

## 游戏猫的天文摄影手记：天文摄影中的频率

先扯一段没用的：到底要从什么样的视角去观察一张图片呢？

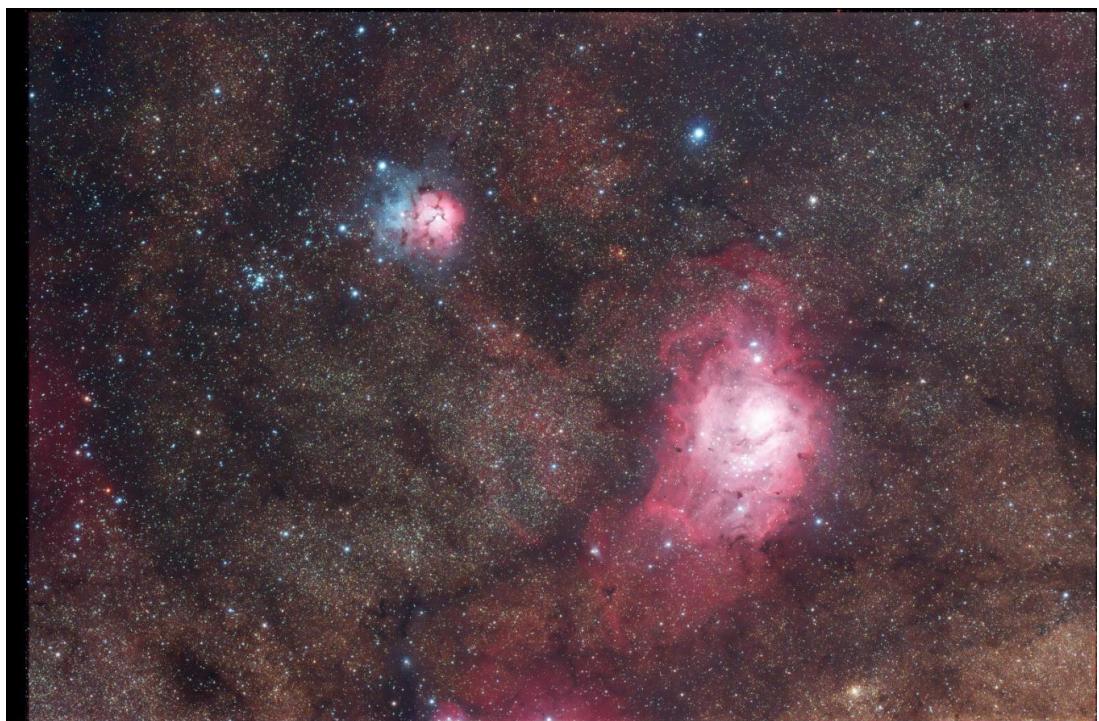
玩人像摄影的喜欢看细节，无忌蜂鸟上人称“100%数毛党”，非得 100% 看图，然后数数这模特脸上有几根猫毛。

玩天文摄影的看啥呢？

实际上怎么看一张图片都是对的，唯独不能“分块”

而图像的处理，怎么做都不为过，唯独不能“画画”

我举一个栗子，一个大家可能都用惯了的调图的办法：



这张 M20-M8 的照片啊，(素材来源：吴振，这一张是我做完了的成品图)，你看是不是右下角亮左上角暗。

然后怎么办怎么办怎么办？这难道不是传说中人见人爱花见花开的 gradient 吗？

所以你不知道从啥时候就学会了一种办法。photoshop 的终极大招啊！选区，羽化 然后提亮，百试不爽！



以后只要是画面的局部有什么亮度异常，色彩异常，或者什么别的东西，全用这招！

然后恭喜你，你离绘画派更近了一步！因为本身已经是银河里的正常结构亮度变化，非得来这么一笔，简直画蛇添足；记得闲聊的时候冬时说过这么一句话

天文摄影我到底要表现啥？1 还是 2？

1 “宇宙很牛逼”

2 “我很牛逼”

你的心态是 1 还是 2，最后能决定你的照片拍出来是什么水准。

---

闲话就扯到这里，上面那个例子我就拿来说一个问题：是摄影，是作画，还是涂鸦，全在于你对图像的理解到了何种程度

先来看啥叫频率。

如果大学里学过的傅立叶分析还没忘的话

你一定记得这样一句话：**任何一个信号，都可以看作无数个频率不一样的正余弦函数的线性叠加。**

一张照片也是一个信号，是亮度在一个二维平面上的分布式。所以我们也可以把图片做类似的分解，分成一级一级的“信号”，这些信号叠加起来就是整张图片了。

为了便于大家理解，我暂时先把一张图片拆成这样四个尺度：

1 **噪声** 没错，读出噪声光噪声全都在里面，噪声起伏的尺度是一个像素(彩色相机会稍微大点)

2 **星点** 星点的尺度通常在 2-4 个像素上，大于 4 的话我觉得你那是跑焦

3 **细节** 照片上比星点再大一点的元素，就是大家所说的细节

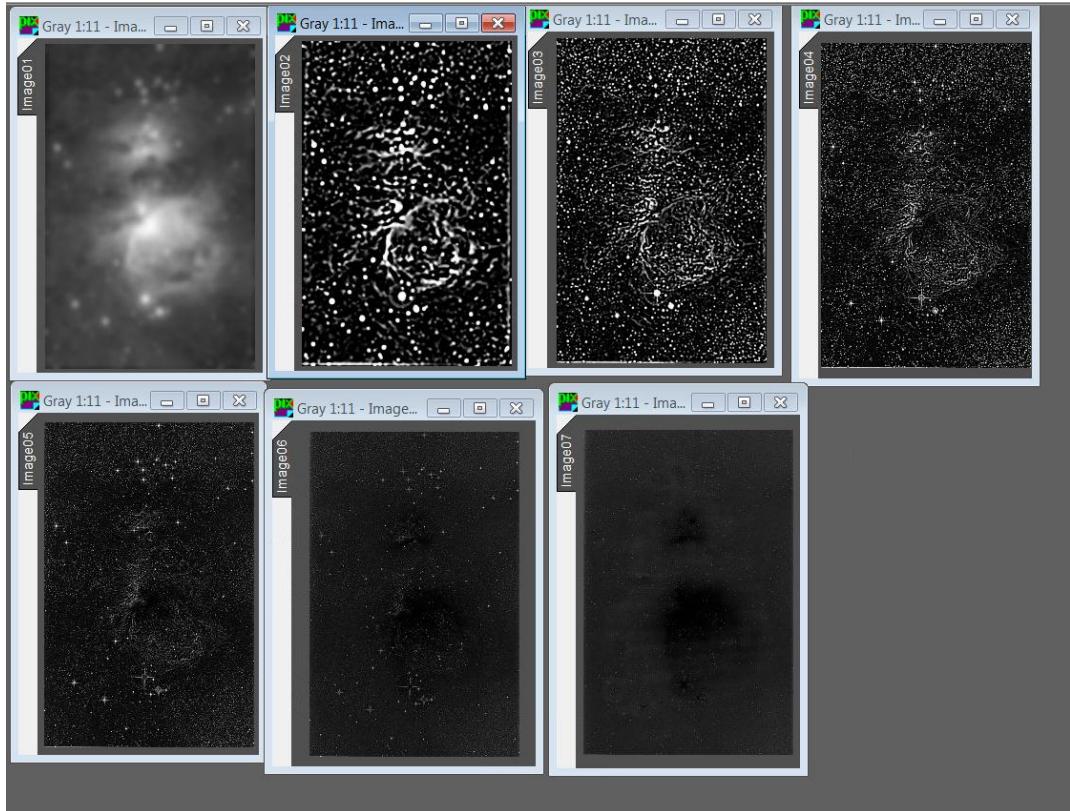
4 **轮廓** 比如说刚才我举的那个栗子里，M8 和 M20 有俩轮廓，一个像骷髅头一个像三叶草，如果你是个近视眼的话，你就看不到细节只能看到轮廓，所以近视眼是个天生的**低通滤镜**，让低频通过高频阻止。

5 **背景** 这是整张图片里最低频的信号，通常跨越成百上千个像素，变化十分平缓

没图你说个歼 8

按照从左到右的阅读顺序，这几张图代表了从低到高的频率分布

后面六张的尺度分别是 32 像素 16 像素 8 像素 4 像素 2 像素 1 像素。第一张图是把这张 M42 减掉这六个频率之后所剩下的东东，就是所有大于 32 像素的部分全都在里面了。

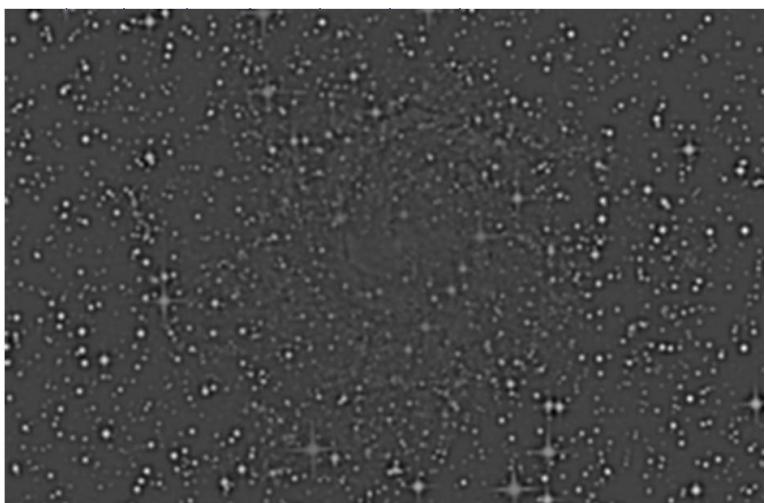
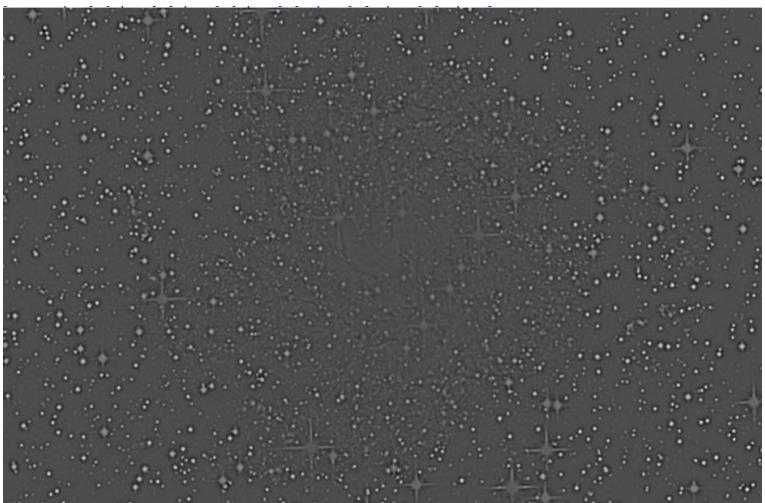
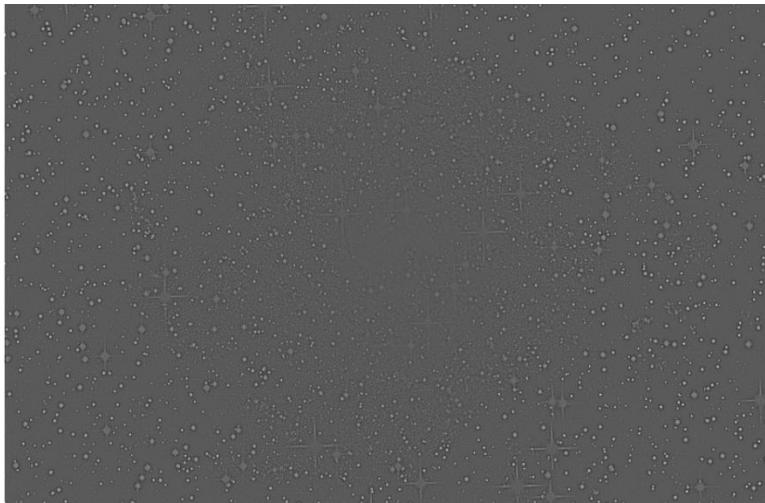


