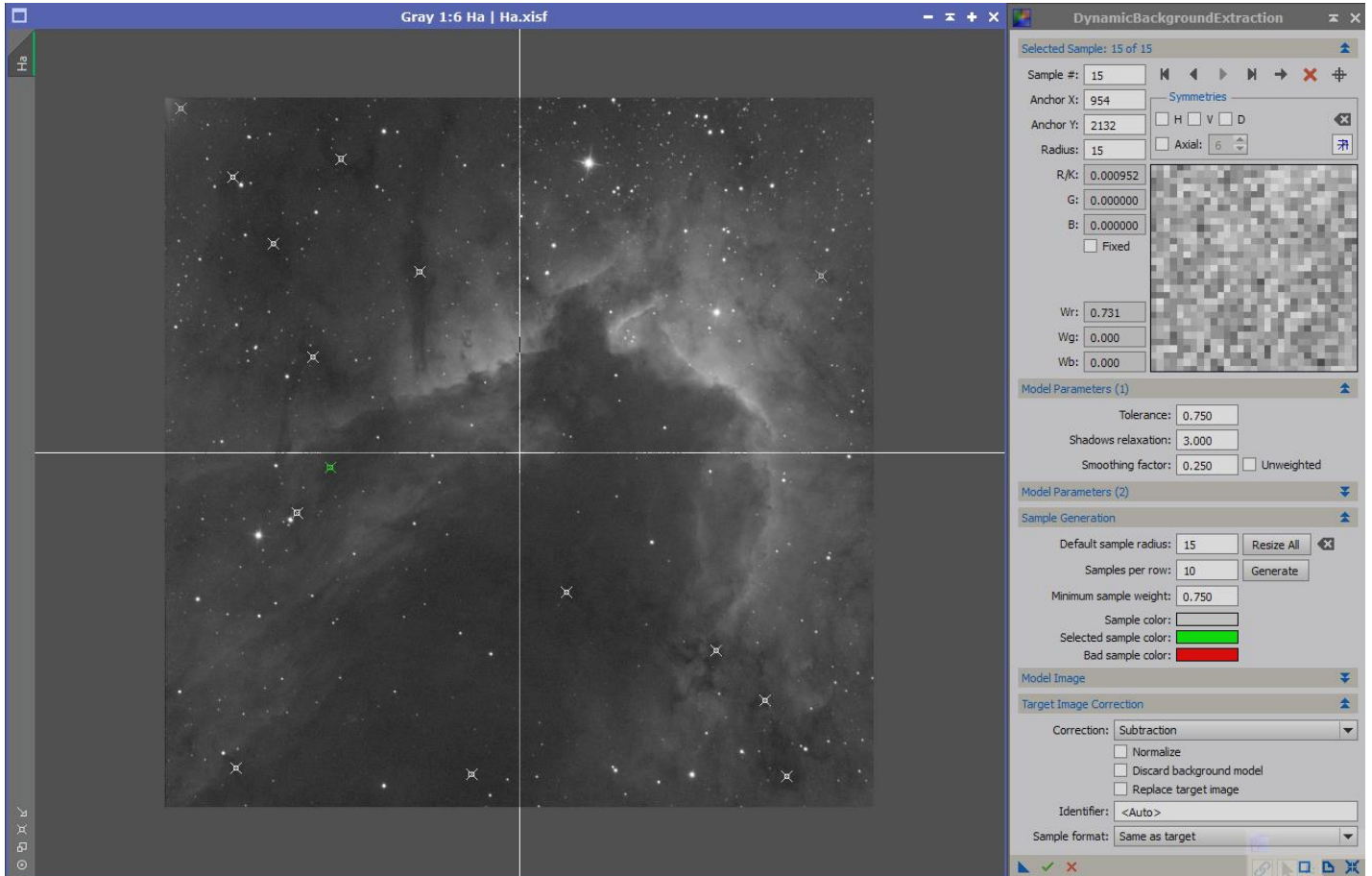
→ Je sauvegarde chaque couche.



## 2. Traitement

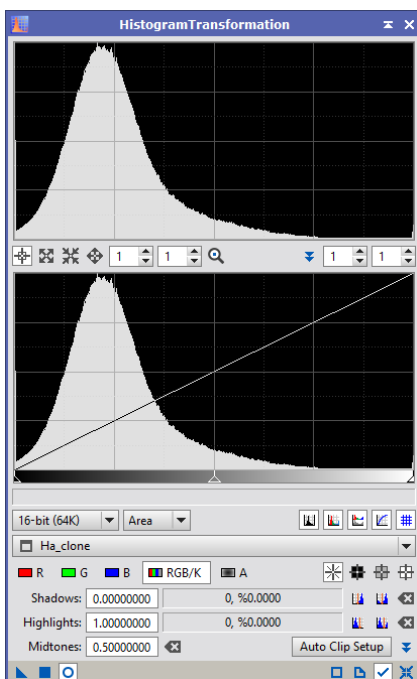
### → Traitement de la couche Ha

Retrait du gradient (pas évident avec autant de nébulosités). Je prends bien soin de conserver mon process icon configuré, qui me servira à appliquer le même retrait de gradient sur chacune des couches.



