

**Lagune et Trifide**  
**Lionel**



# 1. Prétraitement

- Création des master : dark, bias, flat
- Calibration, alignement, intégration avec *Imagecalibration*, *StarAlignment*, *ImageIntegration*
- Dans PixInsight, par défaut les images sont affublées de suffixes, elles sont mieux rangées si on leur met un préfixe : c\_ pour les images calibrées, r\_c\_ pour les images calibrées puis alignées.

Output Files

Output directory:

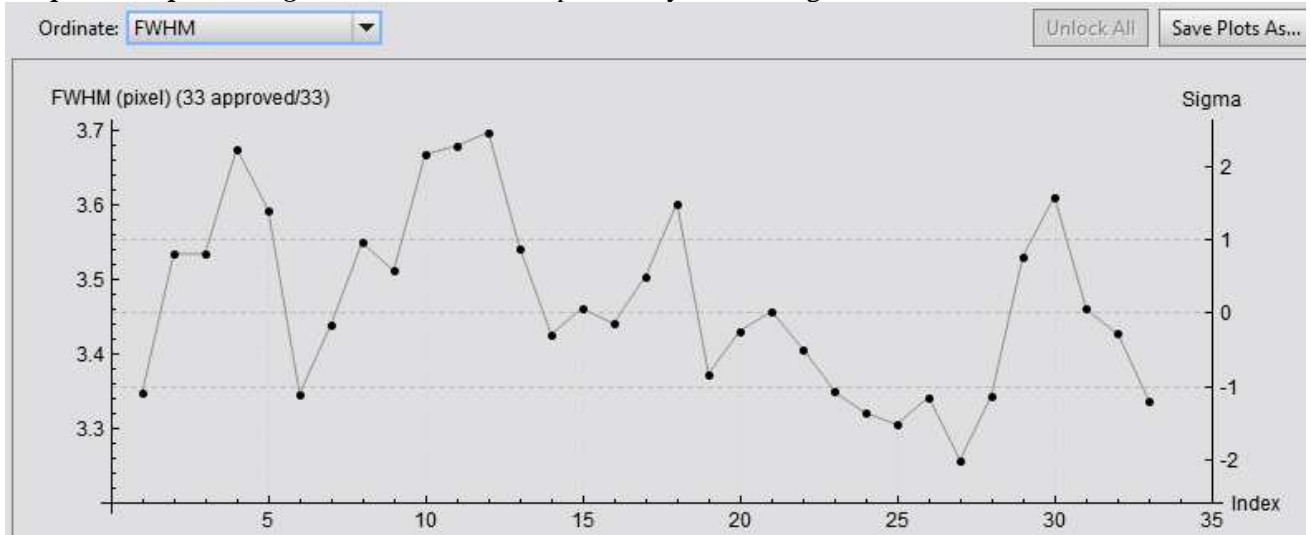
Output extension:  Prefix:  Postfix:

Output Images

Output directory:

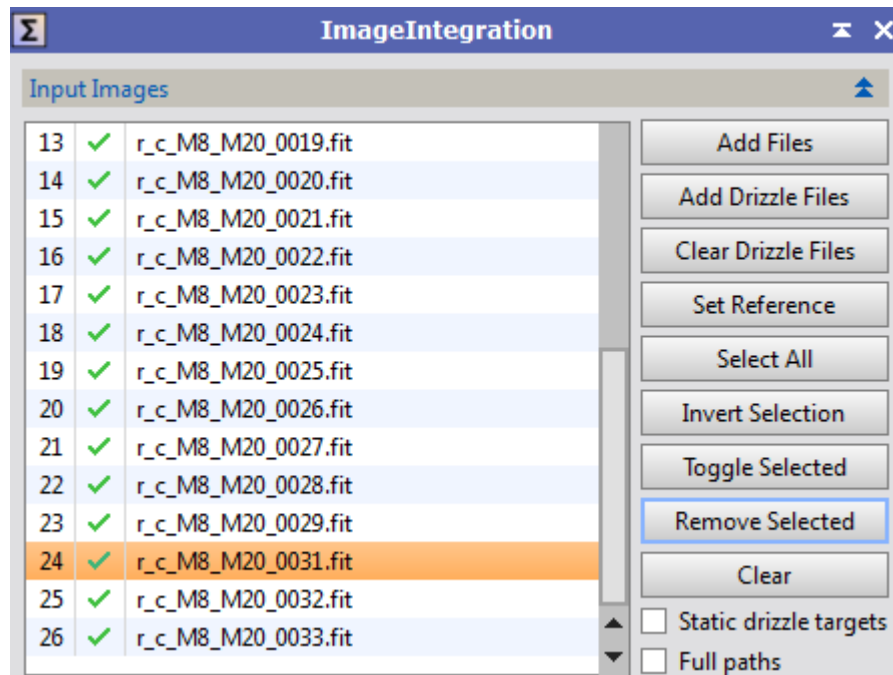
Output extension:  Prefix:  Postfix:  Mask:

- Dans *StarAlignment*, j'utilise toujours la même image de référence, dans ce cas la première image de Luminance, de cette façon, toutes les images de chaque filtre sont déjà alignées.
- Dans *StarAlignment*, j'aligne sur une image binning 1x1, ainsi les images prise en binning 2x2 sont automatiquement redimensionnées à la taille des plus grandes.
- Avant d'intégrer les images les unes sur les autres, je vais choisir les meilleures parmi les 33
- *Script / Batchprocessing / SubFrameSelector* pour analyser les images



En regardant les FWHM, je ne garde que celles sous le trait rouge  
Les images : toutes sauf 4, 5, 9, 10, 11, 18 et 30 qui ont les FWHM les plus élevées.  
Il reste tout de même 26 images.

→ *ImageIntegration*, avec algorithme de réjection : *Winsorized Sigma Clipping*



→ Avec STF, je peux contrôler cette image moyenne, des 26 meilleures, en déliant les couleurs on a même une bonne idée de ce que rendra l'image