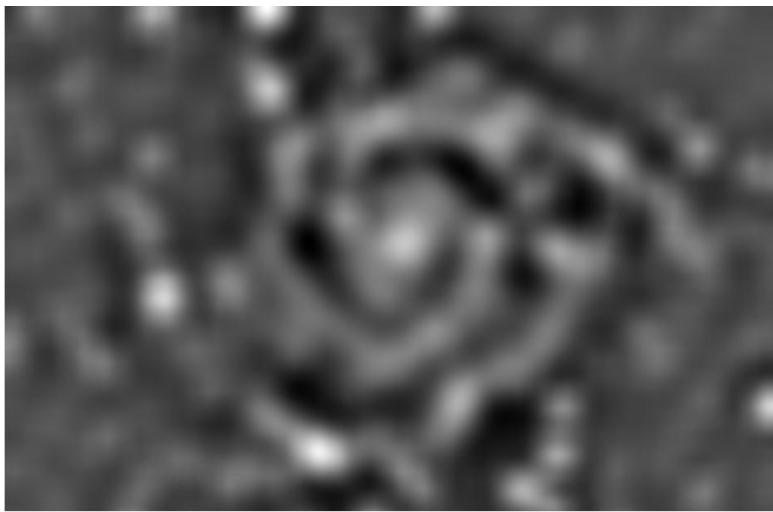
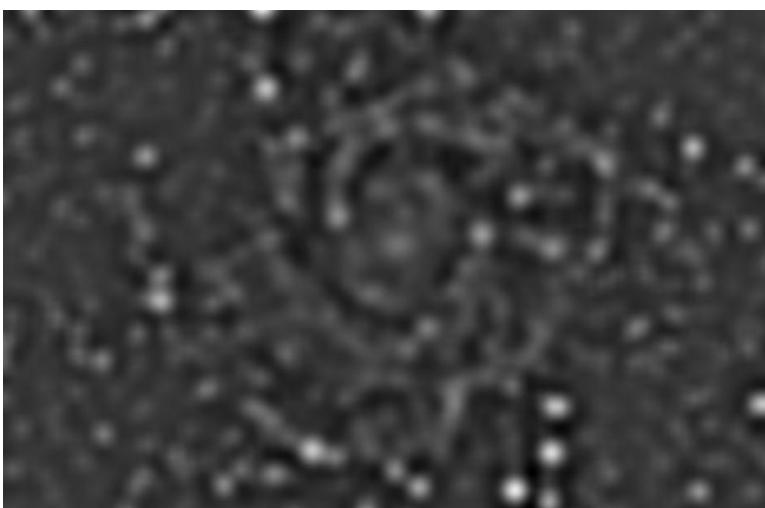
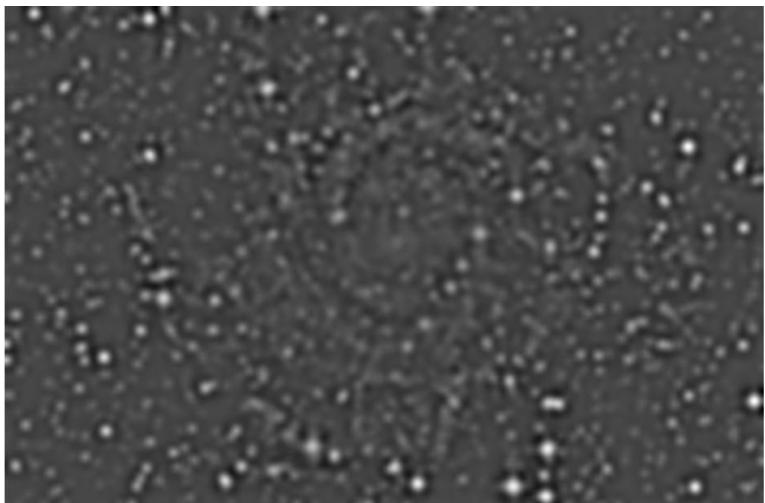
为了让大家看清楚，我把下面的那些频率都加强了一些。

看下原图长啥样吧(雷神扔给我处理的那个 M42)



这就是从频率的角度看，一张图像都是什么构成的。  
如果觉得我这个例子举得不是很恰当的话  
再来看一个另外的栗子，来源于 pixinsight LE 的说明书。





这是尺度依次从小到大拆分一张图片

原片长这样



如果加强高频率的信息，会得到：



加强低频信息就会这样



现在大家可以大致明白这频率是咋回事了吧。。

后面我会慢慢的告诉大家，读懂了频率能干啥？

- 1 处理星点
- 2 加强反差
- 3 人工平场
- 4 降噪

**不要总觉得有现成的教程就可以不懂原理**

反过来说 懂原理的人可以开发属于自己的图像处理手法

---

## 关于软件 pixinsight LE 1.0

我估计在天文摄影爱好者用的所有软件里没有比这个东西处境更加悲惨的了，喂神猫？

因为 99% 的人只会拿他来作人工平场

**而且还做不对**

前段时间土豪们挺热衷于 pix 1.8 的 游戏猫知道 1.8 很强大很可惜本猫买不起，但是**全套的操作其实都可以用 pix LE 1.0 完成的**

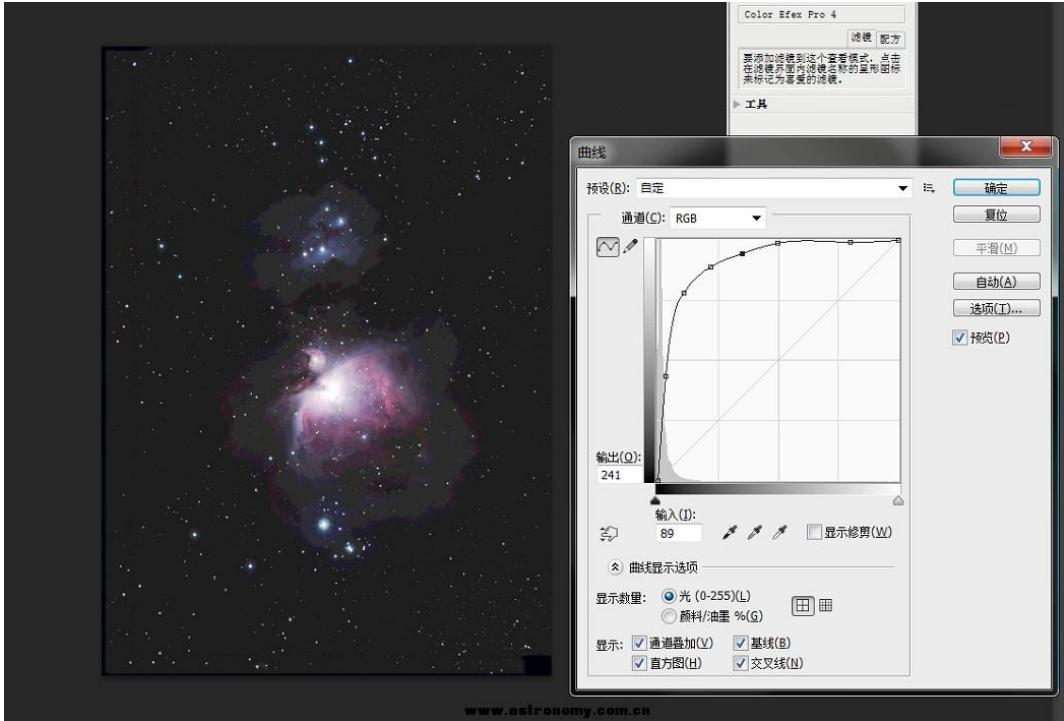
很神奇，这个软件安装包大概有 20 多 MB，但安装出来之后的程序文件只有 5MB 多一点，其他的全都是帮助文档。比起 photoshop 动辄两个多 G 的安装文件夹来说，pix 可谓是实实在在的“轻处理”了。一台用于拍摄的电脑，装一个 MDL 再装一个 PIX LE，就解决了 90% 以上的后期问题，这效率是十分可观的。

首先 pix 这个软件的工作空间可能让人感觉不适应。熟悉 photoshop 的人都知道，photoshop 的灰度空间是 8 位，但 ps 可以读取和处理 16bit 的图片。具体过程是这样的：显示器是 8bit，0-255 的灰阶。但 PS 的曲线是**每次 8 位**。所以如果你在 PS 里做一个很剧烈的曲线的话，就会出现很严重的灰阶断层。

请大家记得这个“每次 8 位”的含义。它的含义把一张 16 位(0-65535)的片子在**亮度**上分成 256 份，每一份是 256 个色阶，所以就变成了(0,255),(256,511),(512,767)....在 photoshop 里被归为 0,1,2..... 这样就让图像每次操作的精度瞬间下降。所以 photoshop 这个软件**严格来说是要被抛弃的**，但如果你的天文摄影如果不追求像 200%

