

# AdaptiveStretch

Transformation adaptative non linéaire de l'intensité basée sur les relations pixel à pixel existantes.

## Catégories : Intensité Transformations

Mots-clés : transformation adaptative non linéaire de l'intensité, manipulation des contrastes

## Contenu

### - 1 Description

### - 2 Paramètres

- **2.1 Seuil de bruit** [Noise threshold](#)
- **2.2 Protection des contrastes** [Contrast protection](#)
- **2.3 Points de courbe maximum** [Maximum curve points](#)
- **2.4 Région d'intérêt** [Region of interest](#)

### - 3 Utilisation

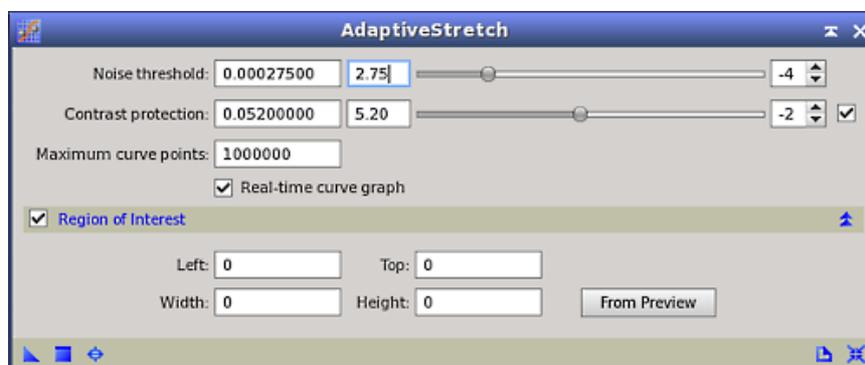
- **3.1 Contrôles exponentiels** [Exponential Controls](#)
- **3.2 Utilisation d'AdaptiveStretch en mode de prévisualisation en temps réel** [Using AdaptiveStretch in Real-Time Preview Mode](#)

#### ■ 3.2.1 Graphiques de courbes [Curve Graphs](#)

### - Références

### - Outils connexes [Related Tools](#)

## 1 Description



**AdaptiveStretch** est un outil général de manipulation du contraste et de la luminosité dans **PixInsight**. Il met en œuvre une transformation non linéaire de l'intensité calculée à partir des relations existantes entre les paires de pixels de l'image cible. Le processus est principalement contrôlé par un seul paramètre de seuil de bruit. Fondamentalement, les différences de luminosité en dessous du seuil de bruit sont considérées comme dues au bruit et à d'autres variations parasites, et sont donc atténuées ou non renforcées. Les

différences de luminosité au-dessus du seuil de bruit sont interprétées comme des changements significatifs dans l'image, de sorte que le processus tend à les accentuer.

Notre implémentation est basée sur l'algorithme décrit par Maria et Costas Petrou. Toutes les différences de luminosité entre des paires de pixels adjacents sont d'abord calculées et classées pour former deux ensembles disjoints : les forces positives, dues à des variations significatives, et les forces négatives, ou variations dues au bruit. Les forces positives tendent à augmenter le contraste de l'image, tandis que les forces négatives agissent dans le sens opposé, protégeant l'image. Le processus utilise ces deux forces pour calculer une courbe de cartographie non linéaire, qui est ensuite utilisée pour transformer l'image par interpolation.

La liste suivante résume les principales caractéristiques et les avantages de l'outil

### **AdaptiveStretch** :

- Il s'agit essentiellement d'un processus à un seul paramètre (un deuxième paramètre de limitation du contraste est facultatif).
- La courbe de transformation calculée est optimale, en ce sens qu'elle maximise le contraste sans intensifier les données parasites.
- Elle permet de trouver très facilement une courbe optimale de manière totalement objective et non biaisée. La même courbe peut être très difficile à trouver en utilisant des ajustements manuels.
- La transformation garantit qu'aucun pixel ne sera coupé, ni dans les ombres ni dans les hautes lumières. Les pixels initialement coupés, ou les pixels qui sont soit noirs soit blancs dans l'image originale, seront préservés dans l'image transformée.
- Offre une fonctionnalité complète de prévisualisation en temps réel, y compris la génération en temps réel de graphiques de courbes.
- Dans les cas difficiles, une région d'intérêt (**ROI**) peut être définie pour limiter l'analyse des variations de pixels par paires à une zone significative de l'image.
- La courbe de cartographie calculée peut être convertie automatiquement en une instance de **CurvesTransformation**.
- Peut être utilisé pour étirer des données CCD linéaires brutes monochromes et des données couleur OSC CCD / DSLR.
- Fournit d'excellents résultats avec des images brutes astronomiques et de lumière du jour.

