Alessio Beltrame alessio@alessiobeltrame.com www.alessiobeltrame.com

INCLUDES A TUTORIAL ABOUT DARK FRAME CALIBRATION IN PIXINSIGHT

CMOS CAMERAS

Dark current theory, measurement & image calibration

- What is dark current, how it changes with exposure time and temperature and how you can measure it.
- Why standard image calibration technique may not work well with camera showing amp glow.
- How to minimize amp glow and how take dark frames in order to prepare Master Darks.
- A complete step-by-step workflow in PixInsight to get rid of amp glow and to obtain well calibrated images

ABOUT THE AUTHOR

My name is Alessio Beltrame and I live in North-Eastern Italy where I was born in 1966. My education includes a 5-year Engineering degree (MS equivalent) with a thesis on artificial intelligence applied to electronic design, but I was always fascinated by physics and astronomy, even before attending the elementary school.

I'm also attracted by photography, with a special interest in soccer. This is one of the genres where technical knowledge of cameras and the physics behind their operation is just as important as the artistic aspects. In the last two years I finally found the time and motivation to combine photography and science in one of my childhood passions: the astronomy.

With a deep scientific/technical background, I just can't resist to look inside my cameras to find out what happens inside the box the shutter clicks.

COPYRIGHT & DISCLAIMER

Whereas I tried to be as accurate as possible in testing the camera and reporting the results, I cannot guarantee a 100% accuracy. The contents of this document are provided as-is, without any explicit or implicit guarantee. Therefore, I'm not responsible for any consequence due to statements presented, herein either direct or indirect. Whatever use you may think of the information provided by this document, use it at your own risk.

While I mention QHYCCD in this document, this publication is not sponsored or supported in whatsoever way by QHYCCD or any other vendor.

Release history:

- *Release 1.0 – December 2017: first release*

Part 0

INTRODUCTION TO THIS DOCUMENT

In this document I explore the topic of dark current measurement and dark frame calibration for a camera based on a CMOS sensor. Traditional techniques to measure the dark current were designed for CCD cameras and do not take into account the specific features of modern CMOS sensors. As a result, those methods may produce results that are unpredictable at best, if not simply wrong. The same applies to image calibration, which should be performed in a specific way in order to obtain the best final results.

This effort follows my review about the QHY163M camera that I recently purchased. While I was measuring the performance of the camera I found some inconsistencies in the results produced by the PixInsight's BasicCCDParameters script with this specific camera, regarding the magnitude of dark current. Also, while I was able to perform a good calibration of my images, I received many feedbacks from other astrophotographers having troubles with amp glow in post processing.

I then decided to investigate why the script in PixInsight produces unreliable measurements of dark current (limited to some types of CMOS camera) and to improve the method I use to measure it. I also prepared a step-by-step tutorial to show how image calibration can be performed in an effective way in order to remove amp glow.

Two words of caution before delving into the technical aspects:

- ★ While I tried to keep things as simple as possible, I also wanted to be precise and to give readers the possibility to falsify my assertions, because that's what Science demands. **Some math is inevitable**.
- ★ In addition to math, **this document is not for the layman**. If you don't know what a dark frame or a flat frame are all about, then this document is not for you.
- ★ In my examples I used the QHY163M camera by QHYCCD. That's the camera I own and know, but many other cameras with CMOS technology (including many DSLR's) work in the same way (no, I'm not talking about other cameras with the same sensor only).
- ★ **English is not my mother tongue**, please be indulgent on me.
- ★ When I wrote this document I planned to publish it also as a multi-part post on my personal website. Therefore, it is organized in two distinct sections: the first section deals with theoretical aspects of thermal noise and with methods to measure the dark current of sensor; the second section is a tutorial about how to perform dark frame calibration in PixInsight for cameras that are affected by amp glow.