看图，或者说干脆去接马赛克后缩图，那用 photoshop 或者是 pixinsight 也都无所谓了。

然而就算用 photoshop，也尽量不要养成“作画”的习惯。



如图，你在 photoshop 里做一个比较夸张的色阶，图像也是分分钟断层。

但如果在 pixinsight 里做同样剧烈的一个曲线，绝不会有断层的效果。

问：所以 pixinsight LE 都能做什么？

答：能做的事情多了去了，天文摄影中 ps 能做的它都能做，而且精度是天文级别的。曲线(包含明度，颜色，色相，饱和度) 强力拉伸工具(Histogram Transform)，蒙版(只有一种算法但极其强大)，最大值最小值中间值滤镜(有圆形孔径的选项挺方便的) 图片运算 (这个功能吊炸天，给任意两个图像做加减乘除乘方最大值最小值以及其他)，DBE(世人皆知，你第一次接触 LE 就是因为这功能吧)，ATWT(也叫 a Trous Wavelet Transforms) 小波变换工具，对频域进行分割(又一个独有的工具) RGB 图像的叠合以及拆解(别以为这东西很鸡肋，实际上用处大了) 其实这个小软件可以覆盖天文后期的绝大部分内容。LE 貌似没有自定义 kernel 滤镜是个遗憾

问：然后 pixinsight LE 不能做什么？

答：画画

问：那更高级的 PI1.8 能吗？

答：不能

问：为啥？

答：因为 pix 系的软件是不允许“绘画”的，**它不是一个商业制图软件，而是一个天文制图软件**

问：那 pix 有动作插件吗？

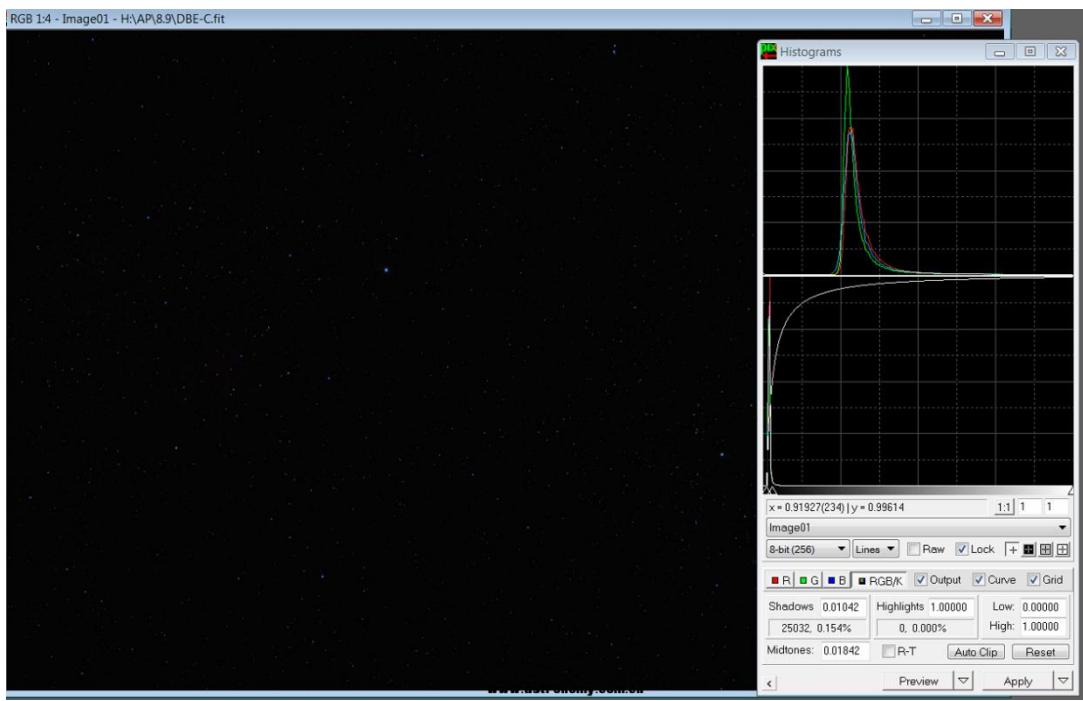
答：你想多了，但是有一个类似于动作插件的东西，叫 process Icon

**如果你喜欢在 PS 里以选区-羽化-调整为核心来作图的话，还是尽早不要碰这个 PI LE 了。。。**

先介绍个最基础的，Histogram Tranform。

其实也就是 PS 里的曲线

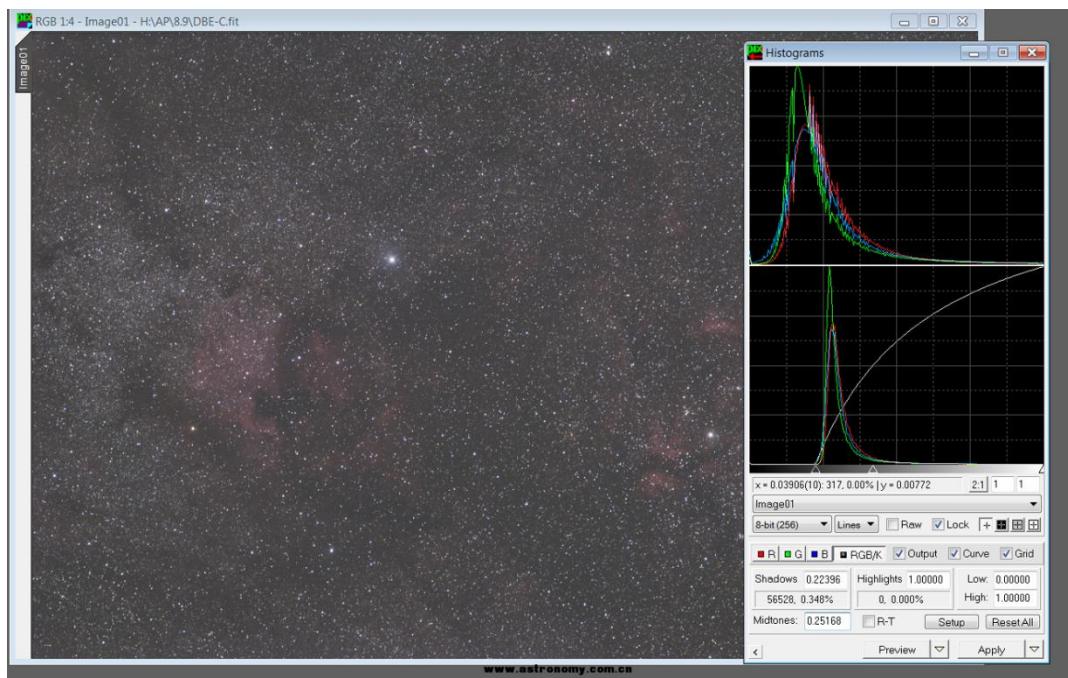
在 pix 里点图片右键然后调出这个功能来



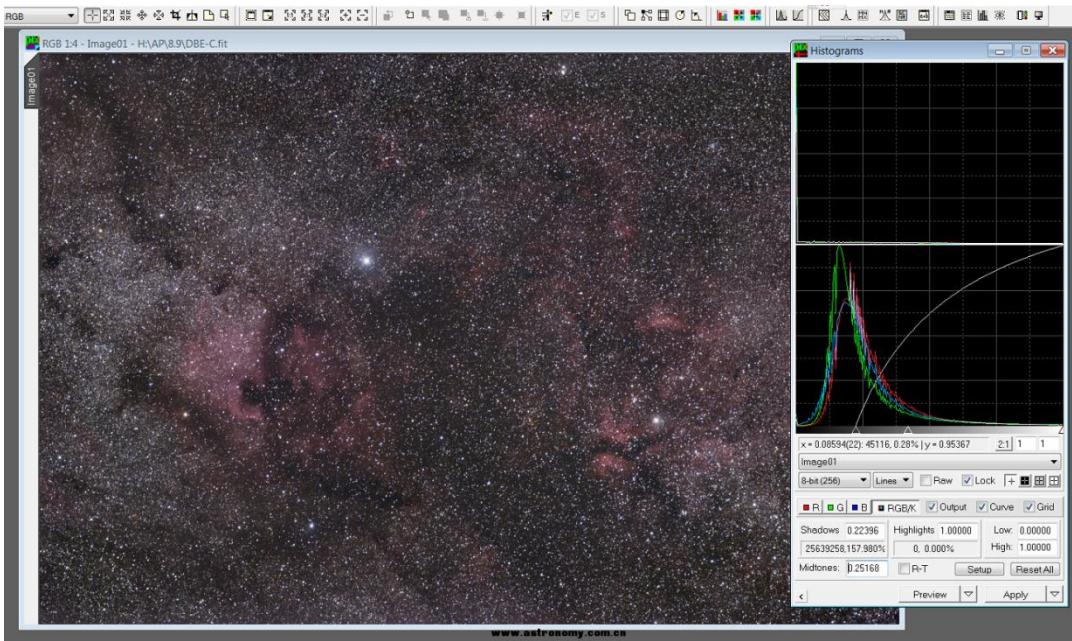
做一个如图所示的曲线

得到的效果

再来做一次力度不太大的



出图



这个过程就是大家熟悉的 DDP。

有好多人问我，猫啊猫啊你怎么不出教程呢？

嘻嘻嘻嘻 游戏猫就是不出教程，气死你气死你气死你，喵。

实际上不出教程是因为各个目标都有自己的处理倾向，不能一概而论。如果我只是把操作步骤说出来了而不说一些基本原理，那对大家的处理没有任何好处。

所以游戏猫的帖子都是先介绍原理然后介绍工具，**至于操作手法我是不给的**，懂了前两者，具体操作自己可以尝试。

ok，继续来看 pix le 这个软件。这个软件里有一个非常强大的功能叫做 mask。它的算法在 pix le 的帮助文件里有讲：[类 Amount 算法](#)，不过如果你不想去明白它的话也无所谓。记住这样几句话：