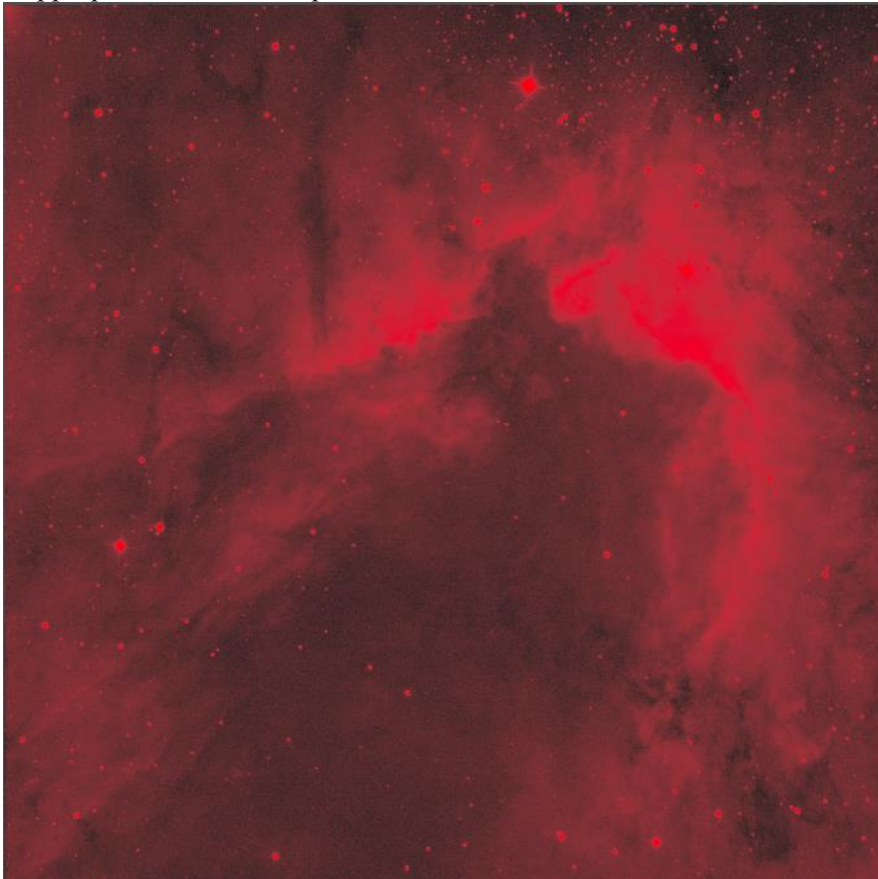
### → Retrait du bruit

Je clone l'image, passe ce clone en mode non-linéaire et resserre l'histogramme pour en faire un masque.

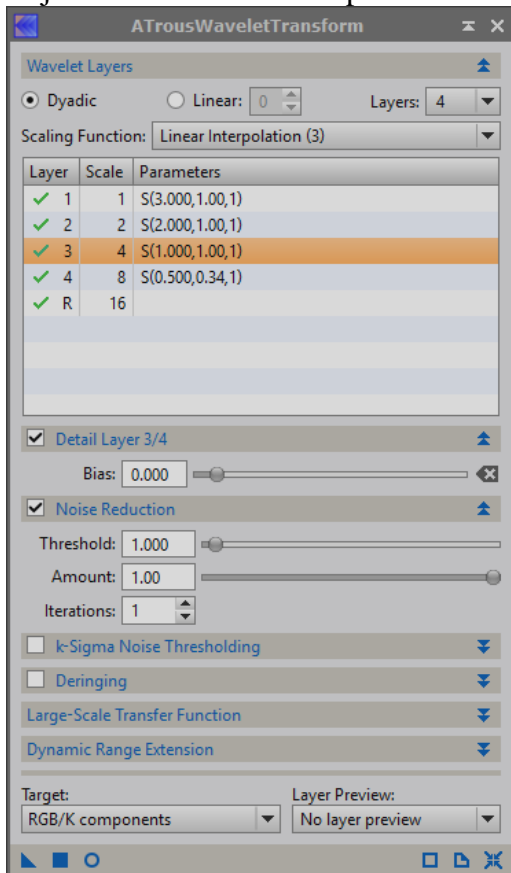




J'applique ensuite ce masque et l'inverse.



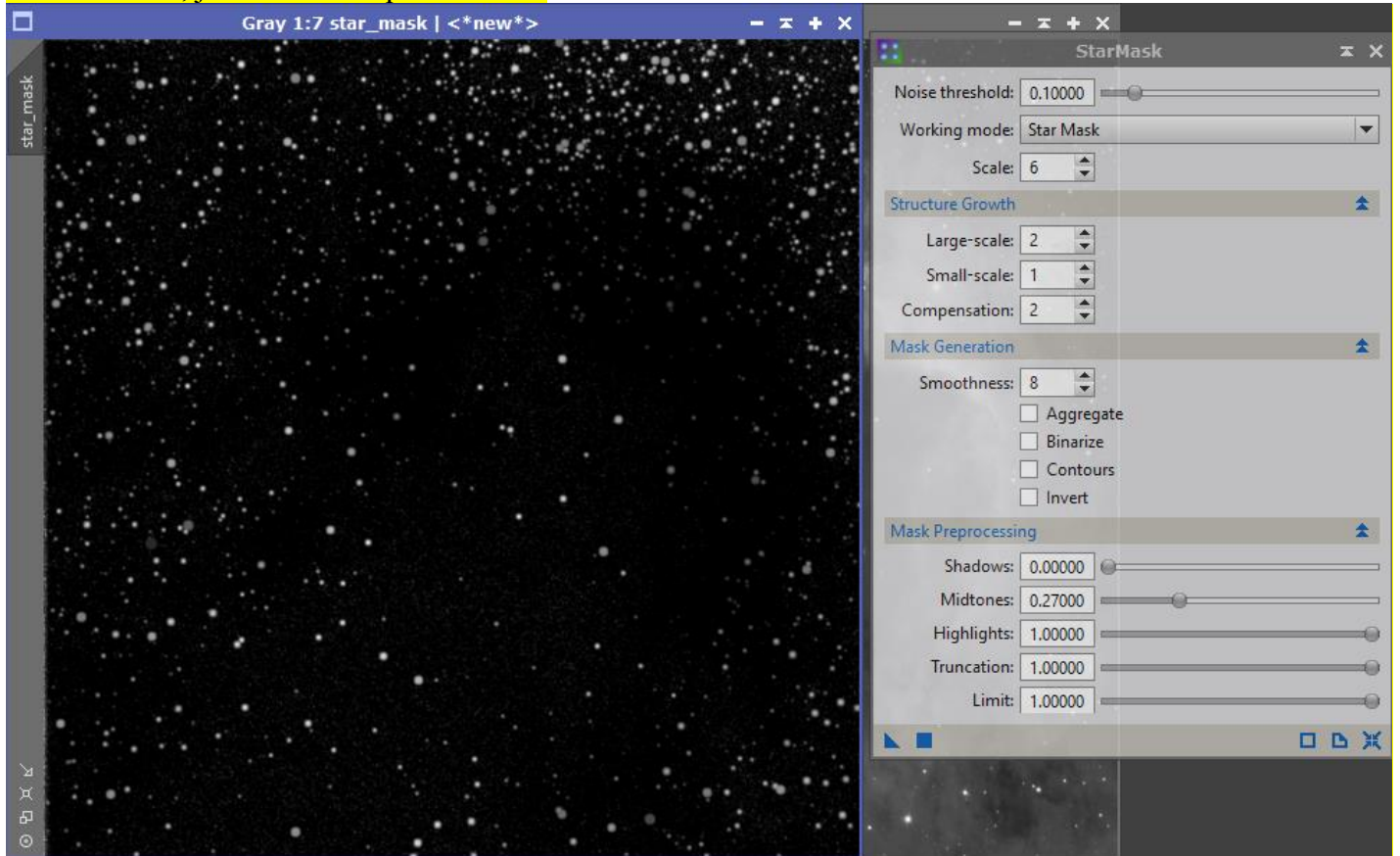
Et je réduis le bruit avec le process *AtrousWaveletTransform*.