Avec **AdaptiveStretch**, nous n'avons pas l'intention de remplacer ces outils fondamentaux, mais de fournir une autre façon d'effectuer des manipulations de luminosité/contraste, y compris l'étape initiale d'étirement non linéaire des données brutes. La caractéristique la plus intéressante de l'**AdaptiveStretch** est qu'il fonctionne en analysant le véritable contenu de l'image. D'autres outils nécessitent un travail purement manuel, et leurs résultats dépendent donc davantage des connaissances et de la capacité de l'utilisateur à comprendre l'image. Il peut être difficile de comprendre les relations subtiles entre les différentes structures d'image et, en ce sens, **AdaptiveStretch** peut être considéré comme un outil puissant pour des analyses objectives.

## **2 Paramètre**

### **2.1 Seuil de bruit**

Le processus **AdaptiveStretch** calcule une courbe non linéaire pour renforcer les différences de luminosité existantes au-dessus du paramètre de seuil de bruit. Les différences de luminosité inférieures au seuil de bruit seront diminuées ou non améliorées dans l'image traitée. Diminuez ce paramètre pour appliquer une transformation non linéaire plus agressive. En général, vous devez affiner ce paramètre ainsi que la protection du contraste pour trouver une combinaison optimale.

Figure 1



Comparaison de différents résultats obtenus en faisant varier le paramètre de seuil de bruit de l'**AdaptiveStretch**. L'image originale est une image DSLR brute 16 bits prise avec un appareil photo Canon EOS 450D.

(a) En haut à gauche : Image linéaire dé-balayée.

(b) En haut à droite : Seuil de bruit =  $1 \times 10^{-2}$

(c) En bas à gauche : Seuil de bruit =  $3 \times 10^{-3}$

(d) En bas à droite : Seuil de bruit =  $5 \times 10^{-4}$

## 2.2 Protection des contrastes

Le paramètre de protection contre les contrastes limite l'augmentation du contraste sur les zones très claires et très sombres. L'augmentation de ce paramètre peut aider à contrôler l'intensification excessive des hautes lumières et l'assombrissement des ombres. Quelle que soit sa valeur, ce paramètre peut être désactivé ou activé ; il est désactivé par défaut. Cette fonction permet de comparer rapidement les résultats avec et sans protection de contraste appliquée.

Figure 2



**Un exemple d'utilisation du paramètre de protection contre les contrastes de l'AdaptiveStretch.** Comme dans l'exemple précédent, l'original est une image brute 16 bits prise avec un appareil photo Canon EOS 450D.

**(a)** En haut à gauche : Image linéaire dé-balayée.

**(b)** En haut à droite : Seuil de bruit =  $1 \times 10^{-3}$ , pas de protection contre le contraste.

**(c)** En bas à gauche : Seuil de bruit =  $5 \times 10^{-3}$ , pas de protection contre les contrastes.

**(d)** En bas à droite : Seuil de bruit =  $1 \times 10^{-4}$ , protection de contraste =  $8 \times 10^{-2}$

### 2.3 Points de courbe maximum

C'est le nombre maximum de points calculés pour construire la courbe de transformation. La valeur par défaut est de 106 points. Pour les images de nombres entiers de 8 et 16 bits, **AdaptiveStretch** peut calculer un maximum de 256 et 65536 points de courbe. Notez que dans ces cas, calculer plus de points de courbe n'aurait aucun sens car la différence minimale possible non nulle entre deux pixels est une unité discrète. Pour les images de nombres entiers de 32 bits, la limite théorique est de 232 points de courbe, ce qui n'est pas pratique pour des raisons évidentes.