Part 1

THERMAL NOISE CHARACTERIZATION AND MEASUREMENT

DARK SIGNAL, OVERSCAN CALIBRATION AND BLACK POINT CORRECTION

Thermal noise can be considered in the same way of light signal. Its net effect is to increase the number of electrons in the pixel way, in a way that is directly proportional to the exposure time. We refer to this accumulated signal as the **dark signal** and we measure it in electrons; if we divide the dark signal by the exposure time, we obtain a physical quantity having the dimensions of an electrical current (electrons/second), so we refer to it as the **dark current**.

As a bias frame is basically a dark frame with the shortest possible exposure ($t \cong 0$), we may reasonably expect the mean of a dark frame to be higher than the mean of a bias frame, but this is not always the case (Hongyun, Why Dark Field Is Darker Than Bias Field, 2017). First of all, the electrons trapped in the pixel well are counted by an Analog to Digital Converter (ADC), but to perform this task the ADC need a reference voltage.

In many cameras only the sensor is cooled, while the ADC and/or the reference voltage generator are not. In addition, bias frames are usually collected at environment temperature and in fast bursts, meaning that the ADC is continuously working, and its temperature rises. Even though the ADC and its reference voltage are relatively stable components (with a temperature drift that can be as low as some tenths of parts per million per degree centigrade), those temperature fluctuations can easily account for an increase of 30-50 ADU’s in the bias frame mean, if they are scaled over the output range of the camera (0 ÷ 65535 ADU’s).

To overcome the effects of temperature drift, it is possible to perform the **overscan calibration**. This technique consists in reading an area of the sensor that is not used for actual light gathering. This area is not affected by light or thermal noise, so it only records the bias level. To carry out the overscan calibration you need to:

1. Ensure that your driver/acquisition software reads the Overscan Area (by default this is likely to be disabled, as it presents itself visually as a black bar on one or more sides of the image)

2. Select the area of the picture corresponding to the Overscan Area of the sensor and take the mean of that area. Let’s call Bias Value the mean you just calculated.

3. Subtract the Bias Value from each and every frame you collected: bias, light, dark, flat. By the way, it is far more efficient to integrate frames first (all biases into a Master Bias, all lights into a Master Light, etc.) and then subtract the Bias Value from the Master Frame. This is true is you’re going to apply the standard image calibration formula:

   (Master Light – Master Dark) / (Master Flat – Master Bias)