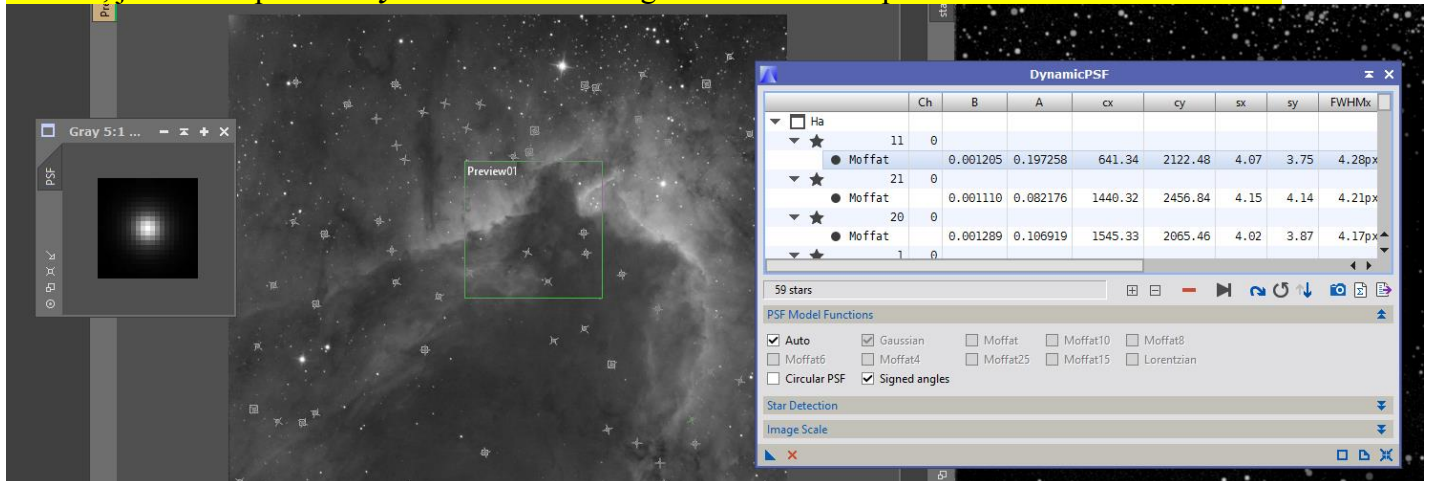


## → Déconvolution

Premièrement, je crée un masque d'étoiles.

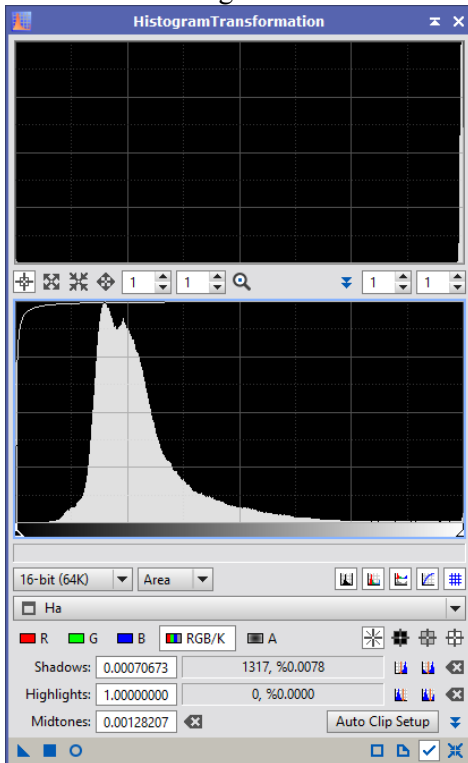


Ensuite, j'utilise le process *DynamicPSF* afin de générer une PSF à partir d'une sélection d'étoiles.