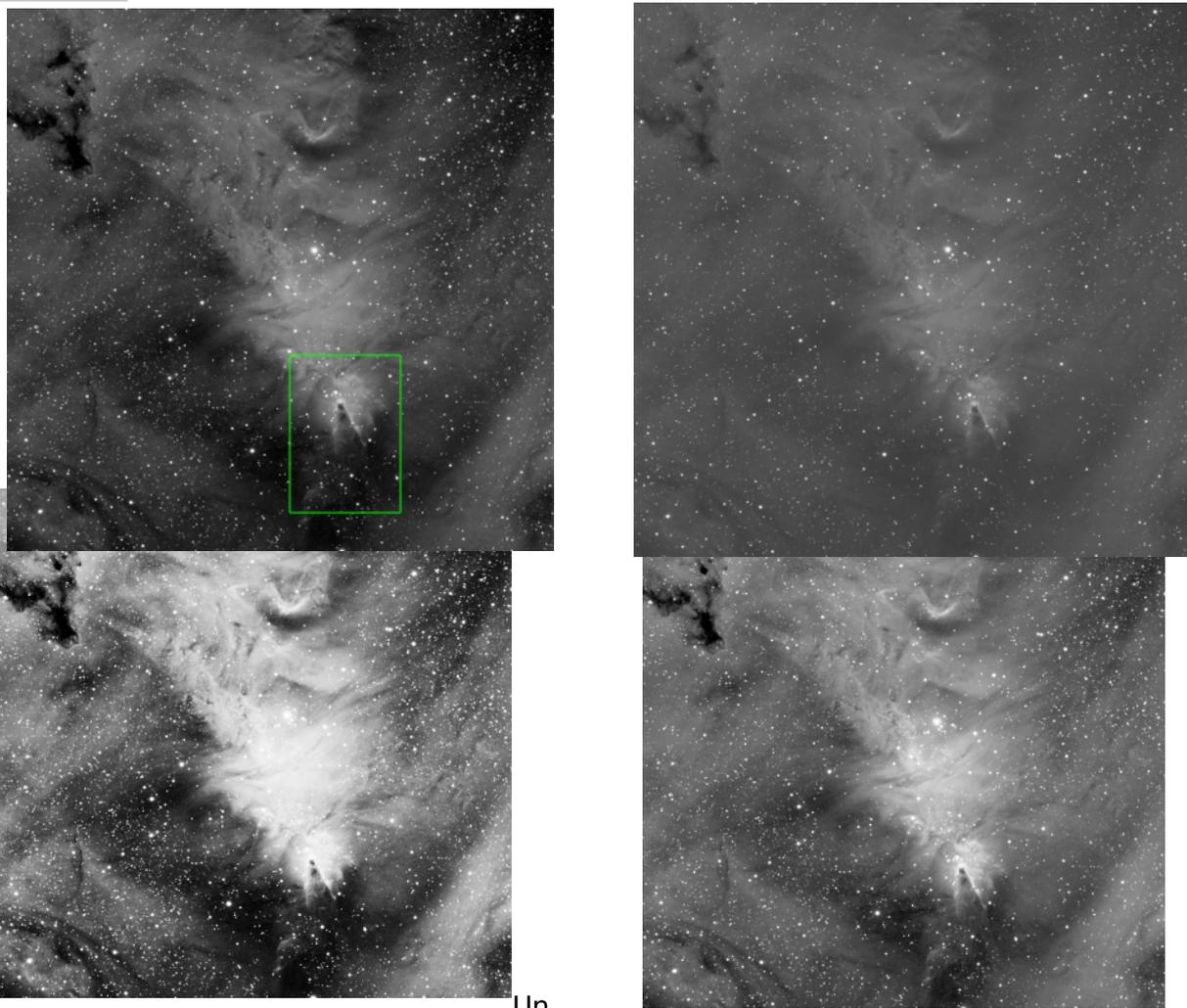
Pour les images en virgule flottante 32 bits et 64 bits, on peut calculer un maximum de 107 et 108 points de courbe, respectivement. Notez que la réduction du nombre de points de courbe peut limiter la capacité de l'algorithme à détecter de très petits changements de luminosité entre des pixels adjacents, ce qui est essentiel pour tenir compte des variations dues au bruit dans l'image. En pratique, cependant, la limite par défaut de 106 valeurs discrètes est plus que suffisante pour la plupart des images en virgule flottante de 32 bits. L'augmentation de ce paramètre peut améliorer le résultat pour les images très riches en virgule flottante et en nombres entiers de 32 bits, résultant généralement de

compositions **HDR** profondes. Gardez à l'esprit que les grandes courbes augmentent considérablement les temps de calcul, elles ne doivent donc être utilisées que lorsque cela est réellement nécessaire. Dans la plupart des cas, la valeur par défaut de ce paramètre est tout à fait appropriée.