- 1 蒙版顾名思义就是把图像的一部分蒙上，另一部分露出来。
- 2 任何图像都能作为另一张的蒙版。
- 3 图像 a 盖上一个蒙版 b，如果 b 是全黑的，那么对 a 的任何操作，都是没用的。
- 4 图像 a 盖上一个蒙版 b，如果 b 是全白的，那么对 a 的任何操作都跟没有蒙版的情况下是一样的。
- 5 如果 b 是介于黑白之间的一张图，那么对带着蒙版的 a 的操作强度，要比 b 全白的时候弱，但不是全无/ 实际上 **photoshop** 里的羽化功能就是这么一回事

蒙版的原理就是这样，蒙版的亮部相当于给图像开了个小窗，只允许对小窗里能看到的东西进行操作。

明白了“蒙版”的含义之后，来看操作的实例：

## 高频下的游戏：starmask 与缩星

请记住一个观点(对你的图像处理非常有用)：**星点的大小是不能缩的！**

这个结论到最后我再告诉大家。但无论如何先记住这样的话：大家所谓的“缩星”实际上是对星点的“压制”，是让密密麻麻的星点变暗，而不是从本质上说的“变细”。让星点变细的方法是 **deconvolution(反卷积)**，那个是彻彻底底的能让星点变细又能提高解析力。下面我们介绍的缩星术，只是一种**让星云看起来更加突出的手段**。

**第一步：制作星点蒙版。**

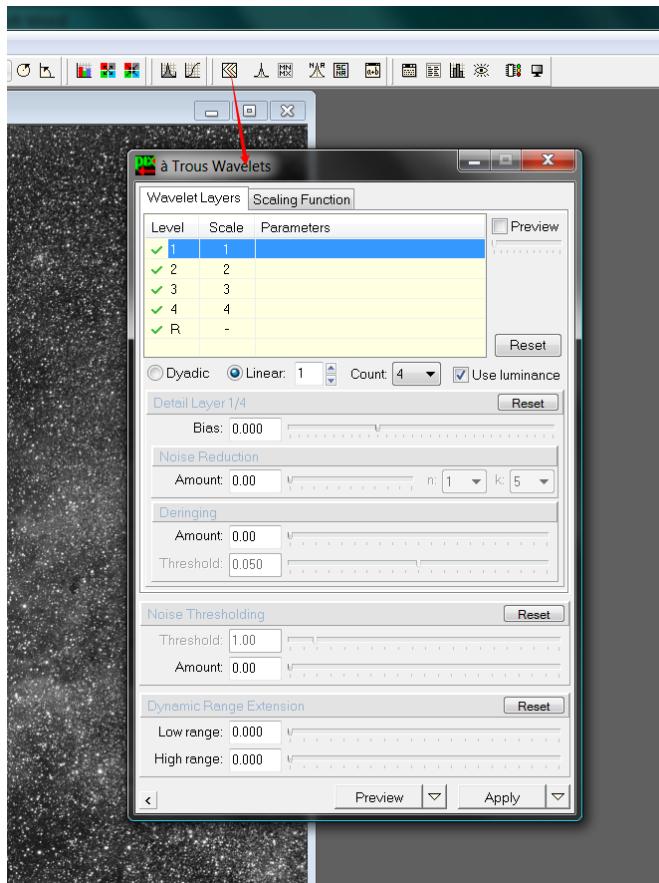
缩星之前你得先选星，如果不选中的话那简直就是给图像自掘坟墓。

那么星点在图像里有什么样的特征来着？

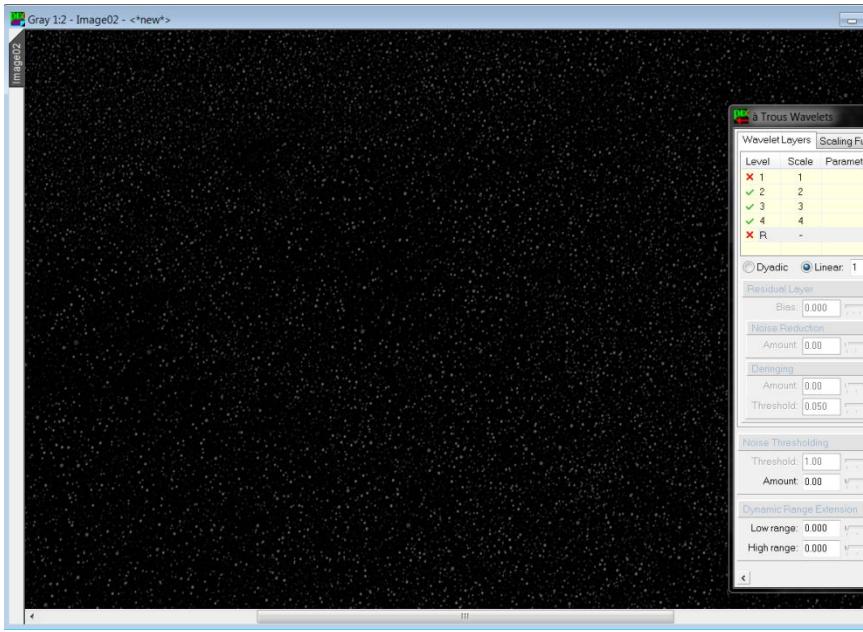
## 高频

所以我们在 pixinsight 里调出小波变换工具。A Trous Wavelet Transform

图像用的是 2015 年 8 月 9 日拍的一张北美洲广域。机器是没有改机的尼康 D7100，镜头是尼康的小痰盂，用艾顿星野赤道仪跟踪。



对于我们经常拍的广域片子来说，星点一般会占据 2-4 像素这几个尺度，所以我们把 1 和 R 的频率删掉，只留下 2 3 4.

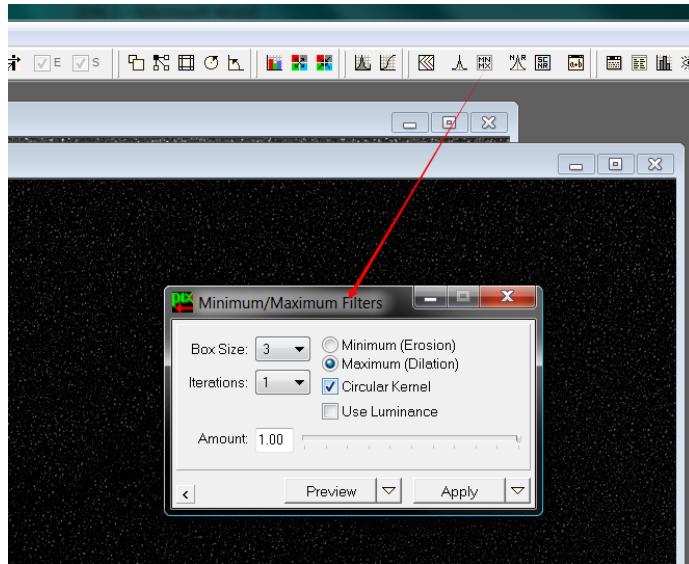


这样就直接剩下星点层了。

接下来我们面临一个问题：怎么把星点层当作蒙版使用？

我们企图制造这样一个效果：**让星点的中心达到全白，让星点以外的区域为全黑**。同时星点的遮罩要完全遮住每一颗星，如果遮不住，后果是非常严重的！

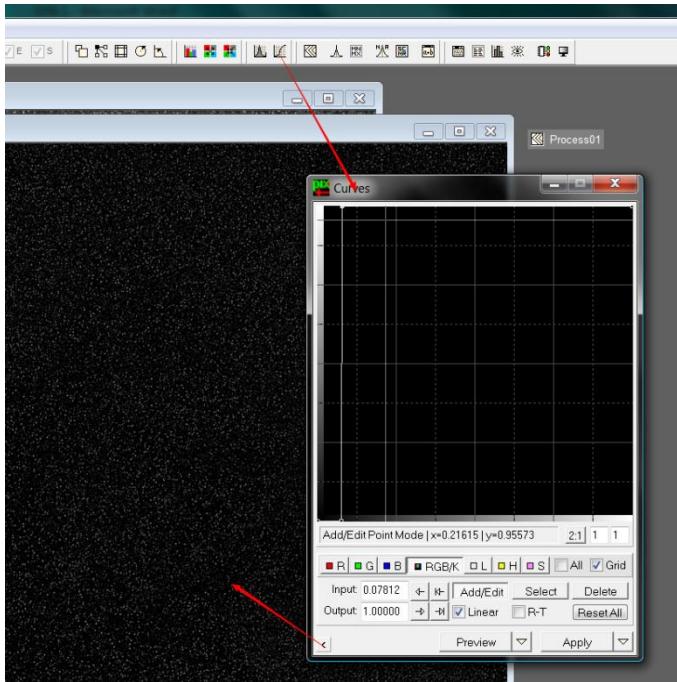
为了达到这个目的，用 pixinsight 里的最大值滤镜把星点扩展一下，以便留出一些余量来。



但是做完最大值滤镜之后的星点还不能用，因为星点的中心还没有达到最亮，这时候把它当作原图的蒙版，对原图星点部分的操作强度不会很高，也就达不到压制星点的效果。

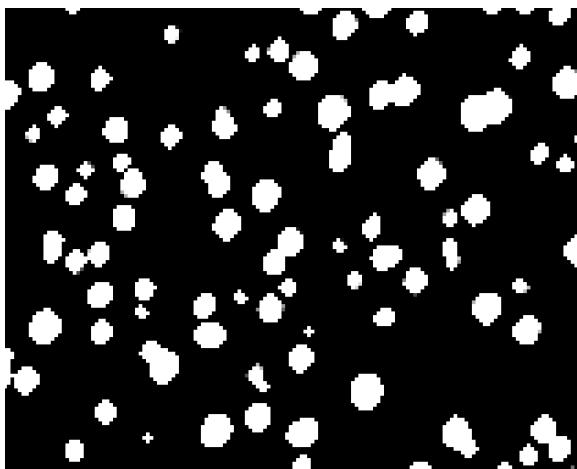
所以现在迎来了最重要的一步：**二值化**。

我们先来解释下什么叫**二值化**。选定一个亮度，让低于这个亮度的像素全部为 0，高于这个亮度的像素全部为 1.