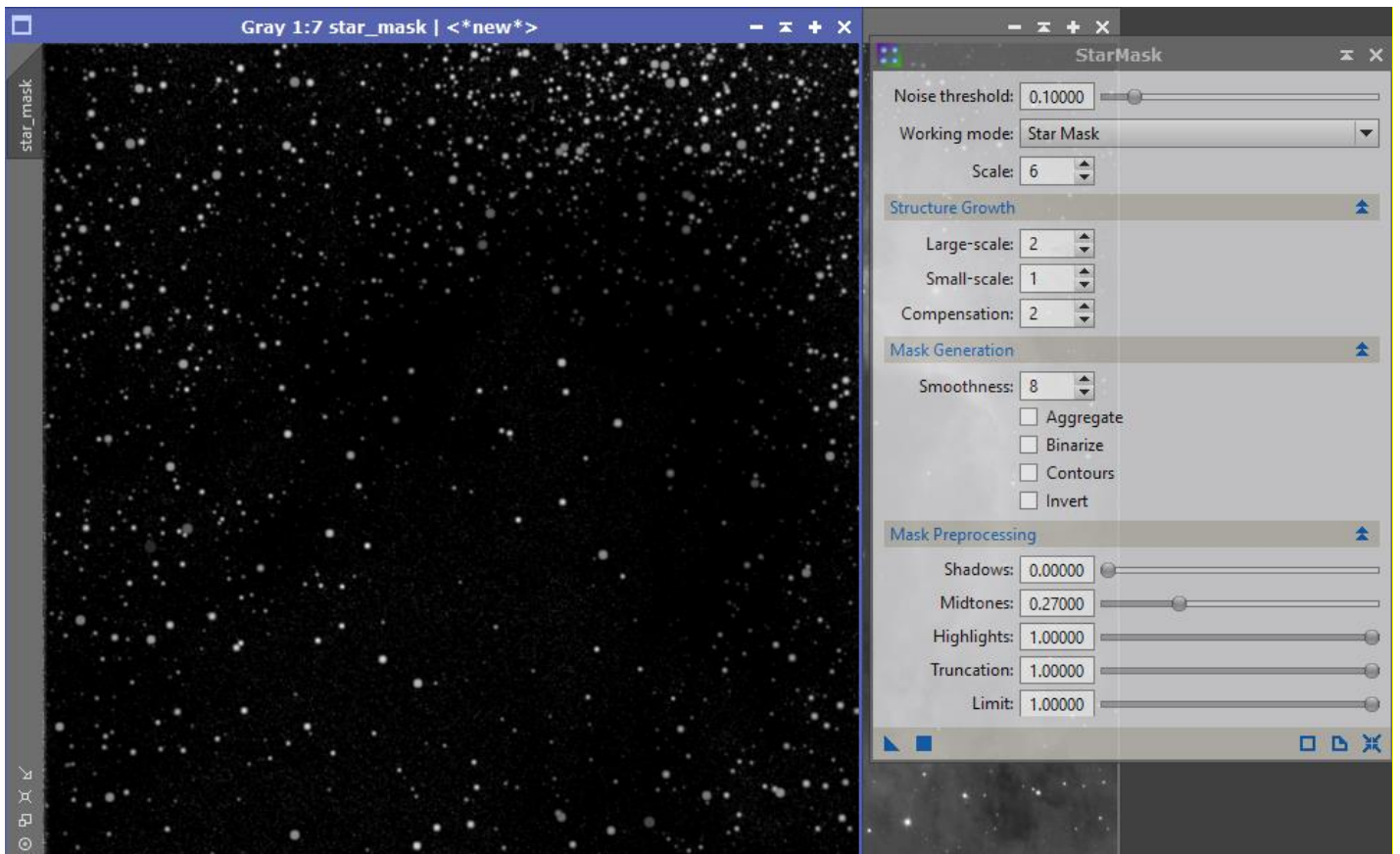
→ Transformation en mode non-linéaire

La montée d'histogramme a été faite en copiant automatiquement les valeurs de la STF.

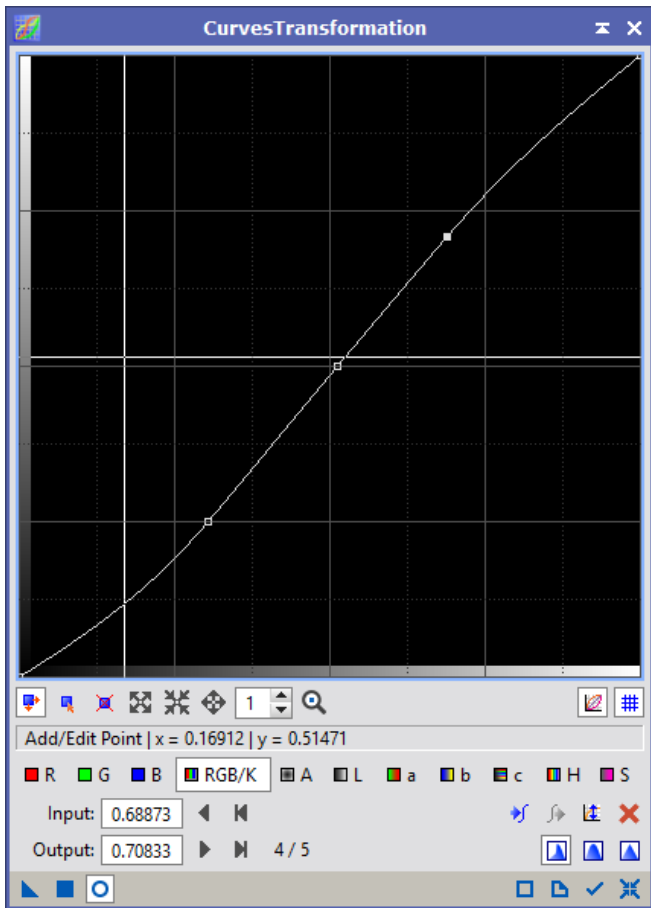


→ Contrastes

Je génère d'abord un masque d'étoiles que j'applique et inverse sur l'image afin de ne pas toucher au contrastes de ces dernières.