## 2.4 Région d'intérêt

Par défaut, le processus **AdaptiveStretch** analyse les différences de luminosité des pixels par paire sur l'ensemble de l'image cible. Normalement, c'est la meilleure option s'il n'y a pas de grandes différences de luminosité, ou si certaines grandes différences peuvent être ignorées sans risque. Un exemple typique est une scène de jour en intérieur où nous décidons qu'il est correct de brûler complètement une fenêtre. Cependant, il existe des cas plus difficiles où ces simples suppositions sont inacceptables, ou bien où elles ne fonctionnent tout simplement pas. Dans ces cas, une région d'intérêt (**ROI**) peut être définie pour restreindre l'analyse à une région rectangulaire de l'image. Le **ROI** est défini par les coordonnées de pixels X et Y de son coin supérieur gauche, et par sa largeur et sa hauteur en pixels. Le processus ignorera un **ROI** de moins de 16×16 pixels, ou un **ROI** qui se trouve complètement à l'extérieur de l'image.

Figure 3



Un exemple de l'utilisation des régions d'intérêt (**ROI**) pour contrôler le processus **AdaptiveStretch**. Dans l'ensemble, cette image est assez plate, ou en d'autres termes, elle est dominée par une quantité écrasante de pixels avec des valeurs intermédiaires. Ceci est plus évident sur le résultat d'une transformation d'histogramme présentée dans la

section (b). L'utilisation de l'ensemble de l'image pour analyser les différences de pixels par paire donne un résultat trop contrasté (c), où les zones les plus sombres sont trop sombres et les zones les plus claires trop plates. Cela se produit parce que l'algorithme **AdaptiveStretch** calcule une transformation pour optimiser le contraste pour un sous-ensemble dominant des pixels analysés.

En limitant les analyses à une zone plus petite, où les structures claires et sombres sont également bien échantillonnées, et où les valeurs intermédiaires des pixels ne sont plus dominantes, le résultat (d) nous permet de montrer beaucoup mieux toute la gamme des structures de l'image. C'est un bel exemple de dynamique d'image complexe. Notez cependant que le résultat obtenu dans la section (c) peut être amélioré en utilisant des techniques de compression **HDR** dans **PixInsight**, par exemple avec l'outil **HDRMultiscaleTransform**, qui ouvre une voie de traitement complètement différente. Avec l'aimable autorisation d'Ivette Rodríguez et Oriol Lehmkuhl.

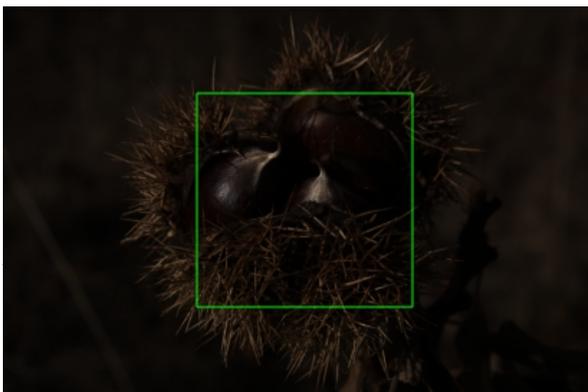
**(a)** En haut à gauche : L'image linéaire avec un étirement automatique de l'écran (**STF**) appliqué (notez que la transformation **STF** automatique coupe les pixels dans les ombres).

**(b)** En haut à droite : Résultat obtenu avec l'outil **HistogramTransformation** en coupant les sections inutilisées de l'histogramme aux deux extrémités et en appliquant une valeur d'équilibre des tons moyens de 0,005.

**(c)** En bas à gauche : Résultat de l'**AdaptiveStretch**, seuil de bruit =  $10^{-4}$ , courbe calculée pour l'ensemble de l'image (aucune région d'intérêt).

**(d)** En bas à droite : résultat de l'**AdaptiveStretch**, seuil de bruit =  $10^{-4}$ , courbe calculée pour l'ensemble de l'image (aucune région d'intérêt) : Résultat de l'**AdaptiveStretch**, seuil de bruit =  $1,2 \times 10^{-4}$ , région d'intérêt fixée au rectangle vert indiqué au point (a).