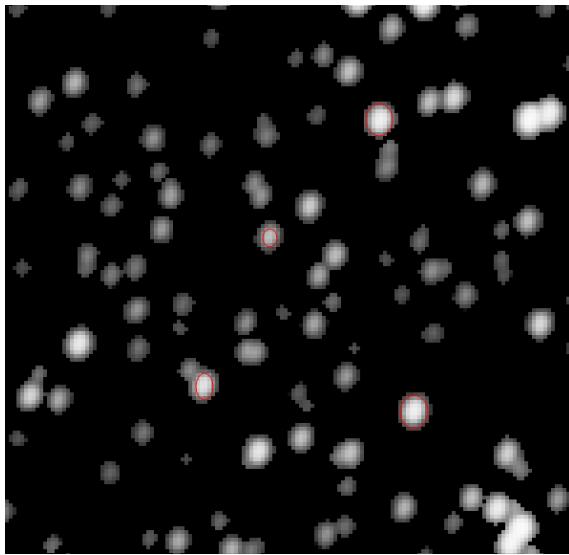


这一步的工具实现，我们选择 pixinsight 的曲线功能。

做完了之后就会有种让人生不如死的感觉：密恐

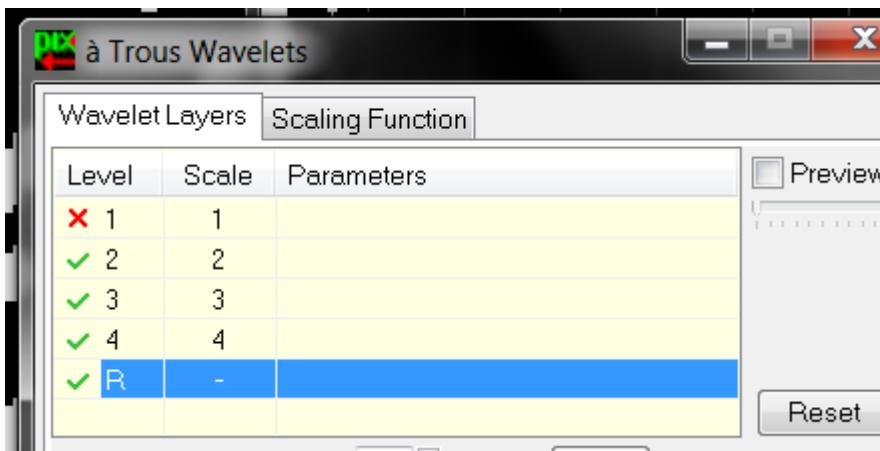


现在我们如果把蒙版应用到原图上去，就会有：



注意看我画红圈的位置，就是原图中的星点，这个蒙版的作用就是只允许星点和星点附近的“天窗”打开，进而以后对它进行的操作全都是在天窗里的，天窗外部的黑色区域由于蒙版为 0，所以不进行任何操作。

下面的操作主要是用最小值滤镜对这些天窗内部的星点进行压制。但是这样做又会出现另外的问题，由于蒙版的边缘是非常锐利的，难免有些带光晕的恒星周围有一圈被压黑了，非常难看。为了解决这个问题，我们来把星点蒙版做一下模糊，让星点和背景之间有个比较良好的过渡

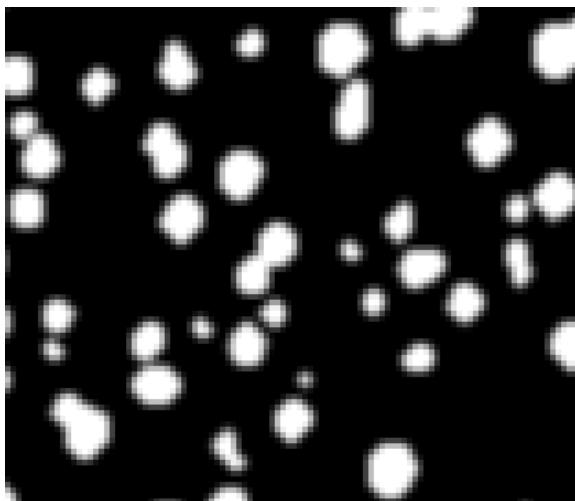


再次调出 A trous Wavelet 工具(简称 ATWT)，干掉第一层。

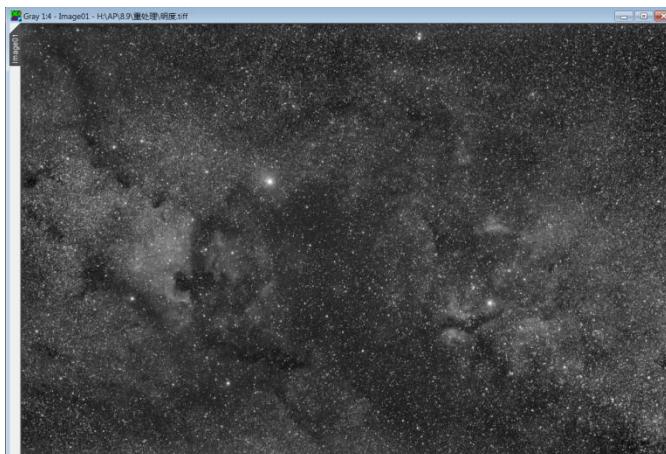
有人会问，做星点的时候不是已经干掉了第一层吗？为什么还要重新干掉第一层

对的！你别忘了，做完了星点图层我们还经历了最大值和二值化两个步骤，而二值化这个步骤瞬间让边缘变得极端锐利。傅立叶分析的知识告诉我们：任何一个锐利的边缘或者是 8 型函数，都会包含有大量的高低频率。这里我们把最高的一个频率除掉，这个边缘的极端锐利的特征就消失，如果再砍掉第二个频率，那么星点层就更加模糊.....以此类推。实际上也可以在 MDL 里做一个半径为 1.0 像素或者 1.5 像素的 Gaussian Blur 滤镜，高斯模糊也有阻断高频信号的特性

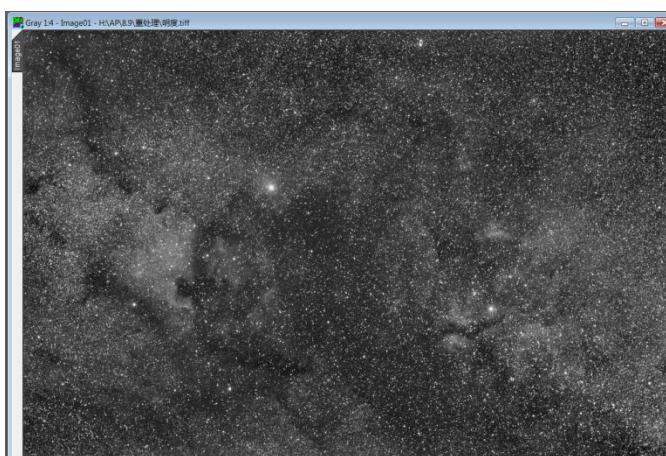
于是用这样的变换作用在星点蒙版上，得到的蒙版就像这样比较柔和了



这时候我们再对原图做一次最小值滤镜，就可以放心的把恒星压的很暗很暗，星云也就突出了  
缩星后



缩星前



这个星点图层的制作方式，将作为一个“基本功”，以后的处理文档里不会再提。

星点蒙版除了能缩星以外，还有更加神通广大的用途：[遮星拉伸\(MaskedStretch\)](#)

如果你真的学会了星点图层的制作和 mask 的基本思路，那我觉得这部分是根本不用讲的！

缩星的时候你干什么？[遮住星云压制星点](#)

所以 MaskedStretch 的时候我们要做什么？[遮住星点强调星云](#)

这部分就留作一个思考题，让读者自己来完成。[请注意调节参数，否则星点的边缘会很难看，再有就是要注](#)

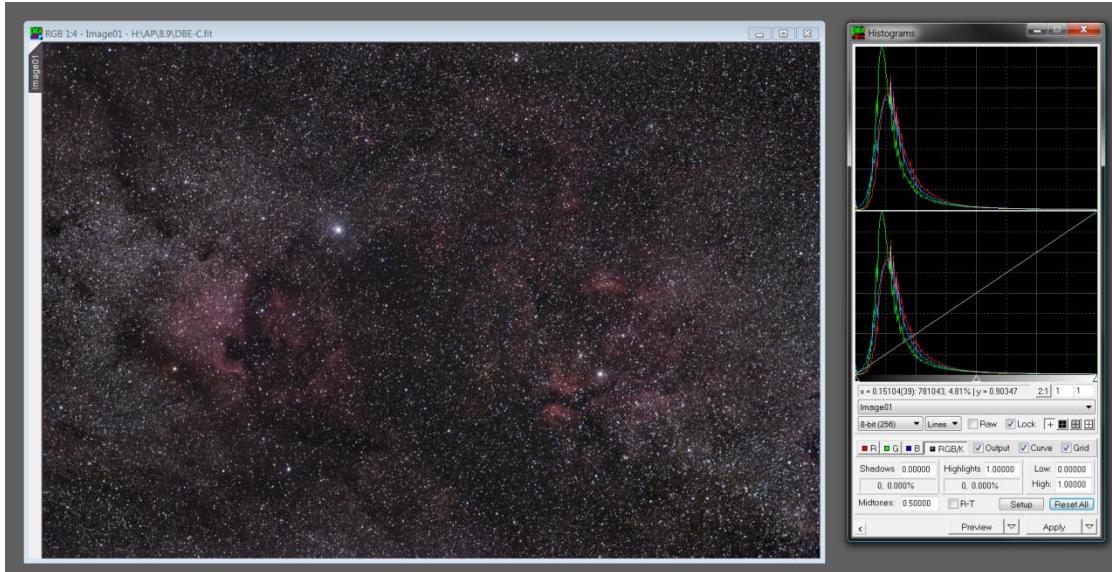
意背景的黑度和星点遮盖的分寸。

看到这里我觉得每一个爱思考的同学都会明白蒙版是干什么的，一句话，用来做给定频率/亮度范围的遮罩。由你来决定遮住哪部分，开放哪部分。但有一个非常要緊的大前提：你遮住的和你开放的部分，禁止带有人工的痕迹。(实际上 pixinsight 也不会给你画画的机会 233)

那么以上两种操作在 **photoshop** 里能不能做呢，答案是依然可以，但是 **photoshop** 的做法多多少少有些粗糙，尤其是望远镜带有一些星芒时，更是如此。如果说追求最好的效果 我推荐用 **pixinsight LE**。其实无论是 PI 还是 PS 都能做。