J'applique ensuite une légère courbe



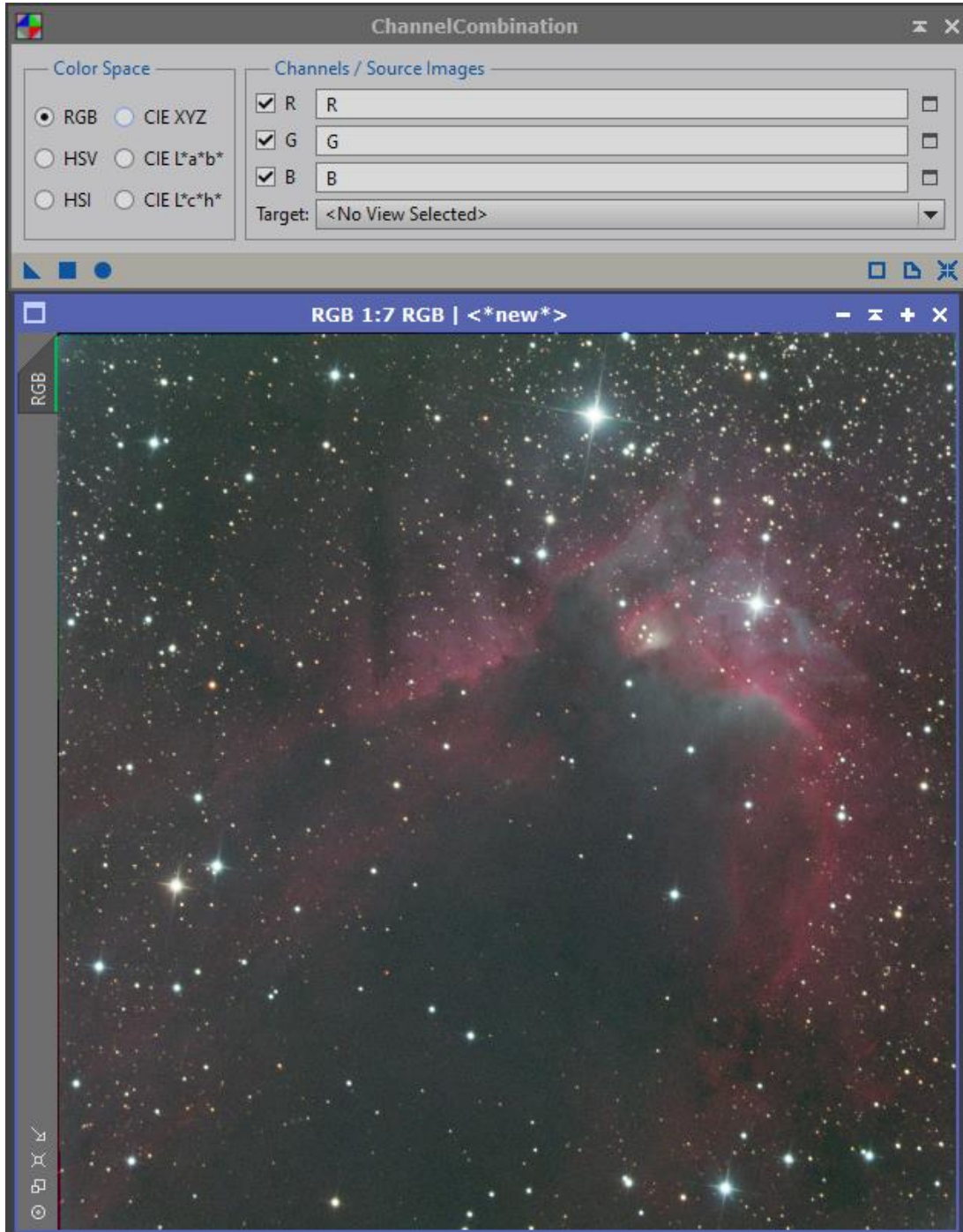
→ Mon image Ha est traitée



→ Traitement des couches RGB

→ Création de l'image RGB

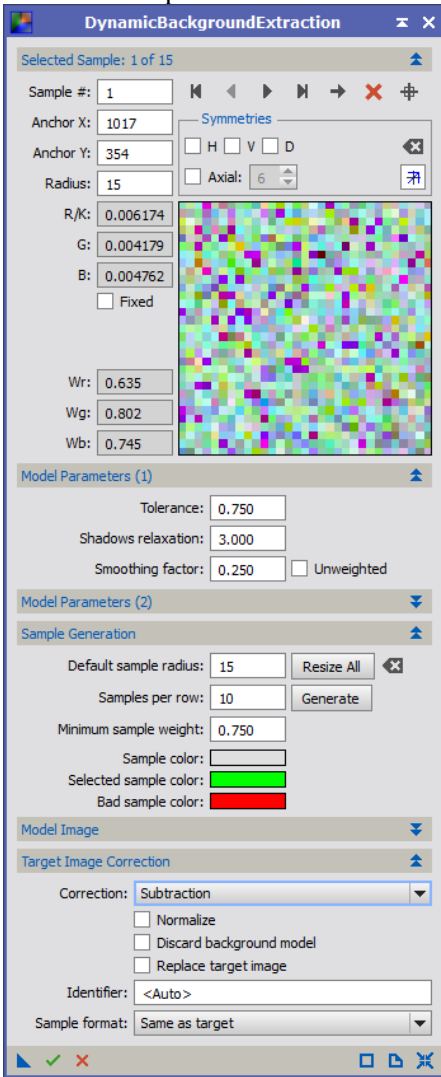
Je crée une image RGB linéaire avec le process *ChannelCombination*.