Figure 4





autre exemple de **AdaptiveStretch** avec une région d'intérêt (**ROI**). Dans cet exemple, un **ROI** couvrant le sujet principal a été utilisé pour empêcher l'augmentation excessive du contraste de l'image traitée. Le manque relatif de saturation des couleurs peut être facilement compensé par une courbe de saturation des couleurs.

**(a)** En haut à gauche : L'image linéaire. Il s'agit d'une image brute de reflex numérique sans baies, acquise avec un appareil photo Canon 450D.

**(b)** En haut à droite : Résultat de l'**AdaptiveStretch** avec un seuil de bruit =  $8,7 \times 10^{-4}$  et une protection contre le contraste = 0,355, pas de retour sur investissement.

**(c)** En bas à gauche : Résultat de l'**AdaptiveStretch**, seuil de bruit =  $6,7 \times 10^{-4}$ , protection de contraste = 0,375, ROI réglé sur le rectangle vert indiqué au point (a).

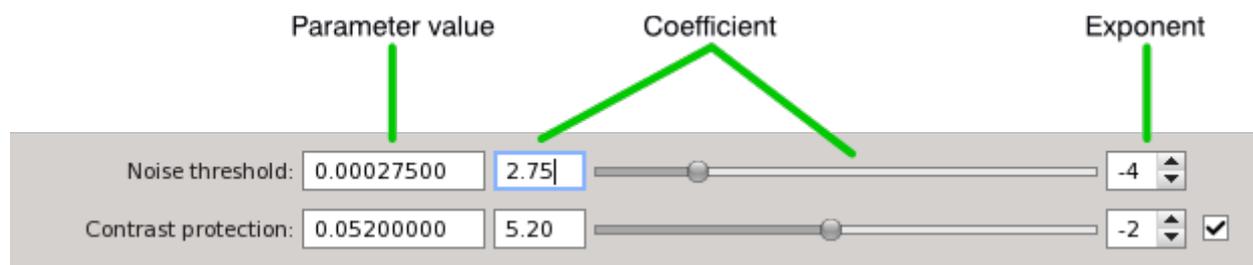
**(d)** En bas à droite : résultat de l'**AdaptiveStretch**, seuil de bruit =  $6,7 \times 10^{-4}$ , protection contre le contraste = 0,375, **RCI** réglé sur le rectangle vert indiqué au point (a) : Le résultat de (c) après une courbe de saturation des couleurs appliquée avec **CurvesTransformation**.

### 3 Usage

#### 3.1 Contrôles exponentiels

Le seuil de bruit et les paramètres de protection contre les contrastes de l'**AdaptiveStretch** sont extrêmement sensibles. Ils peuvent varier de zéro à un avec une précision fractionnaire de huit chiffres. Cette précision étendue est nécessaire pour traiter les images en virgule flottante et en nombres entiers de 32 bits. Pour les images d'entiers de 16 bits, il faut généralement 4 ou 5 chiffres décimaux. Par conséquent, les commandes habituelles combinées d'édition et de curseur qui sont omniprésentes dans la plateforme **PixInsight**, sont tout simplement trop grossières pour contrôler ces paramètres.

Pour ces paramètres, nous avons mis en place un contrôle spécialisé avec trois sous-contrôles : un champ d'édition pour entrer la valeur du paramètre avec 8 chiffres décimaux ; un combo édition/glisser pour spécifier le coefficient en notation scientifique, avec 2 chiffres décimaux dans la plage de 1 à 9,99 ; et une boîte de rotation pour spécifier l'exposant dans la plage de 0 à -8.