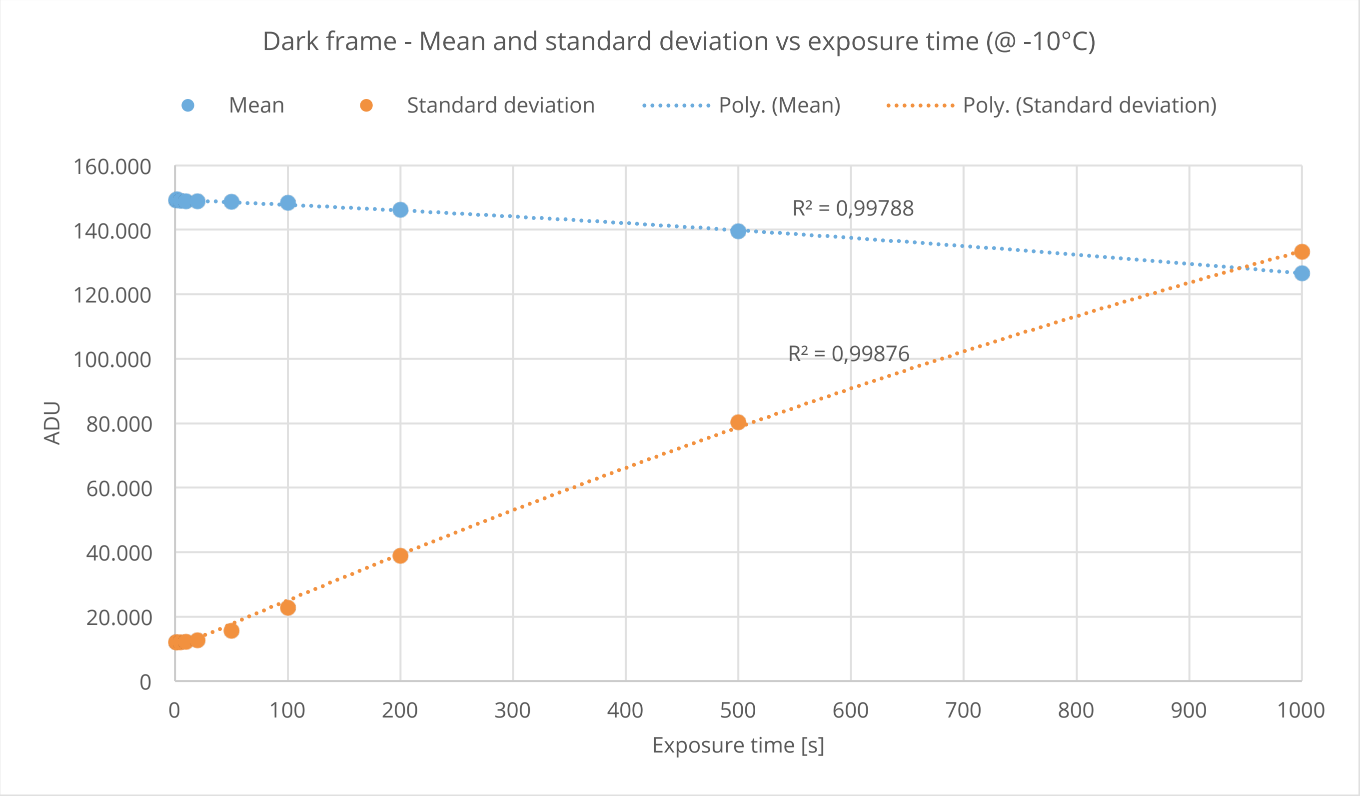
   but I will return on that formula in part 2.

In cameras using CMOS technology, there may be an additional factor to consider: Some CMOS sensors implement the **Optical Black Level Calibration**. In these sensors, a portion of the pixels is shielded from light and they form the Optical Black Area. The sensor uses the average signal in this area to set the black point for the image. It must be noted that the Optical Black area is different from the Overscan area, because it is affected by thermal noise (even though both areas do not contain any light signal). In a sense, it just like taking a dark frame in a dedicated area of the sensor and subtracting it from the signal in the remaining part of the image. Please note that this is a **constant value** subtracted from all pixels, so the standard deviation of the final image (i.e. its noise) is not affected at all. The noise is still there, but the image does not appear brighter and brighter, with growing exposure time, due to the accumulation of dark signal.

This would have no effect regarding the measurement of dark current, if there was no amp glow. The amp glow is a localized increment of temperature of the sensor in proximity of the amplification circuitry. If this increment happens to be near the Optical Black Area, then the sensor overestimates the dark signal and overcorrect the final image. As a result, the Dark Frames turn out to be darker than they need to be.

If your software does not perform Overscan Calibration, and perhaps your sensor implements the Optical Black Calibration, it may happen that the mean of a Bias Frame is higher than the mean of a Dark Frame. **This has no effect on the quality of image calibration (if you are doing it in the right way), but it may lead to errors in the measurement of dark current, is the standard procedures are used.**

The following graph shows how the mean of a dark frame changes when increasing the exposure for the QHY163M camera (Beltrame, QHY163M review, 2017):



As long as the amp glow is negligible, i.e. up to around one minute of exposure, the mean is pretty stable, then it slowly drops due to a slight black point overcorrection. Please note that the mean falls down according to a quadratic law and **this may preclude the possibility of using dark frame scaling when performing image calibration**.