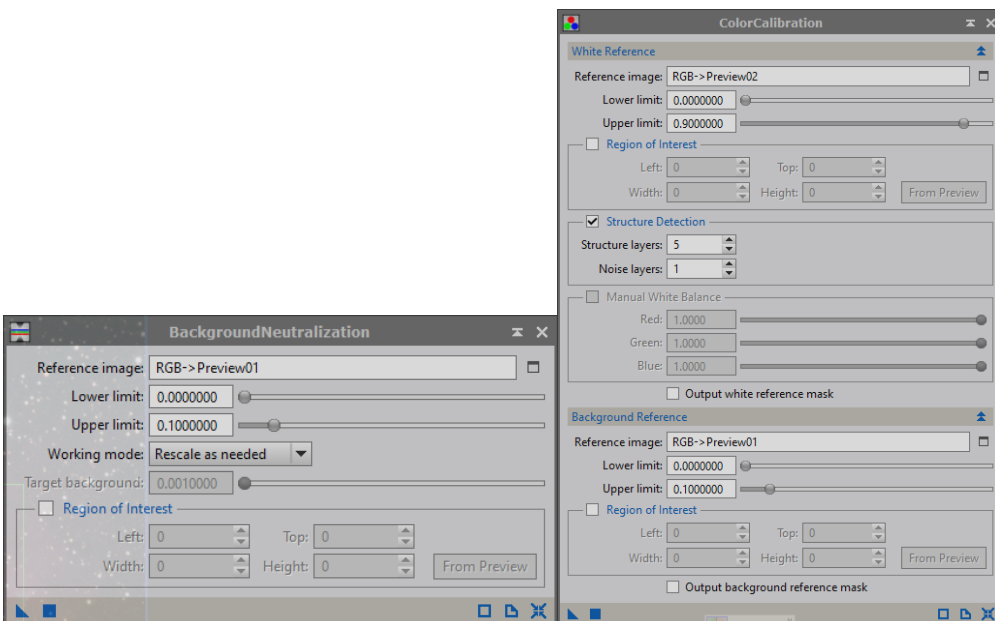
→ Retrait du gradient

Je réutilise le process icon DBE créé pour la couche Ha que j'applique tel quel à mon image RGB



→ Calibration des couleurs

Avec *BackgroundNeutralization* et *ColorCalibration*.



J'enlève ensuite la dominante verte avec **SCNR**.

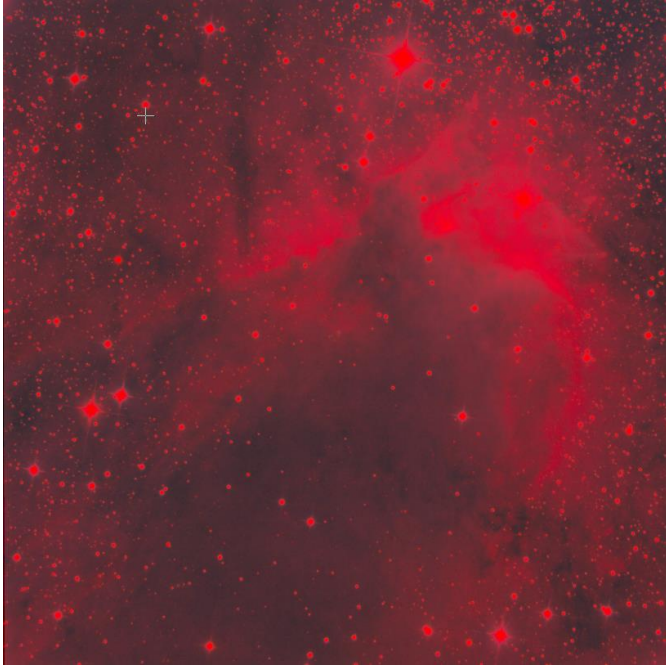
→ Traitement du bruit

Je génère une Luminance pour m'en servir de masque.

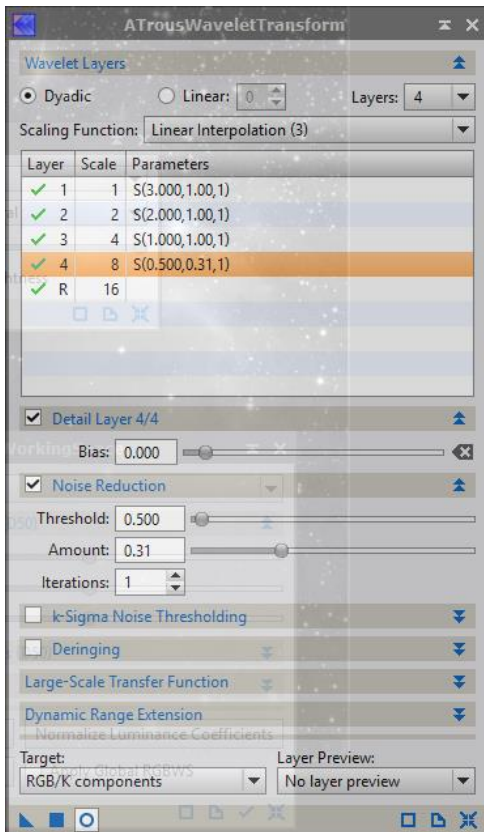
Pour cela, j'équilibre le poids des couleurs avec **RGBWorkingSpace** (valeurs à 33% pour chaque couche), puis j'extrais la couche L.

Je transforme cette luminance en mode non-linéaire et je resserre les histogrammes.

J'applique et inverse ce masque à l'image.



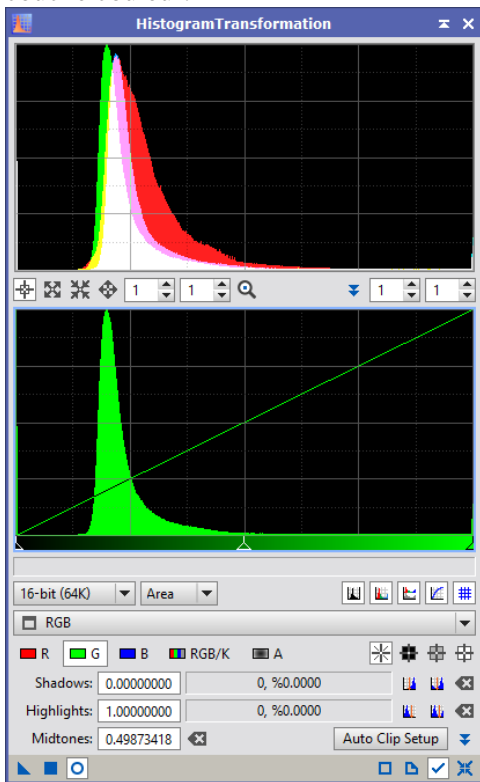
Puis applique ma réduction de bruit avec **AtrousWaveletTransform**





→ Passage en mode linéaire

Je recopie automatiquement les valeurs de la *STF* dans *HistogramTransformation*, puis affine manuellement chaque couche couleur.