## 低频下的游戏：shadowmask 与饱和度提升，HDR

天文摄影中有个比较让人头疼的事情：如何增加饱和度。我们来体会一下来自饱和度的嘲讽：



还是刚才那张天鹅座广域。对原片进行 Histogram Transform(其实就是 DDP 不同软件里的叫法不一样罢了)以后得到一张图像。

局部放大，注意看暗部



不得不说彩色噪点是相当多的，而这还是在星云部分的饱和度没上来的情况下。一个没改机的单反，F4 状态下拍了 1 个多小时，各位大哥饶命啊我做不到原生的信噪比有多高。。

提高饱和度目前流行的方法有这些，(总有一款适合你)

1 photoshop，图像—调整—饱和度—然后拉高(日常摄影转过来的经常会这么玩)

2 Photoshop 里转换成 L\_ab 模式，提高 ab 两个通道的对比度(目前用的最多的方法)

3 pixLE，饱和度曲线与背景遮罩

第三个办法基本上没有人用。实际上它是一种“欺骗眼球”的办法。但是在这里，游戏滴还是要推荐一下这个做法，它的好处确实有不少。

先铺垫一点背景知识：人眼的感光特性。

我们天文学的祖先们在给星星划分亮度的时候，使用了对数坐标。1 等星比 6 等星亮 100 倍，于是有了今天的星等系统，它是个对数坐标，星等每差 1 个刻度，亮度增加 2.512 倍。

然而为什么这么划分！人眼感受上去从 1 到 6 等星的“感官亮度”可是均匀的啊！

细思极恐。

如果深入思考的话，你会得到这样一条结论：**人眼对弱光下光线变化的响应，远比强光下要敏感得多。**

实际上这个特性有效地保护了人类视觉的宽容度，听觉亦是如此。天文照片必须要做 DDP 这种强烈的拉伸操作才能适应人眼，也是这个原因。

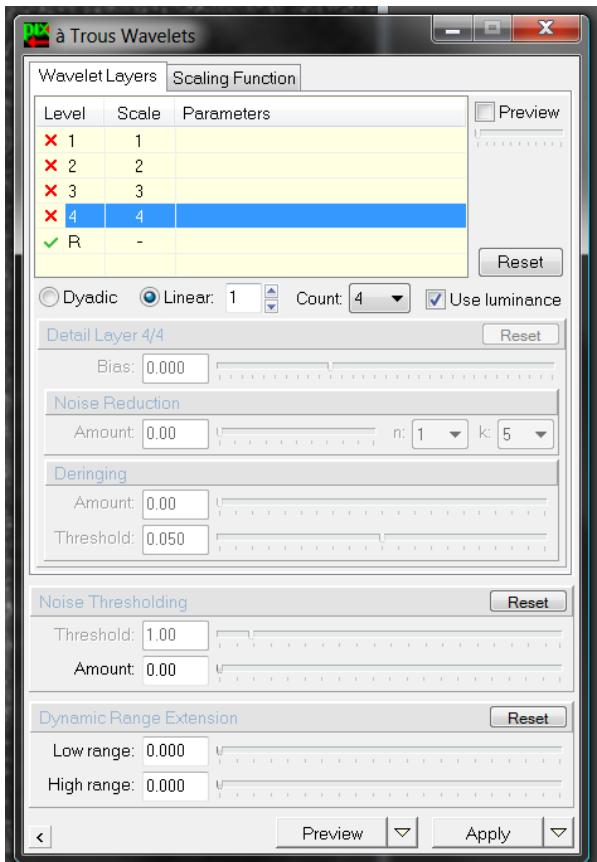
既然人眼对暗部那么敏感，那我们提高饱和度的时候，不去把每一个像素的饱和度都等幅度的提高，而是把每个像素的饱和度按它的【亮度】作为乘算因子提高，让亮的像素饱和度提高的多些，暗的像素饱和度提高的少点，这样不就完事了嘛。

所以我们当务之急是要制作一个蒙版，把暗部遮住，亮部露出来。

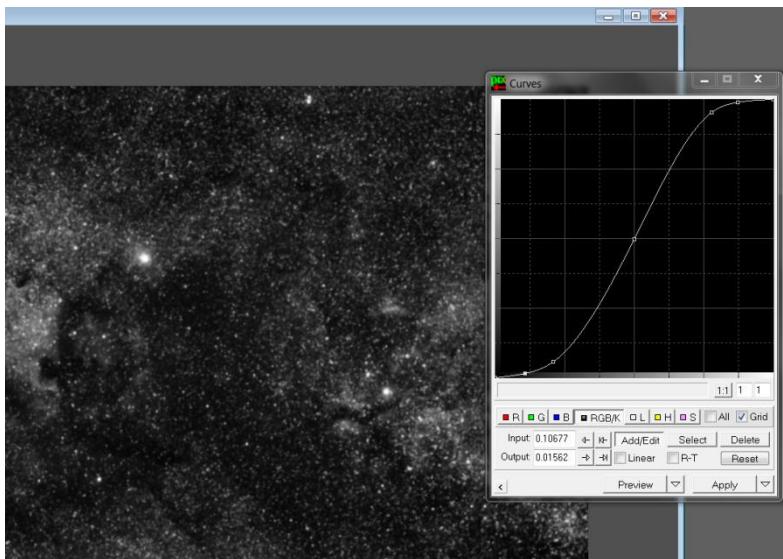


先来复制一层，然后 Image-Color spaces – convert to grayscale，把图片转成灰度

然后调出小波变换(ATWT)工具



在小波工具里 X 掉三到四个频率，4 像素以上的尺度才会被保留。这是我们打低频蒙版的基础。

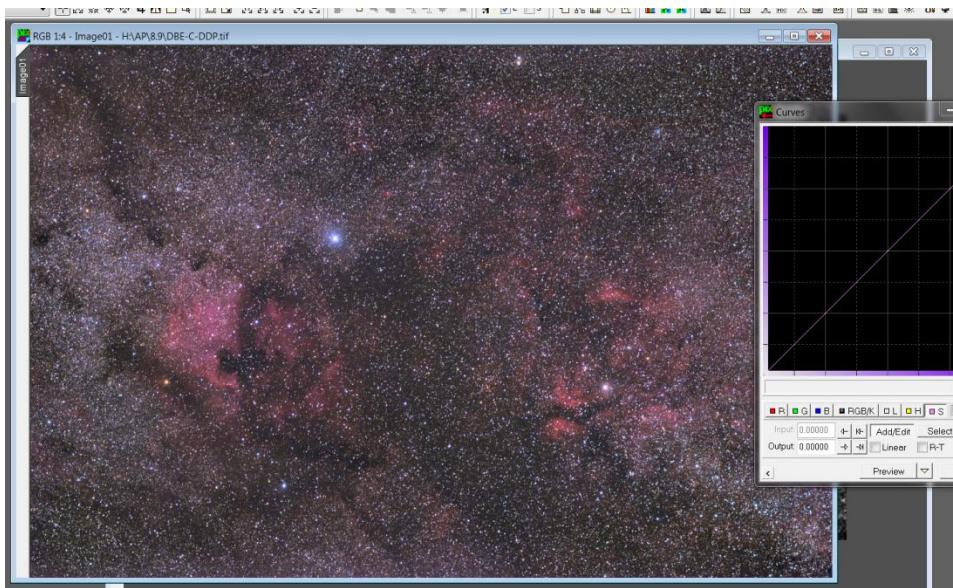


接下来要给蒙版做一个这样的强烈反差曲线，让它黑的更黑，白的更白。但要注意，黑的地方不要压死。否则图像看起来多少会有些不自然。这个尺度还得根据目标自己把握。这里的要点是，暗部的饱和度可以压制，但不能全都降到 0 搞没了，那样做是违背图像平衡这个大前提的。



然后打开曲线工具里的 **Saturation** 饱和度曲线，做一个这样的曲线，把低饱和度的区域提起来，高饱和度的区域稍微压制一下(主要是防止星点周围出现纯色)，作用到原图上。如果觉得不够的话可以作用两三次。毕竟亮部也是被蒙版遮住了一些的，饱和度曲线并没有那么立竿见影的效果。但是只要暗部的噪点被压制住，多做几次曲线，色彩总会完美的还原出来的。

这个办法适用于那些背景的灰度和色彩被校准得非常准确的片子。无论什么时候，保证背景的中性灰总是图像处理中最重要的事情。



效果如图所示，自己对着相关的步骤重复一遍，可以对比这个办法和 **photoshop** 里增加 a-b 通道对比度的办法，这个方法做出的色彩更通透，同时暗部噪点会得到不错的抑制。

适用于大多数发射或者反射星云的摄影。

当然还有其他的方法。

**LRGB** 摄影中，**RGB** 提倡 **bin1** 拍摄是什么？答案是提高噪点频率。后期提高饱和度的时候甚至可以直接 **X** 掉 **ATWT** 里的第一层，然后结合这里的低频蒙版去操作。总之天文摄影的后期处理核心是思路+工具，而不是你照着几个教程做一做就能做的多好。

如何运用 **shadowmask** 做 **HDR**，这个问题留给读者自己思考？

## 极低频的游戏：DynamicBackgroundExtraction 与人工平场

按游戏惯例，还是先讲思路，再说工具。

天文摄影不可避免地会受到光害影响。就算是在条件最优秀的观测地(波特尔二级的水平)，天光渐变的影响依然存在。至于这天光到底有多少影响，具体要看拍摄的天体有多深。天体越暗弱，视场越大，对天光渐变扣除的需求就越高。