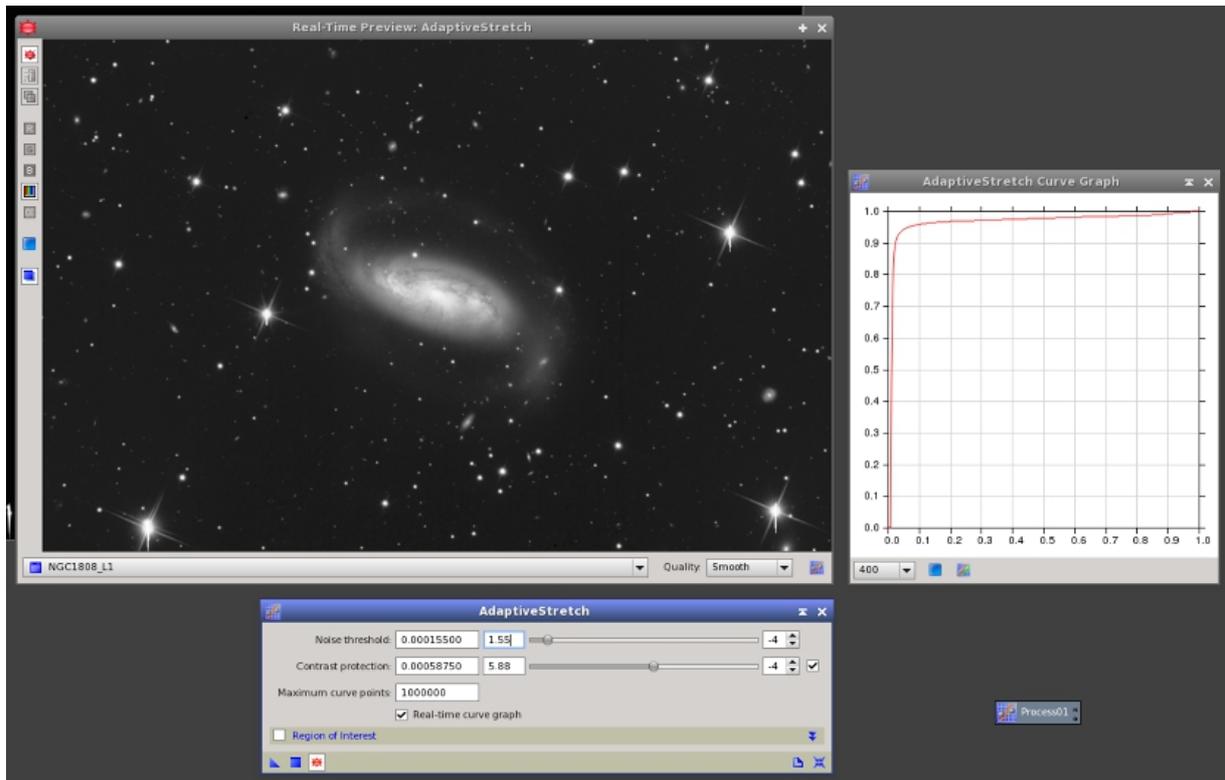


Par exemple, dans la figure ci-dessus, la valeur du paramètre de seuil de bruit est de 0,000275, ce qui équivaut à  $2,75 \times 10^{-4}$  en notation scientifique. En augmentant et en diminuant l'exposant d'une unité, vous pouvez rapidement changer l'échelle du paramètre d'un ordre de grandeur. En déplaçant le curseur, vous pouvez régler le coefficient avec précision. Cela vous permet de contrôler un paramètre sensible très facilement avec la souris sans avoir besoin d'écrire des valeurs numériques.

### 3.2 Utilisation d'AdaptiveStretch en mode de prévisualisation en temps réel

Bien que ce soit un processus plutôt automatique, **AdaptiveStretch** est un outil très interactif. En fait, vous ne pouvez pas appliquer cet outil correctement si vous n'affinez pas ses deux paramètres principaux au cours d'une session de prévisualisation en temps réel. La raison en est qu'il s'agit d'un outil adaptatif : il calcule une transformation basée sur les différences de luminosité existantes entre des paires de pixels adjacents dans l'image cible. Bien sûr, vous pouvez aussi ajuster les paramètres de l'**AdaptiveStretch** en utilisant les prévisualisations standard dans **PixInsight**, ce qui vous permet d'obtenir des résultats très précis, mais la prévisualisation en temps réel est tout simplement plus pratique et plus facile à utiliser. Une fois que vous avez trouvé de bons paramètres avec la prévisualisation en temps réel, rien ne vous empêche de les affiner en utilisant des prévisualisations standard, si nécessaire.