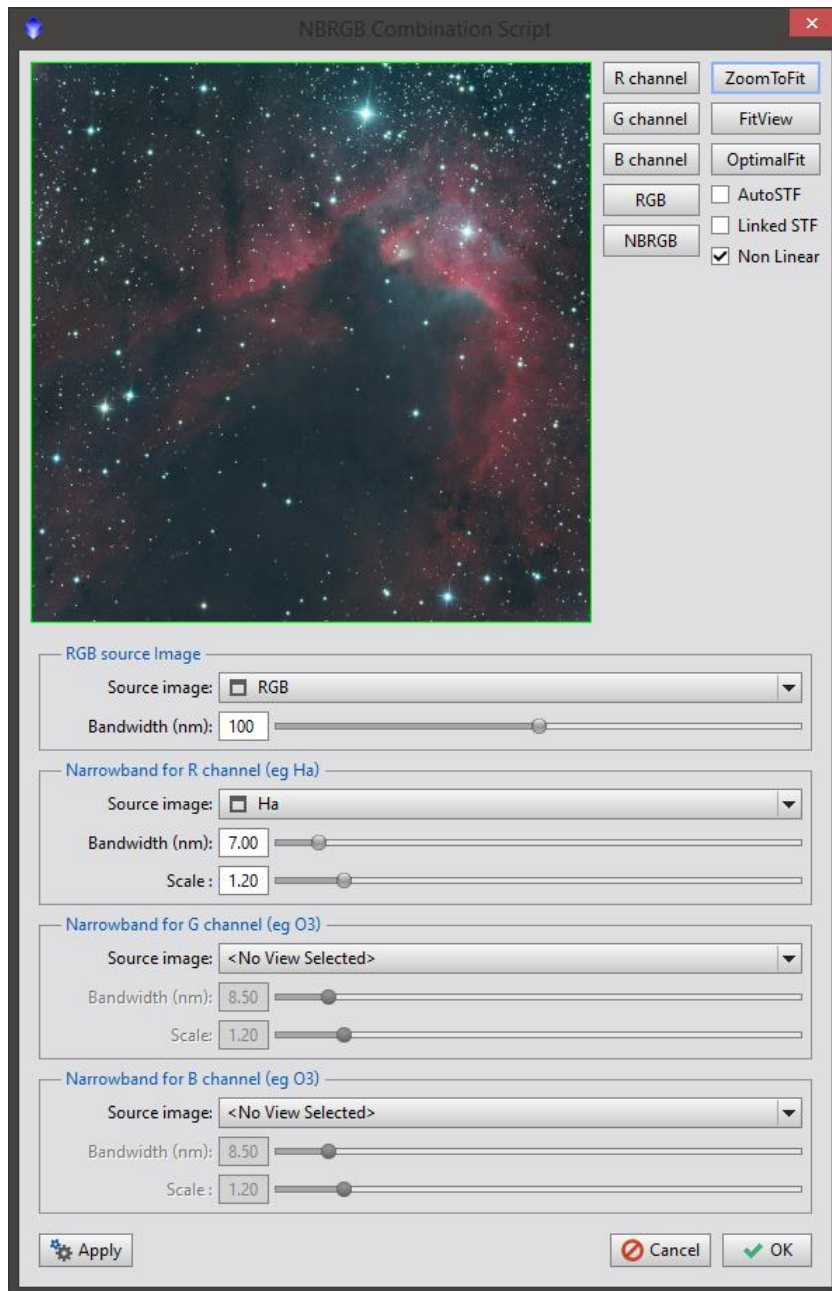
→ Mon image RGB est traitée



## → Combinaison et traitement Ha-RGB

J'aurais souhaité utiliser le script Ha-RVB-AIP mais celui-ci ne semble pas fonctionner. Il reste bloqué.

J'utilise donc le script NBRGB :