一张广域深空天体的照片，经过拉伸之后通常是这样的：

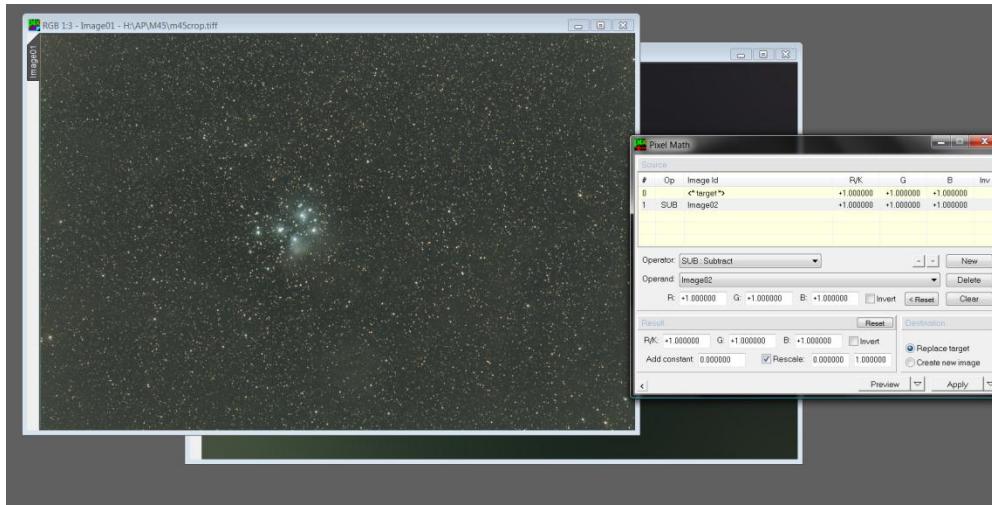


这张照片拍摄的时候已经左下角受到了观测地一盏灯的影响，导致 M45 周边的反射星云完全看不清楚。我们接下来会通过一定的手段把这个天光(姑且称作天光吧)渐变扣除。

如果读者读到这里有一定的思路的话，可能会做以下尝试：

- 1 复制一张，打开 ATWT 工具，X 掉 1,2,4,8,16,32,64,128 层，scaling function 选择 5X5 B-Spline
- 2 然后用 pixel math 给两张图做减法，“减掉”背景，得到正确的照片。

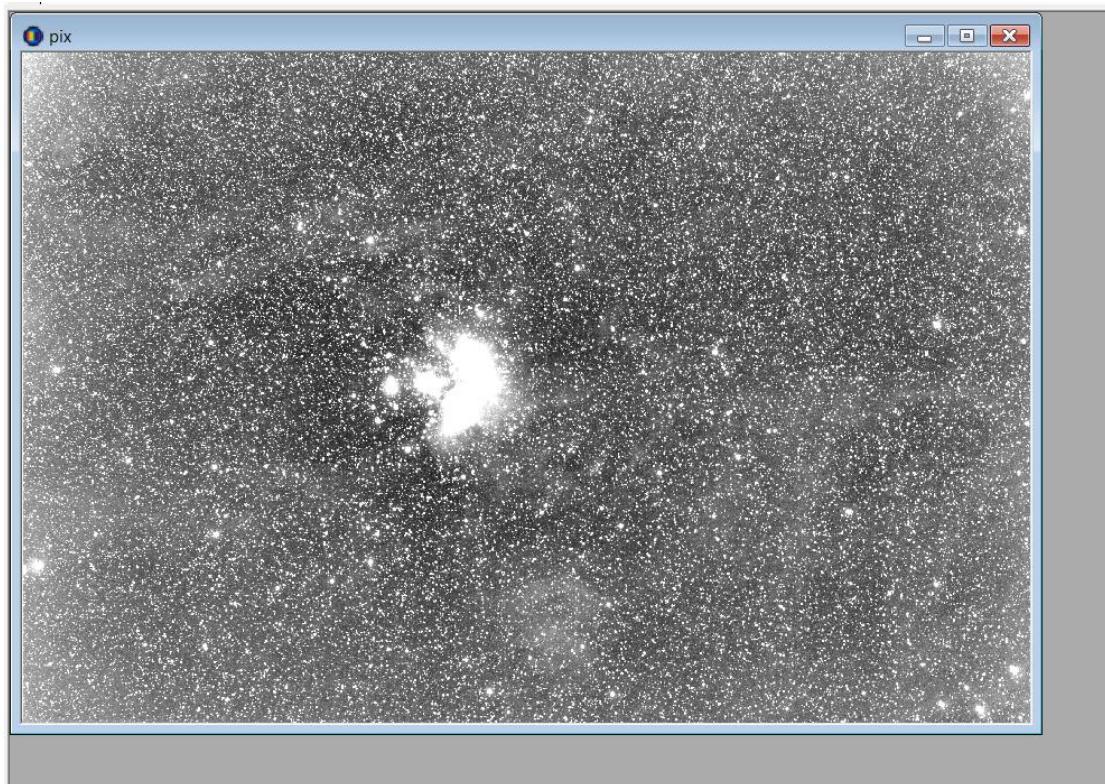
现在得到了这样的图



看起来好多了

错了！人在做天在看！直减低频留隐患！

我们把这张图再放到 MDL 下详细看看你就知道问题在哪儿了。



Screen stretch 调到最高，且不论色彩是否准确(颜色校准不是这篇文档讨论的问题)，已经能够看到这张图的严重缺陷了：四角是亮的，M45 周围却暗了一圈。为什么会这样？

我直接给答案，有两点致命的问题：

第一，极低频并不等于背景，就算在极低频率下看这张图，M45 主体那块依然是比真正的“背景”要亮，只要一相减，肯定周围一片黑乎乎。

第二，使用已经做完 ddp 的片子做背景扣除，本身就是不合理的

重要的话说三遍！

使用已经做完 ddp 的片子做背景扣除，本身就是不合理的

使用已经做完 ddp 的片子做背景扣除，本身就是不合理的

使用已经做完 ddp 的片子做背景扣除，本身就是不合理的

第一条可能看的并不是那么明显，但是第二条在这张截图上已经很清晰的展现了

道理是这样的：所谓的背景渐变，实际上分成两部分。一部分叫做四角的减光比。这是光学透镜引起的渐变，需要用减掉暗偏的亮场除以减掉暗偏的平场。还有一部分就是天光的叠加。扣除天光与天体光线的叠加，需要我们做减法，用叠加完的亮场减去天光分布。

上面所有的操作，必须在线性的，未拉伸的图片下进行，否则你的背景是不会准确的。

背景不能准确，那天体的色彩和背景必然受影响，为了弥补它，就只能到 photoshop 里面手工选区—羽化—调整.....

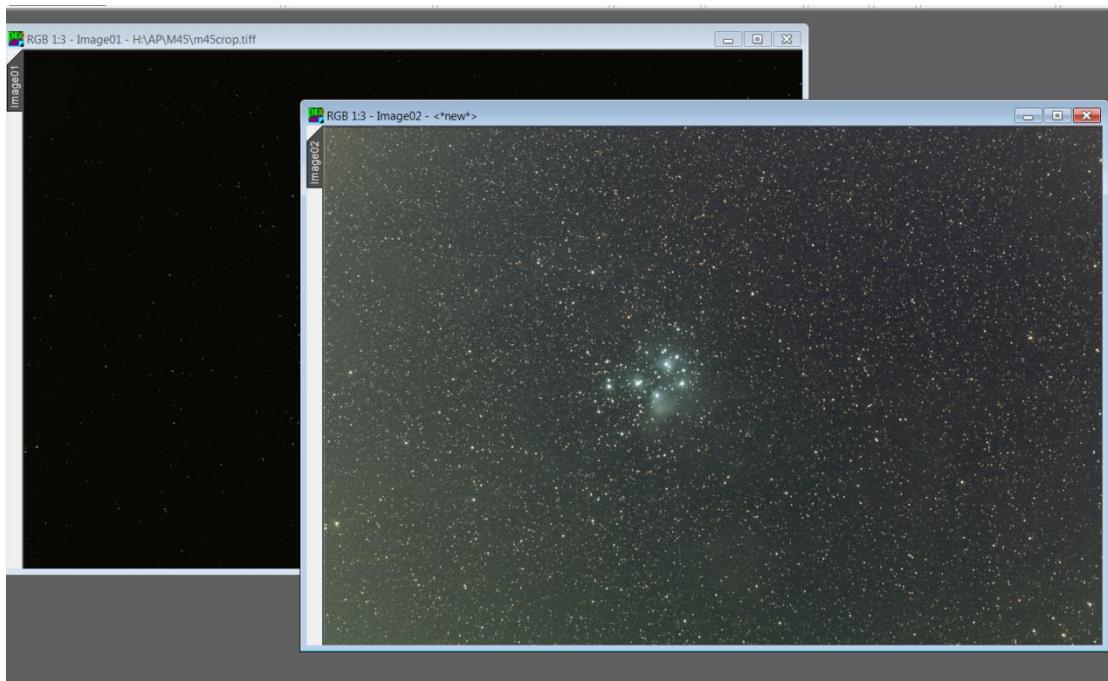
这便是游戏猫一直反对使用 GradientXTerminator, fitswork, 还有 photoshop 动作插件做人工平场的原因----软件连天光是什么样的都无法知道，还谈什么人工平场。

正确的做法如下

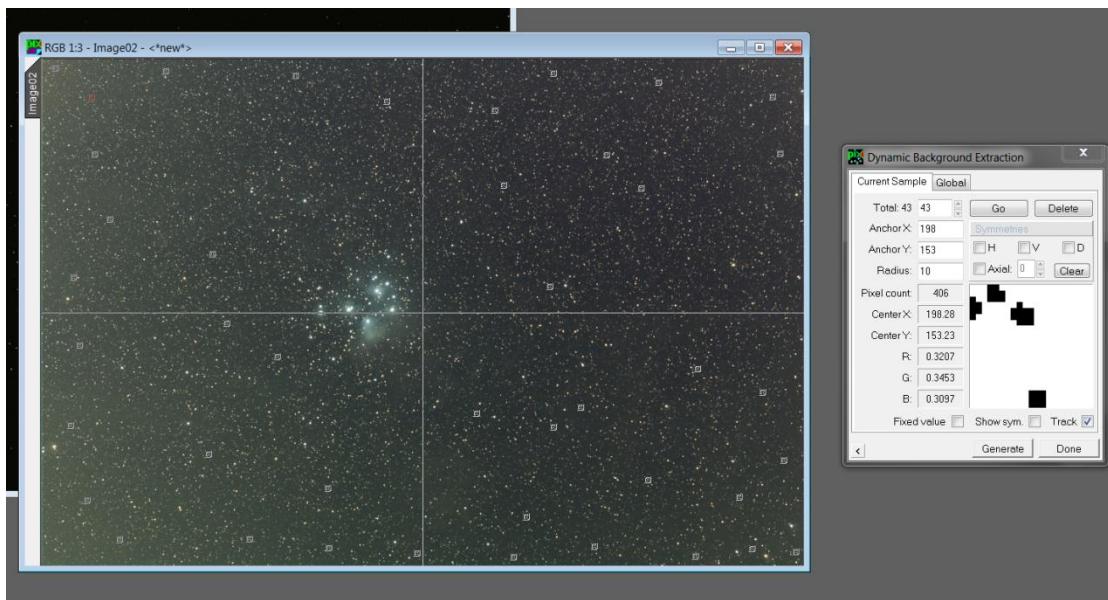
首先打开一张黑糊糊的线性照片 A

复制一张得到 B

在 B 图上先粗略对齐 RGB，然后进行强烈的拉伸



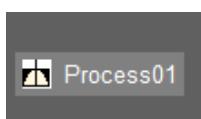
大致看清楚了天光的分布以后，我们打开 dynamic background extraction 工具