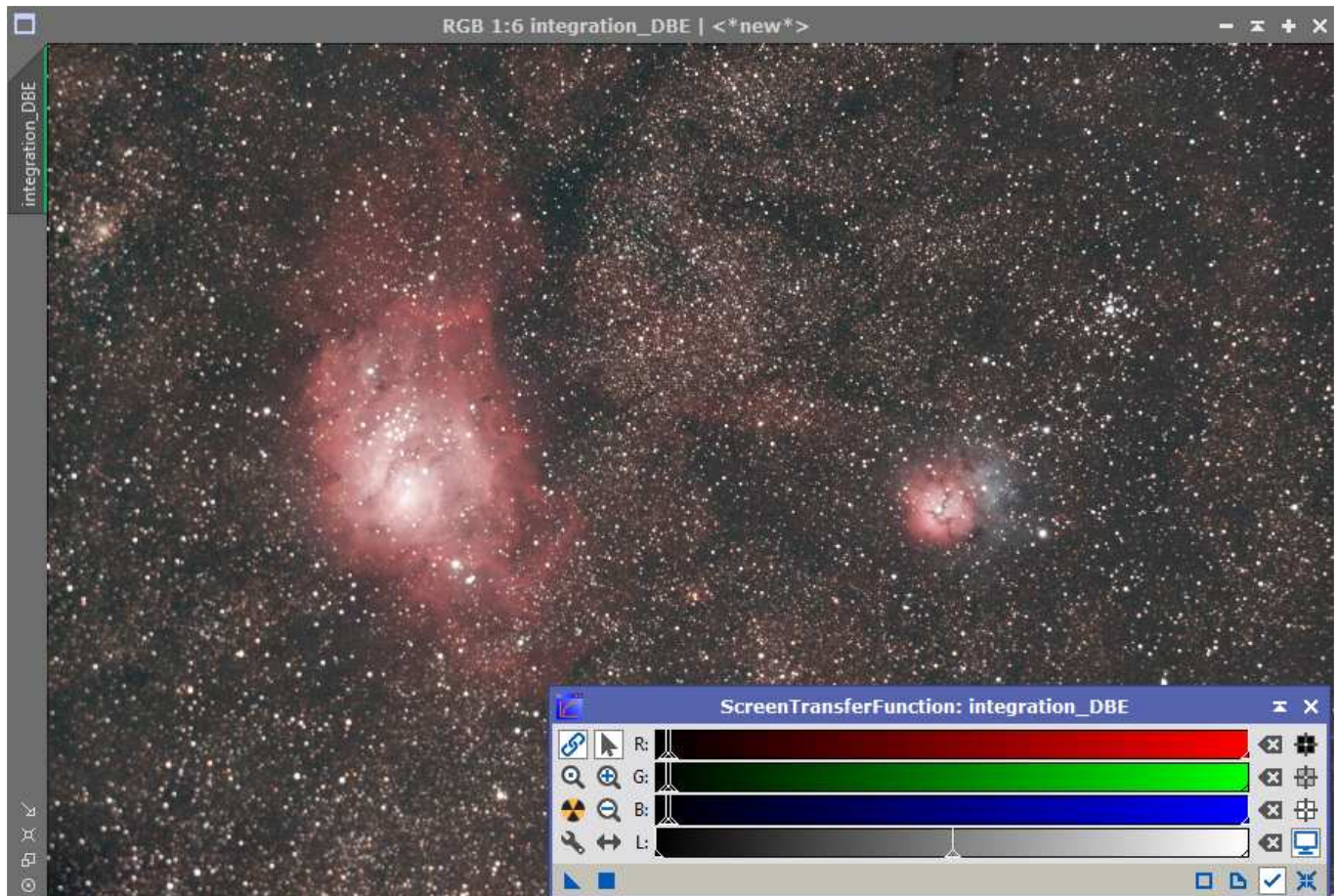


On constate immédiatement l'intérêt de moyenner les images pour diminuer le bruit qui lui est aléatoire. A gauche, une image brute, à droite la moyenne des 26 images.

## 2. Traitement linéaire

### 2.1 Traitement de l'image : les