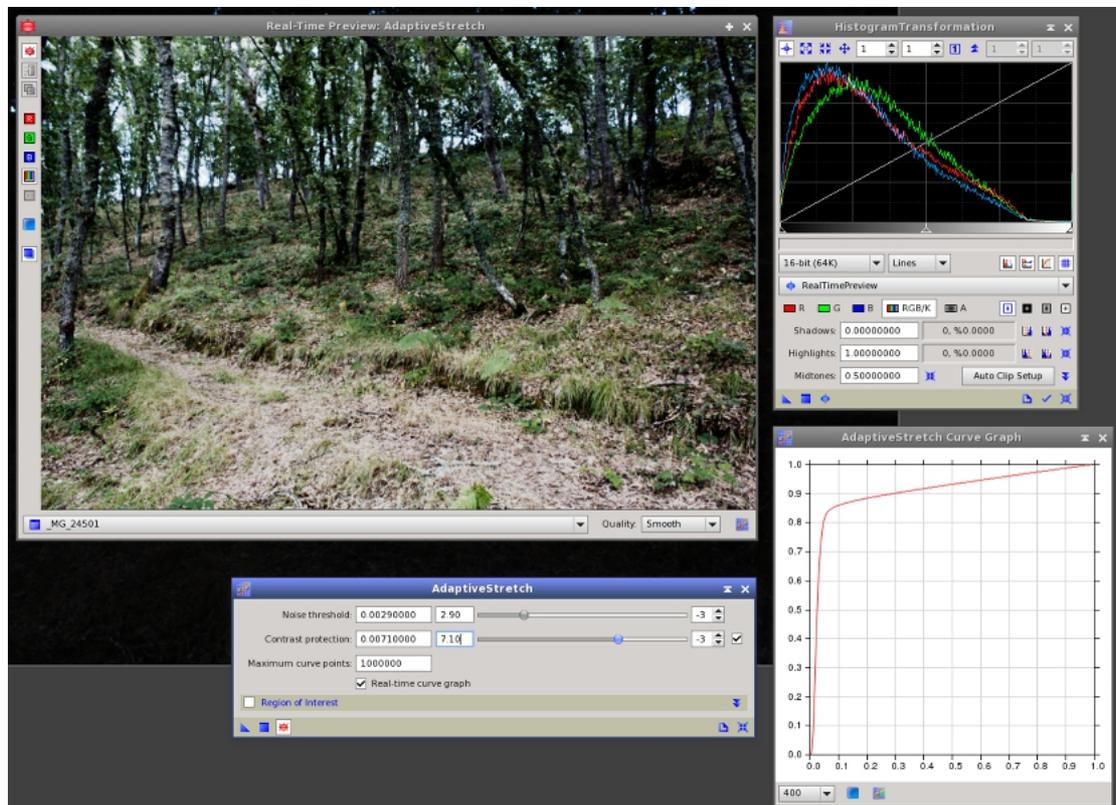
Figure 5



Travailler avec l'outil **AdaptiveStretch** en mode de prévisualisation en temps réel sur une image CCD brute

de NGC 1808. Image reproduite avec l'aimable autorisation de Jack Harvey / SSRO.