选点。注意这些点一定要选在真正的“背景”上。当然照片上很可能看不出到底哪里才是背景，这种情况下需要你去猜(真够郁闷的怎么是猜)，先猜一次，之后再慢慢的修改。

这时候我们不要点 generate

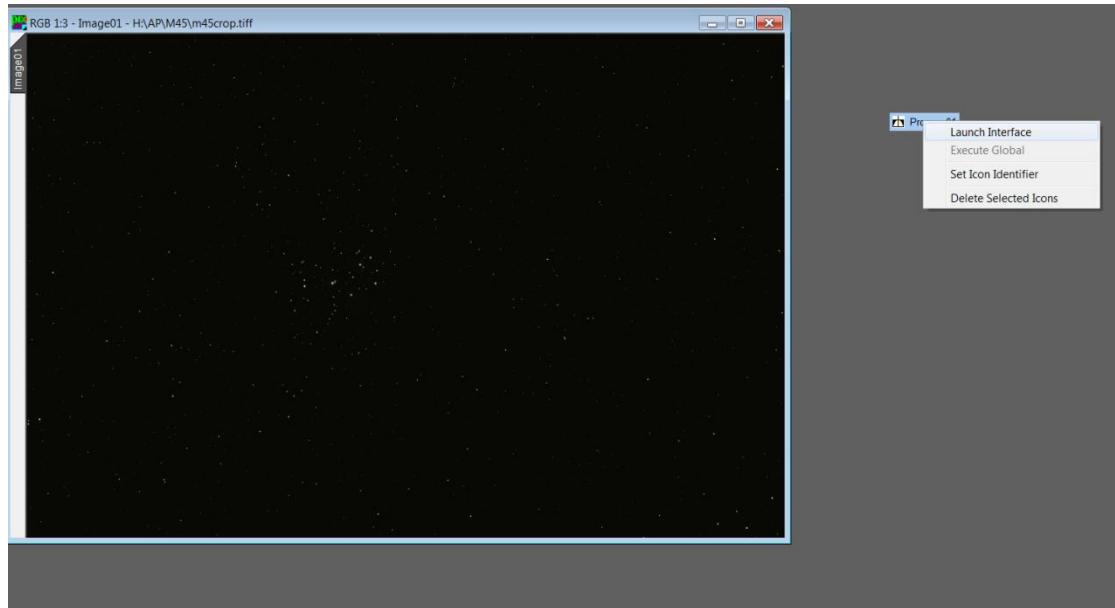
按住窗口左下角的小箭头，拖动到软件的空白部分。



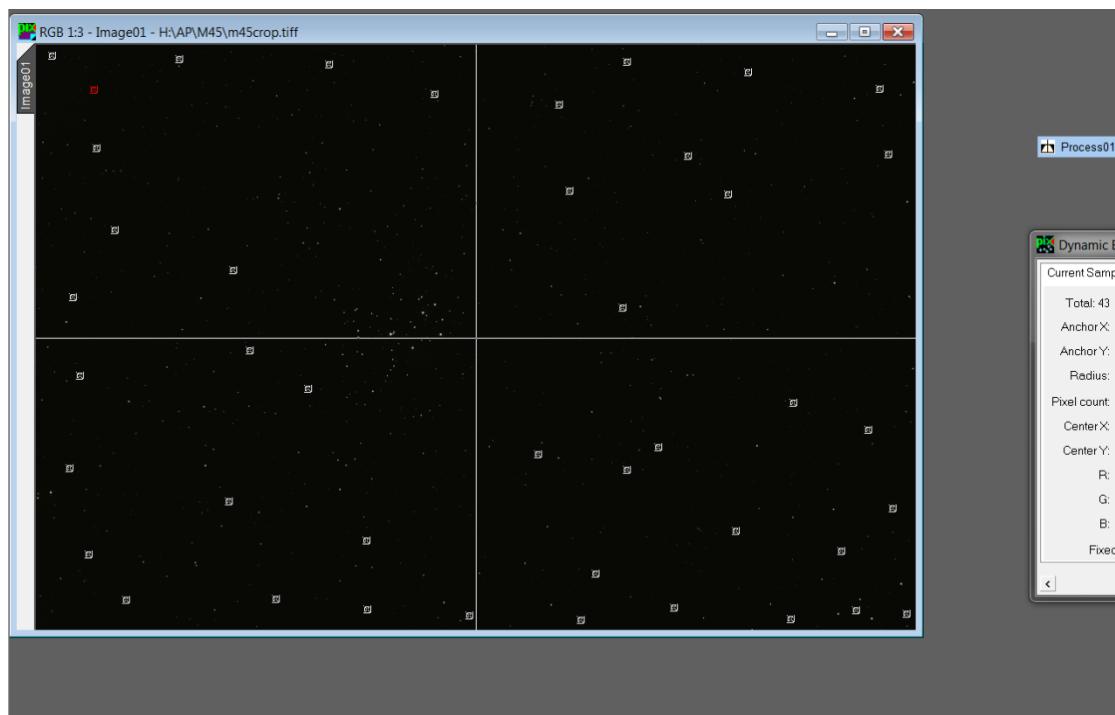
得到一个 process icon。这个东西在 pixinsight 的每一个窗口里都有，它的功能是重复窗口里的动作。

相当于 **photoshop** 里动作插件的效果。我在已经拉伸的图像 **B** 上建立了如下分布的控制点，那以后把这个 icon 拖动到图像 **A** 上，也就可以在【线性条件】下提取线性图像 **A** 的背景了。

这么做是为了避免在黑黑的图像上选点的时候误伤到星点或者云气，造成天光的判断不准。

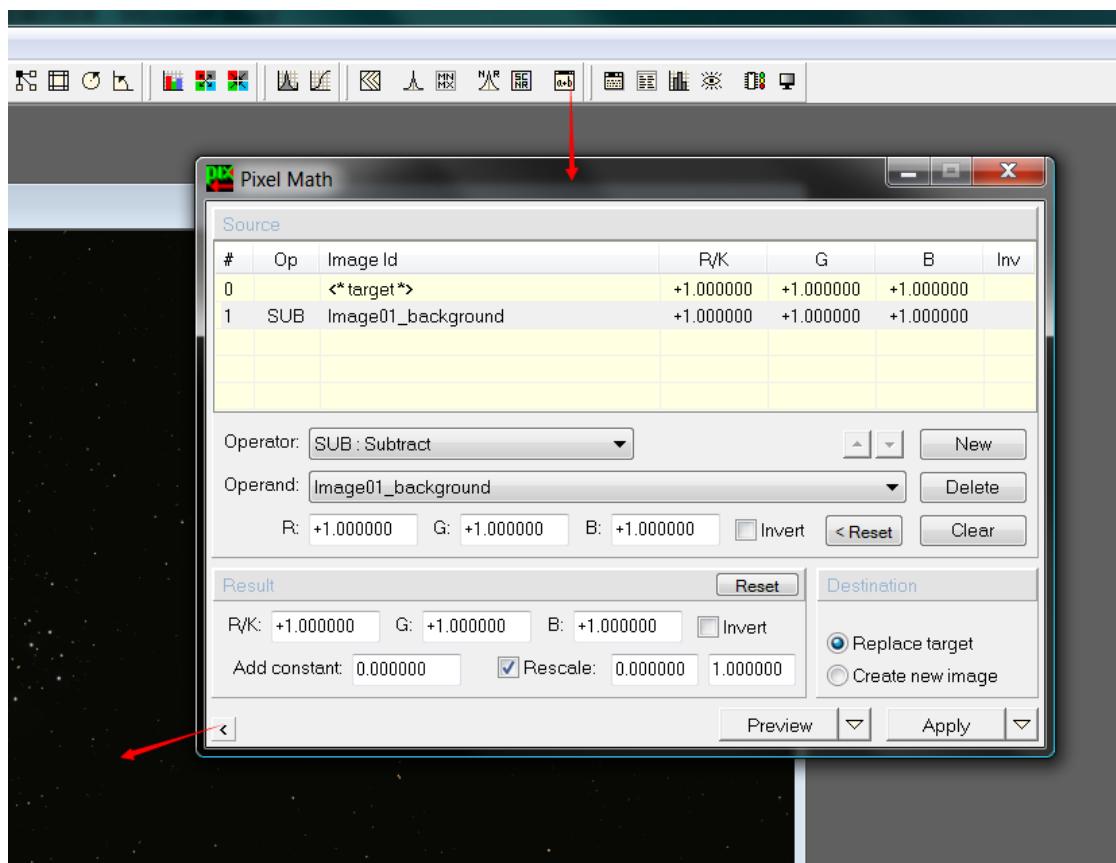


在原图上 launch interface



得到了一张一模一样的点分布图

然后点 **generate**，把背景提取出来，



最后调出 pixel math 模块，算符选择 subtract 减法，operand 选择你刚才的那张线性背景，然后拖动左下角的小箭头到原始线形图上去，完成人工平场的校正。



这次得到一个相对正确的天体信息

至此我们可以看到天体的背景不再有那样的渐变。背景平了，天体的云气自然就显示出来，需要的只是一点

点耐心去调整这些打点的位置。

最后把游戏猫的图像处理思路做一个整理，简单来说就是“**校准能做多准，拉伸就能走多远**”一步一步的扎实做好基本功，对图像处理总是有帮助的。从频率的角度去认识天文摄影的图像，本来应该是一个非常自然的观点。游戏猫在这里没有详细介绍 ATWT 工具里对于每个频率的增强和修饰手段(denoise,deringing,bias,dynamic range extension 等.)，这些东西会在后续的文档里详细说明。Pixinsight LE 这个软件，游戏猫都既不希望大家把它们奉为神明，也不希望看到有人说“它只是个做人工平场的”甚至是“不拍平场的偷懒工具”。每一个工具都有自己的优势和劣势。举一个例子，fitswork 的 GradientRemoval 模块是个非常傻瓜化的东西，只有水平和竖直两种调节模式，然而 fitswork 的猜色算法却让 pixinsight1.8 也望尘莫及。

因为

**决定效果的不是工具而是观念**