réglages de base

- *DBE* car on voit bien un léger vignettage sur les bords de l'image.
- *Background neutralization*
- *ColorCalibration*



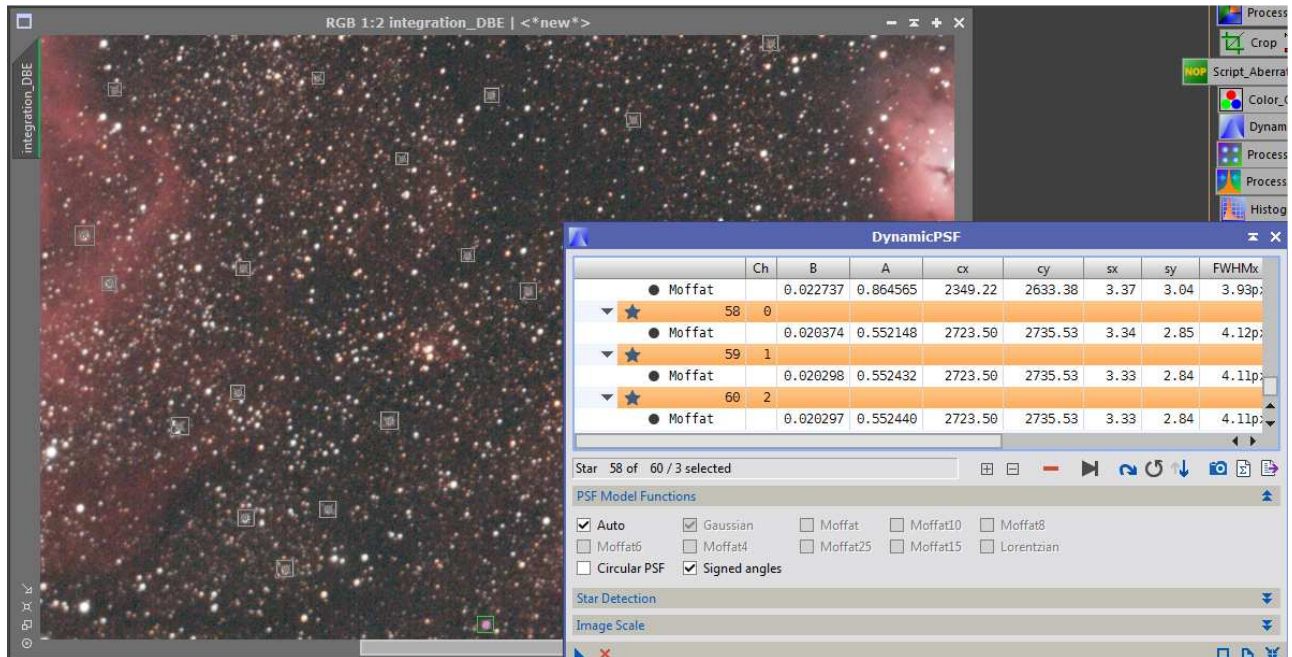
*STF*, avec les canaux liés, on constate que les couleurs sont maintenant bien équilibrées.