NOISE SOURCES IN DIGITAL IMAGING

This topic is so vast that an entire book wouldn't be enough to discuss it in depth, see for instance (Martinec, Noise, Dynamic Range and Bit Depth in Digital SLRs), (Hornsey, Noise in Image Sensors), (Nakamura, Image sensors and signal processing for digital still cameras, 2006). In this paper I'll only list the main sources of noise in a digital image. Astro-photographers should be familiar with 3 categories of noise:

- Shot noise
- Thermal noise
- Readout noise

but in no way the above list is exhaustive. Speaking about dark current measurement, shot noise is nil as we have no light hitting the sensor. In most cases it is then assumed that the noise affecting a dark frame is limited to thermal noise and readout noise. Remembering that noise adds up in quadrature:

$$N_{TOT}^2 = N_1^2 + N_2^2 + \cdots + N_k\text{^}2$$

the above hypothesis can be written as:

$$N_d = \sigma(D) = \sqrt{N_{th}^2 + RON^2}$$

where N_d is the total noise of the dark frame D, N_{th} is the thermal noise and RON is the readout noise. However, the above formula is only valid as a first approximation and there are many cases cases where the error due to ignoring other sources of noise if far from negligible. Such sources include amp glow (and more generally other types of fixed pattern noise - FPN) and 1/f or random telegraph signal (RTS). I will ignore pixel response non-uniformity (PRNU), as it depends on light hitting the sensor, so it is not relevant for dark frames.

The above equation is then rewritten as:

$$N_d = \sigma(D) = \sqrt{N_{th}^2 + RON^2 + FPN^2 + RTS^2}$$

WHY POPULAR DARK CURRENT MEASUREMENT METHODS MAY NOT WORK

Two methods to measure dark current are very popular. PixInsight implements both in a script named BasicCCDParameters. The first method uses a single dark frame and a bias frame; the procedure consists in:

1. Subtract the bias frame from the dark frame and pretend that the resulting image contains only the dark signal;
2. Calculate the mean of the resulting image;
3. Divide the mean by the exposure time: the result is the dark current.

The above steps are summarized in the following formula:

*Dark_current = gain * (mean(Dark) – mean(Bias)) / dark_exposure_time*

There are two problems with this method. The first one lies in Hypothesis #1: after subtracting the Bias frame, the resulting image contains not only dark noise, but FPN and RTS too. For CCD cameras the contribution of FPN and RTS may be negligible, but in CMOS cameras fixed pattern noise (most notably, amp glow) can produce a little overestimation of dark current.

The second problem is much worst. We have seen before that, due to Overscan and Optical Black calibration, a Dark frame may be darker than a Bias frame. If mean(Dark) < mean(Bias), in the above formula we obtain a **negative value of dark current, which is clearly a physical nonsense.** In other cases the effect of Overscan/Optical Black calibration is not so extreme, but in any case **it leads to an underestimation of dark current**.

PixInsight offers a second method to measure the dark current, using two dark frames with very different exposures (the recommended ratio is 1:10). **Unfortunately, a bug in the script leads to a wrong value for the dark current when using this method** (Beltrame, PixInsight - BasicCCDParameters: flaw in calculation of dark current?, 2017) and, for long exposure dark frames and small readout noise, **the result of this script can be a lot smaller that the real value, thus providing unrealistic low values of dark current**. At the time of writing (November 11th, 2017) the bug is still present in the script (BasicCCDParameters v0.3.1).

AN ALTERNATIVE WAY TO MEASURE THE DARK CURRENT

I learned a possible workaround to get a more reliable measure of dark current from Christian Buil (Buil, 2017). The method consists in taking two dark frames **with the same duration and the same temperature**, let's call them D1 and D2. Then, you subtract D2 from D1 (in PixInsight use PixelMath – in ImageJ you use Image operations – Subtract). As the fixed pattern noise (and amp glow!) is the same in both images, by subtracting one from the other we eliminated its contribution to total noise and we are left with thermal noise (due to dark current random nature) with readout noise and with random telegraph signal (signal shot noise is obviously zero, as we are using dark frames).