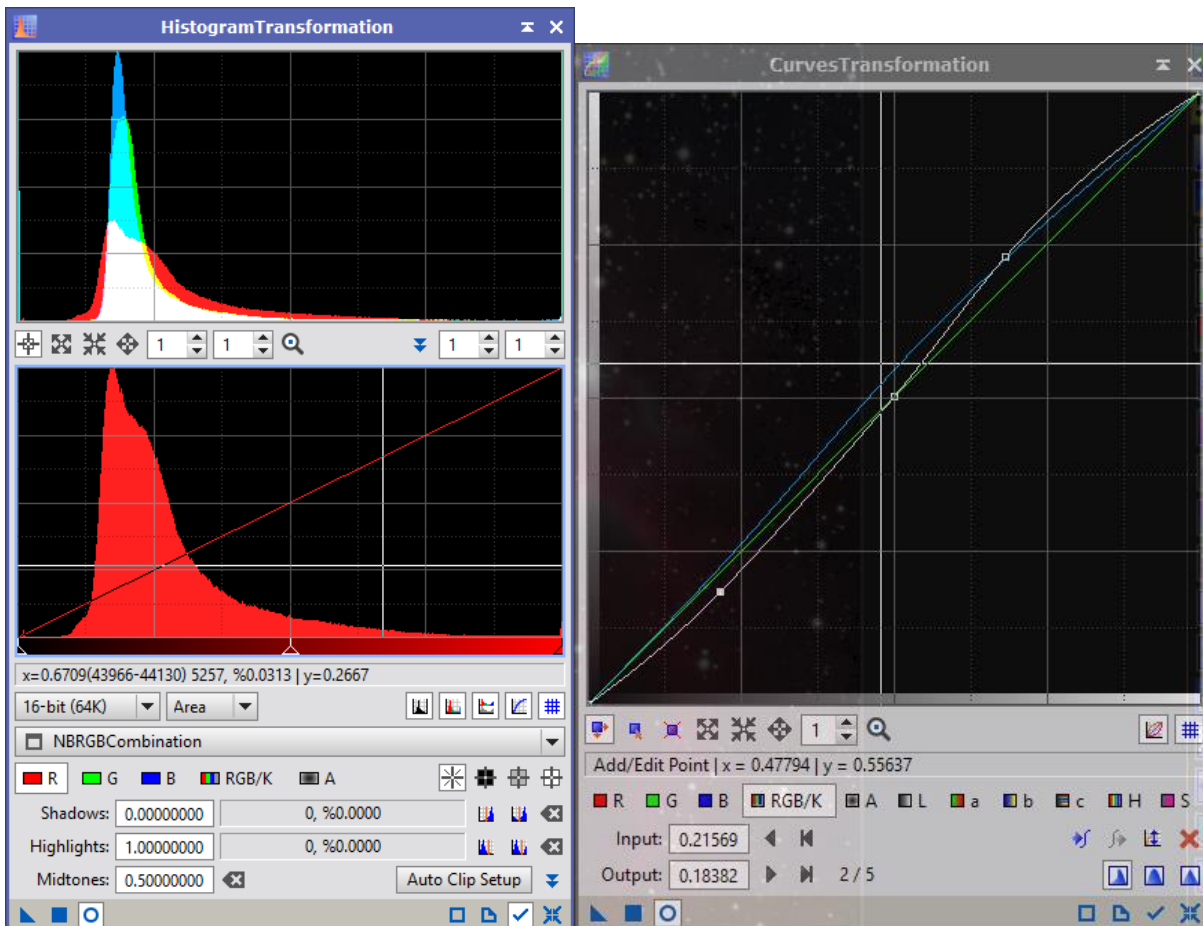


J'ai bien précisé que mes images sources ne sont pas linéaires.

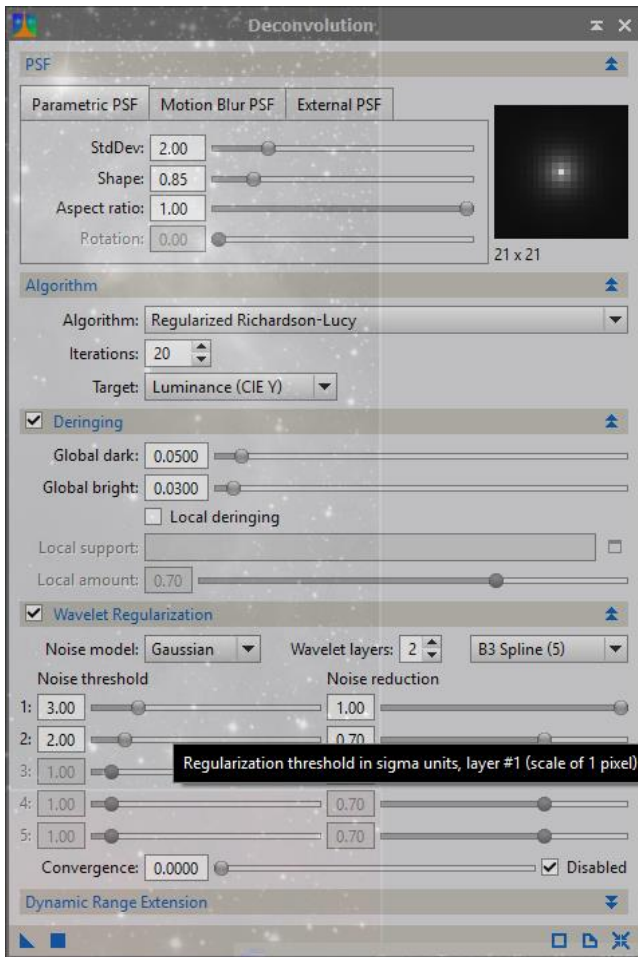
Mon image Ha-RGB est générée mais mérite un traitement un peu plus poussé, en particulier concernant les couleurs qui ont été dénaturées.



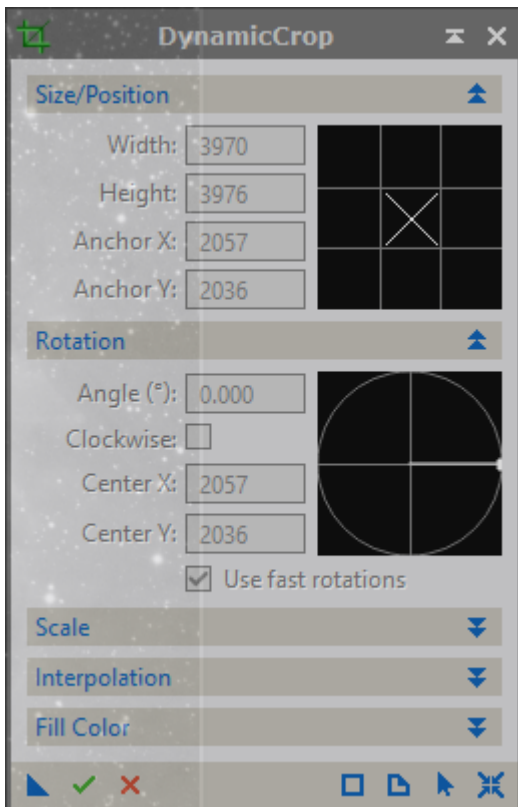
Je recalibre mes couleurs avec l'histogramme et les courbes



Déconvolution manuelle pour faire ressortir les détails :



Je fais ensuite un *DynamicCrop* pour retirer les bandes noires.





Traitement terminé !