A ce stade, l'image n'est que la moyenne des meilleures avec les niveaux bien réglés. Il faut maintenant faire ressortir les détails, c'est là que commencent véritablement les traitements.

## 2.2 Traitement de l'image couleur : suppression du bruit, accentuation des détails

### → *DynamicPSF*

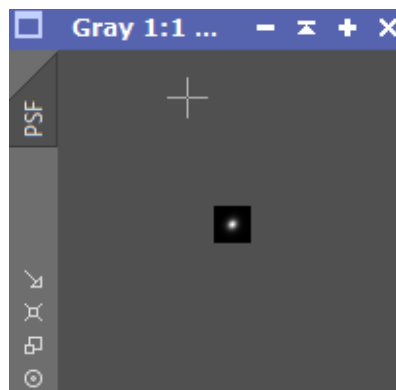
Extraction de la PSF de l'image : les étoiles sont légèrement empâtées en pleine résolution  
On clique sur les étoiles de l'image (une soixantaine)



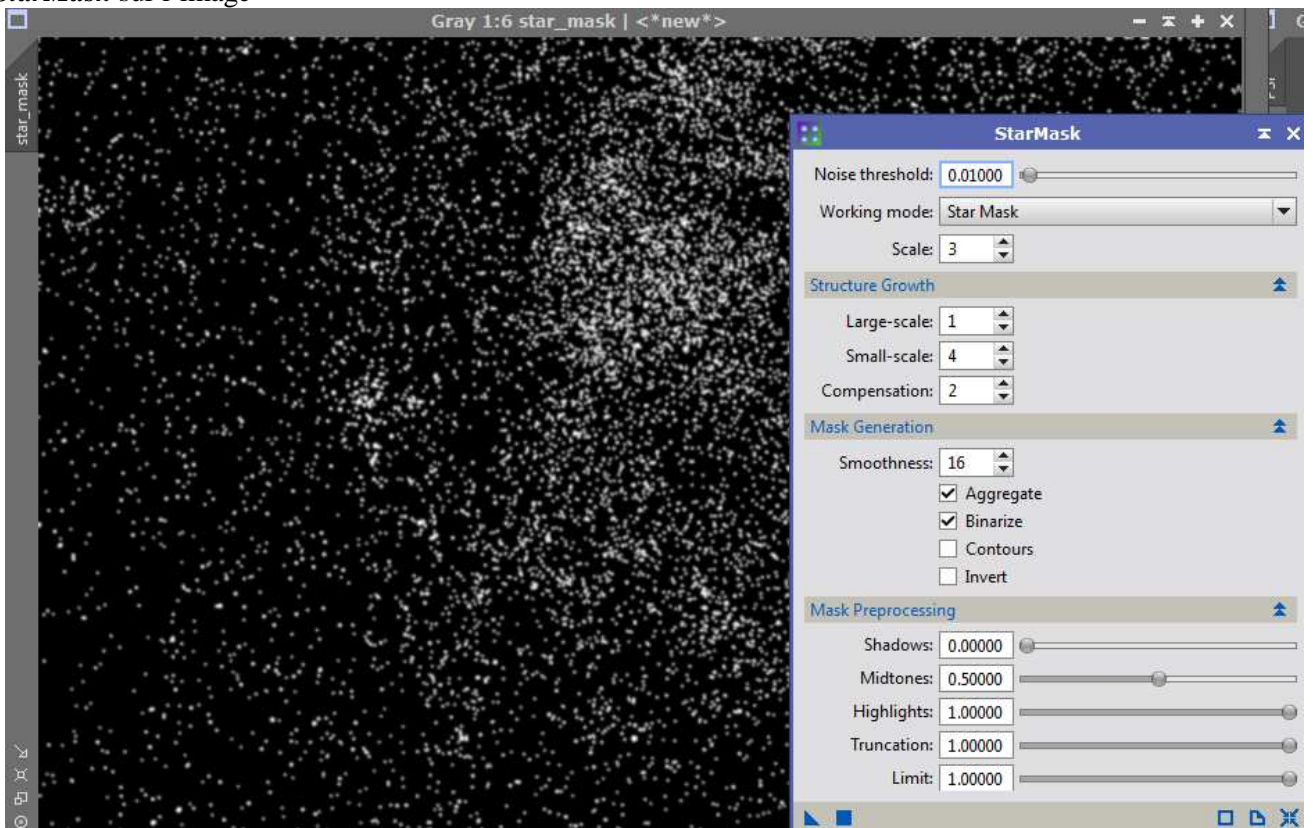
On classe les étoiles de la meilleure à la moins bonne (la flèche vers le bas).