Now, for the property of Poisson distribution we may calculate the total noise of the difference image as its standard deviation σ(D1-D2), that is:

$$\mathrm{\sigma}_{1-2} = \mathrm{\sigma}(\mathrm{D1} - \mathrm{D2}) = \sqrt{\mathrm{\sigma}(\mathrm{D1}) + \mathrm{\sigma}(\mathrm{D2})} = \sqrt{2}\,\mathrm{\sigma}'$$

$$\mathrm{\sigma}' = \frac{\mathrm{\sigma}_{1-2}}{\sqrt{2}}$$

Where I assumed that:

$$\sigma' \cong \sigma(D1) \cong \sigma(D2)$$

In other words, the standard deviation of the two dark frames is approximately the same and equal to σ'. This is very reasonable, because two darks taken in the same conditions should be (on average) very similar (of course, individual pixels may assume radically different values).

Now, for a dark frame we may write:

$$\mathrm{\sigma} = \sqrt{\mathrm{D_c t} + \mathrm{RON}^2 + FPNU^2 + RTS^2}$$

Where Dc*t is the dark signal (do not confuse it with dark noise, which is the square root of dark signal!), given by the product of dark current D_c times exposure time; RON is the readout noise; FPNU is the fixed pattern noise. We can also rewrite the above equation as:

$$\sigma^2 = D_c t + RON^2 + FPNU^2 + RTS^2$$

$$D_c t + RTS^2 = \sigma^2 - RON^2 - FPNU^2$$

We may find the value of RON from the sensor's datasheet (there are also several simple ways to measure it from flat and bias frames, but I will not cover that topic in this paper) and we can calculate σ for any given dark frame, but we cannot discriminate the contribution of dark current and RTS from the contribution of fixed pattern noise. However, if we subtract a dark frame from another dark frame, the fixed pattern noise cancels out while the standard deviation is given by the formula we find above, so we are left with:

$$D_c t + RTS^2 = \left(\frac{\sigma_{1-2}}{\sqrt{2}}\right)^2 - RON^2$$

The contribution of RTS to the total noise is generally small and we may decide to ignore it by putting:

$$RTS \cong 0$$

Finally, if we solve for D_c we find out that:

$$D_c = \frac{\left(\frac{\sigma_{1-2}}{\sqrt{2}}\right)^2 - RON^2}{t}$$

If we decide to disregard the approximation regarding the RTS, we may still note that RTS^2 is necessarily zero or greater than zero, which allows us to rewrite the above formula as an inequality:

$$D_c \le \frac{\left(\frac{\sigma_{1-2}}{\sqrt{2}}\right)^2 - RON^2}{t}$$

Here I'm expressing all variables in ADU instead of electrons. If you want to find dark current in electrons per second, simply multiply the dark current D_c by the square of the gain. Otherwise, in the above formula you can multiply the standard deviation σ_{1-2} by the gain G and enter the readout noise in electrons:

$$D_c \le \frac{\frac{(G\ \sigma_{1-2})^2}{2} - RON_{e-}^2}{t}$$

I though it is possible to extend Buil's formula by taking several darks and more precisely an even number, let's call them D_1, D_2, D_3, … D_n. Then, we construct a composite image D as:

$$D = D_1 - D_2 + D_3 - D_4 + \cdots + D_{n-1} - D_n$$

$$\sigma(D) = \sqrt{n}\,\sigma'$$

where again:

$$\sigma' \cong \sigma(D_1) \cong \sigma(D_2) \cong \cdots \cong \sigma(D_n)$$

By repeating the same steps above, we find that

$$D_c \le \frac{\frac{(G\ \sigma(D))^2}{n} - RON_{e-}^2}{t}$$

with the relevant difference that we are now working on several frames, so statistical fluctuations and the influence of measurement errors on single frames should be considerably reduced.