**Figure 6**



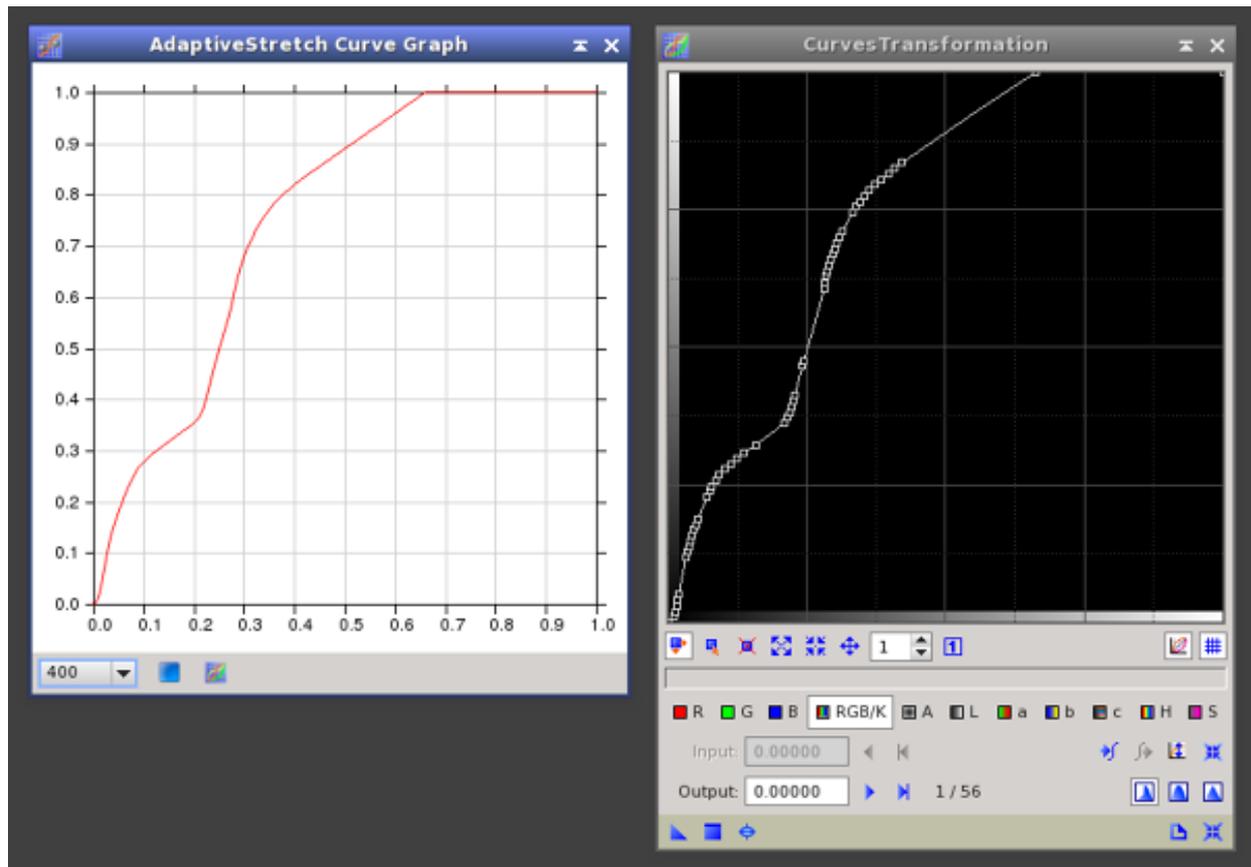
La vue virtuelle **RealTimePreview** peut être utilisée pour inspecter l'histogramme de l'image transformée sur l'outil **HistogramTransformation** au cours d'une session en temps réel **AdaptiveStretch**. L'histogramme et le graphique de la courbe vous fournissent une foule d'informations pour contrôler l'image finale.

### 3.2.1 Graphiques en courbes

L'une des caractéristiques les plus utiles de l'**AdaptiveStretch** est sa capacité à générer un graphique de courbe en temps réel. Ce graphique représente la courbe de transformation en cours de calcul en fonction des paramètres actuels du processus. Le graphique est dessiné sur une fenêtre auxiliaire lorsque l'outil fonctionne en mode de prévisualisation en temps réel, si la case à cocher du graphique de courbe en temps réel est cochée.

Le graphique de courbe vous permet d'évaluer et de comprendre pleinement la transformation appliquée, tandis que vous pouvez voir le résultat sur la fenêtre de prévisualisation en temps réel, et même son histogramme sur l'outil **HistogramTransformation** avec la vue virtuelle **RealTimePreview** sélectionnée (un exemple de cela est montré sur la figure ci-dessus). Deux boutons de la fenêtre **AdaptiveStretch Curve Graph** vous permettent de rendre le graphique sous la forme d'une nouvelle fenêtre d'image (sous la forme d'une image couleur RGB 8 bits) et de convertir la courbe **AdaptiveStretch** actuelle en une instance de **CurvesTransformation**. Cela est utile pour appliquer la même courbe de transformation à un ensemble d'images. Comme le processus **AdaptiveStretch** dépend du contenu de l'image traitée, chaque instance en cours d'exécution applique une transformation différente à chaque image particulière ; cependant, **CurvesTransformation** n'est pas un processus adaptatif et n'a donc pas cette limitation.

Figure 7



Une courbe **AdaptiveStretch** et son équivalent **CurvesTransformation**.

## References

[1] Maria Petrou and Costas Petrou, *Image Processing: The Fundamentals, Second Edition*, John Wiley and Sons Ltd., 2010, pp. 377–382.

## Related Tools

[HistogramTransformation](#), [CurvesTransformation](#), [AutoHistogram](#), [LocalHistogramEqualization](#)