On choisit le critère MAD (le plus à droite, j'ai choisi de ne pas dépasser  $5 \cdot 10^{-3}$ ), Median Absolute Deviation, un bon critère pour juger de la qualité des étoiles.

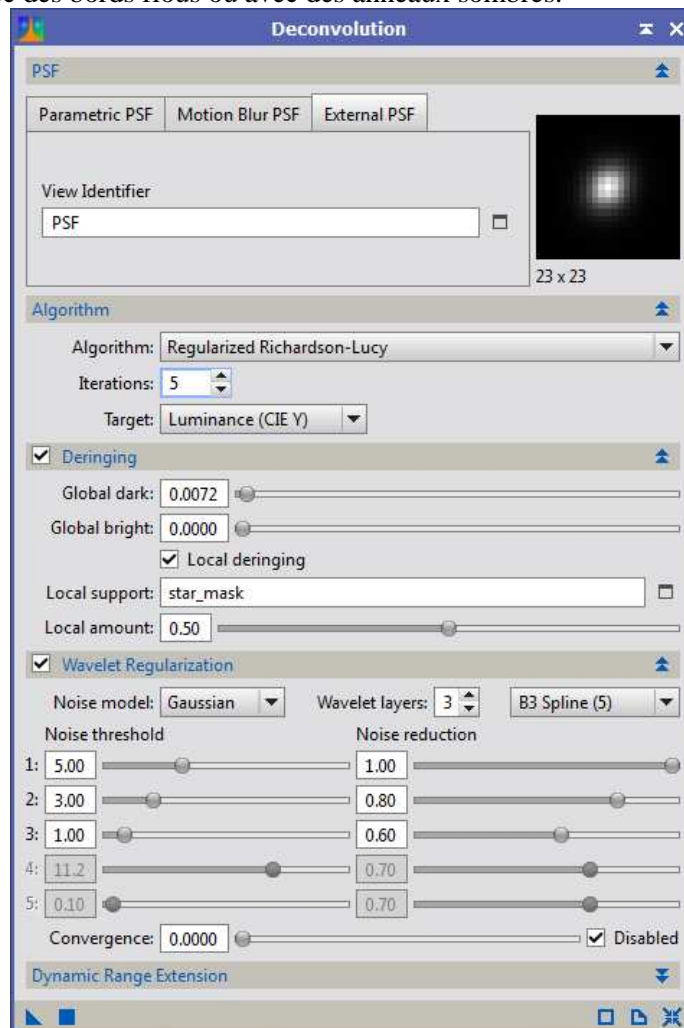
Et on clique sur l'appareil photo pour exporter la PSF de l'image qu'on utilisera pour affiner les détails. La PSF (la fonction point), c'est l'image d'un objet ponctuel (une étoile à l'infini) à travers un instrument, le point idéal est affublé de tout un tas de défauts, notamment la tache d'Airy. Pour retrouver l'image idéale de l'étoile, il faut déconvoluer l'image avec la PSF, l'image de ce que sont devenues les étoiles.