As an example, I recently evaluated the dark current of the QHY163M using the basic Buil's formula with two dark frames, obtaining a value of 0.006 electrons per second per pixel at -20°C. I repeated the measurement using up to 14 frames, obtaining the following results:

Frames	**Dark current [e-/s]**
2	0,0047
4	0,0054
6	0,0051
8	0,0047
10	0,0052
12	0,0053
14	0,0051

As you can see, there is an initial fluctuation, then the measured value seems to settle around 0.005 electrons per second per pixel, which is my new estimation for the dark current at -20°C.

A word of caution when you perform these calculations: be aware of upper bound and lower bound truncation, particularly if you are using integer variables. That's the reason way I chose to write the formula for the combined image by alternating plus and minus signs (instead of, for instance, to sum half the images and then to subtract the remaining half). It's better to use floating point math and 64-bit variables are even better.

Next, remember that we assume that RTS is zero. In this case RTS adds up several times, so the final result may be a little bit overestimated. To this end, it's better to work with long exposure darks, in such a way to have thermal noise much higher than RTS (in which case it is legitimate to assume RTS ≃ 0).

Part 2

DARK FRAME CALIBRATION WITH PIXINSIGHT AND CMOS CAMERAS

In Part 1 (Dark signal, overscan calibration and black point correction) I explained why the combined effect of overscan calibration, black point correction and amp glow may produce the counter-intuitive result that dark frames appear darker than bias frames. I also shown that the effect of amp glow may be non-linear with respect to exposure time.

If the calibration of light frames is not performed properly, the above consequence may lead to the following effects:

- **Clipping to zero**: this is due to the fact that the mean of Bias frame may be higher than the mean of Dark frames; as a consequence, if the calibration includes the subtraction of the Master Bias from the Master Dark (more on that later), it is likely that many pixel are clipped to zero (the obvious workaround in to add a pedestal to prevent clipping).

- **Ineffective removal of amp glow**: if calibration is performed with a Master Dark that was obtained with an exposure time that is not the same of Light frames, the resulting calibrated frames may be under- or over-corrected.

HOW PIXINSIGHT PERFORMS CALIBRATION (BY DEFAULT)

The recommended way to do image calibration in PixInsight is the following (Peris):

1. Integrate Bias frames into a Master Bias.

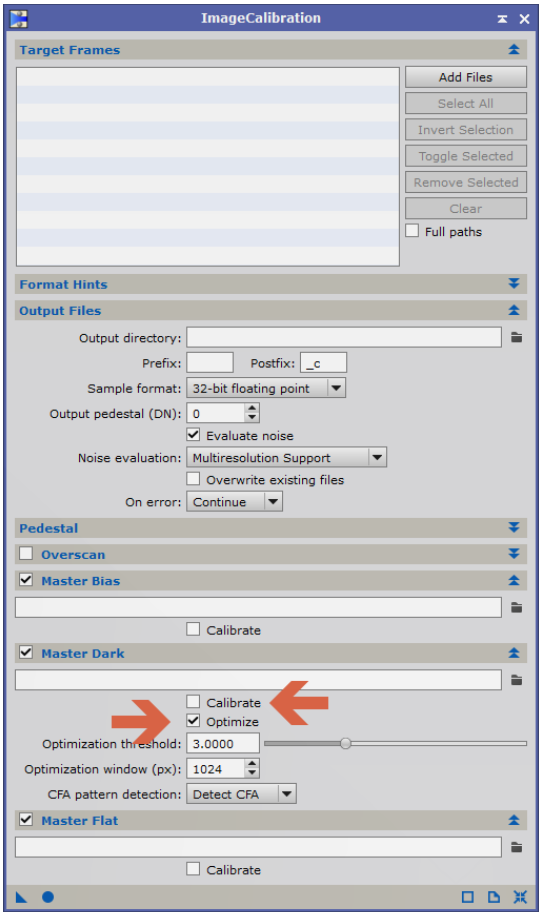
2. Integrate Dark frames into a Master Dark.