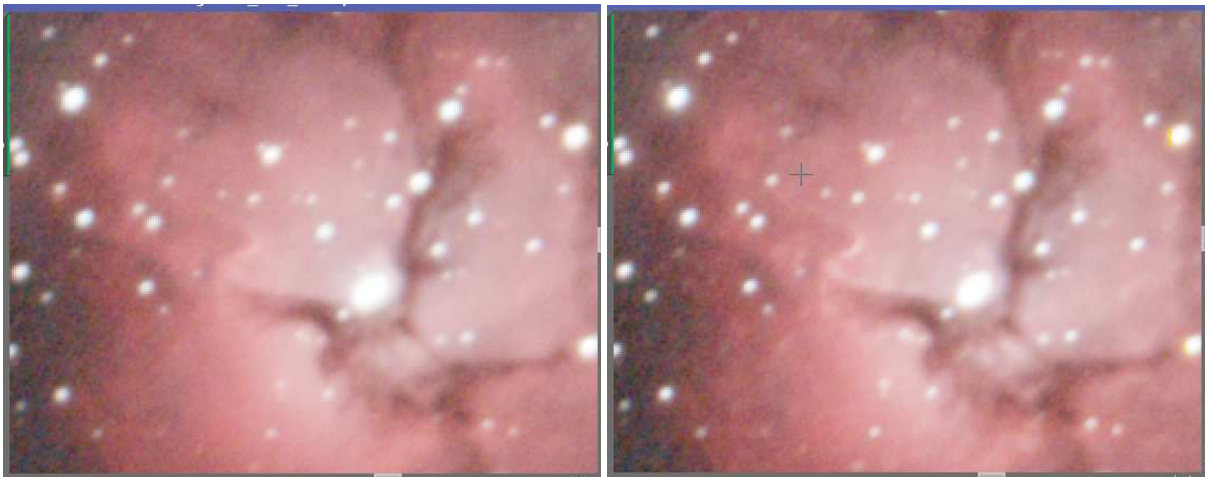


→ *StarMask* sur l'image



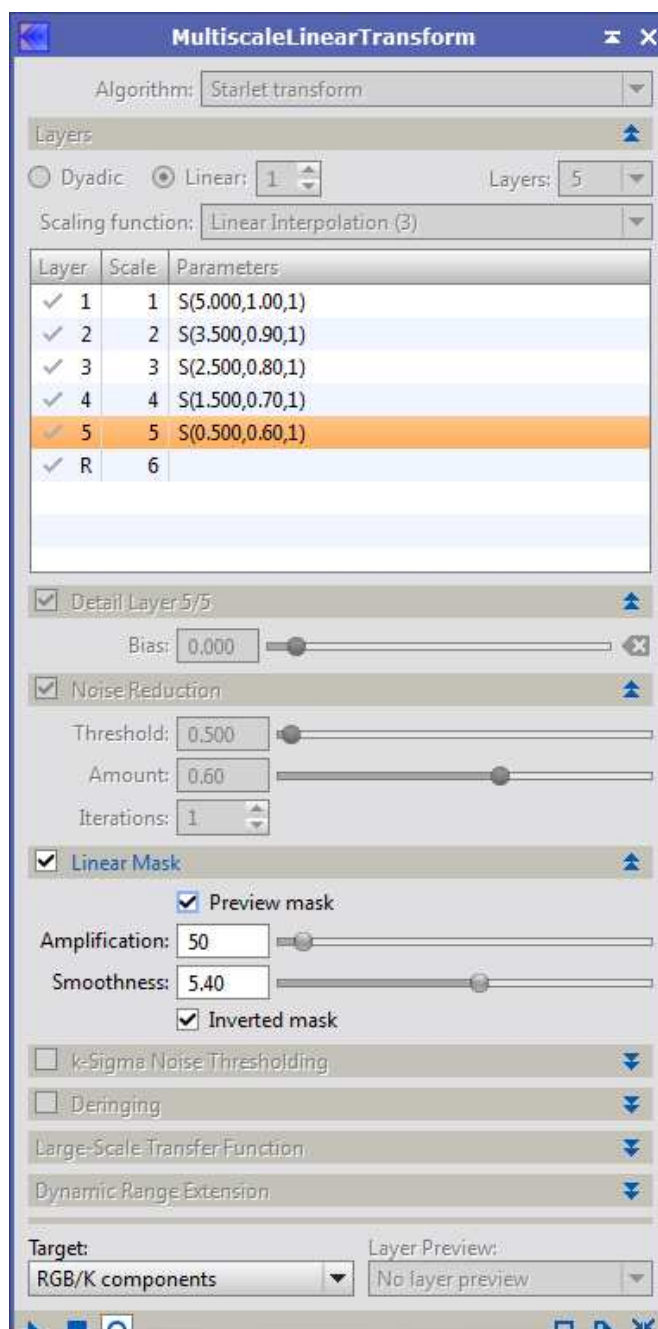
→ *Deconvolution*, pour affiner les étoiles. On utilise la PSF de l'image elle-même pour corriger « correctement » les défauts de l'optique. Réglages à trouver entre le nombre d'itérations, le Global Dark du deringing pour se situer entre des étoiles avec des bords flous ou avec des anneaux sombres.





Les détails ont été affinés, les étoiles sont plus ponctuelles, il faut maintenant corriger le bruit.

- ➔ Réduction du bruit avec **MultiscaleLinearTransform** (on est toujours avec une image linéaire)  
On fait un masque (inversé) pour appliquer la réduction du bruit sur le fond du ciel





On voit le résultat, avant à gauche, après à droite.

### 3. Traitement final

- Extraction de la luminance et décomposition en R,G, B à partir de l'image couleur
- On refait un clone de l'image L et un de l'image R
- Avec *PixelMaths* : (astuce AIP pour dynamiser les contrastes)
  1. clone L =  $0.7L + 0.3R$
  2. clone R =  $0.7R + 0.7L$
  3. L = max(clone L, clone R)
- On recombine l'image couleur avec *LRGBCombination*
- Extraction de la Luminance et binarisation avec *HistogramTransformation* pour créer un masque
- Avec le masque qui protège le fond du ciel pour appliquer le *LocalHistogramEqualization*
- On supprime le masque et on accentue légèrement les couleurs *ColorSaturation*
- *CurveTransformation* pour ajuster les niveaux
- *ADCNR* pour réduire le bruit
- *HistogramTransformation* pour régler les couleurs et fixer les contrastes
- *Crop*, pour couper les bords.
- *Resample* Pour réduire l'énorme résolution de l'image, qui prend un temps fou à traiter. Ça peut se mesurer en heures ou en nombre de litres de thé que j'ai eu le temps de boire pendant que PixInsight faisait les traitements !