3. Calibrate Flat frames with Master Bias and Master Dark.

4. Integrate calibrated Flat frames into a Master Flat.

5. Calibrate Light frames with Master Bias, Master Dark and Master Flat.

This is the default process window for the ImageCalibration Process, the tool that PixInsight offers to perform image calibration:



☐ Full paths

Output directory:

Prefix: Postfix: _c

Sample format: 32-bit floating point

Output pedestal (DN): 0

☑ Evaluate noise

Noise evaluation: Multiresolution Support

☐ Overwrite existing files

On error: Continue

☐ Overscan

☑ Master Bias

☐ Calibrate

☑ Master Dark

☐ Calibrate

☑ Optimize

Optimization threshold: 3.0000

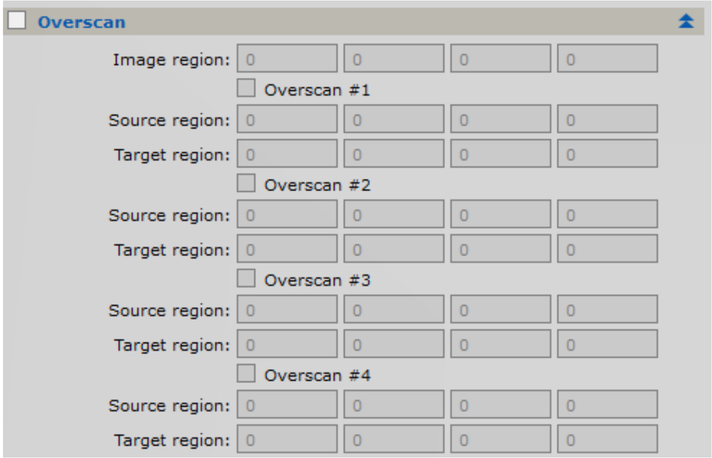
Optimization window (px): 1024

CFA pattern detection: Detect CFA

☑ Master Flat

☐ Calibrate

The nice thing is that PixInsight offers a full-fledged Overscan calibration feature, that allows to define up to 4 overscan regions:



☐ Overscan

Image region: 0 0 0 0

☐ Overscan #1

Source region: 0 0 0 0

Target region: 0 0 0 0

☐ Overscan #2

Source region: 0 0 0 0

Target region: 0 0 0 0

☐ Overscan #3

Source region: 0 0 0 0

Target region: 0 0 0 0

☐ Overscan #4

Source region: 0 0 0 0

Target region: 0 0 0 0

But I'd like you to focus on the two options under Master Dark tab, those marked with a red arrow: **Calibrate** and **Optimize**. The Calibrate option enables the overscan correction (if the relevant information has been entered) and **subtracts the Bias from the Dark**. The Optimize option scales the Master Dark in order to meet the exposure time and temperature of Target frames. In detail:

"[You] don't [need to] worry about differing temperatures and exposure times between dark and light frames. [ImageCalibration process] will always rescale the dark noise to match every light frame [...]

thermal noise must be bias-subtracted: only thermal noise must be rescaled in the master dark to match the thermal noise in the light frame." (Peris)

Therefore, the standard way of doing image calibration in PixInsight may be summarized with the following formula:

$$Calibrated\ light\ = \{Light\ -\ Bias\ -\ k(Dark - Bias)\}\ \frac{mean(Flat)}{Flat\ -\ Bias}$$

where the constant k represents the scaling factor of the Dark frames; also, the term mean(Flat) normalizes the Master Flat in order to preserve the brightness of the Light frames. If we ignore Flat normalization and do not apply scaling, the formula becomes:

$$Calibrated\ light\ = \frac{Light\ -\ Dark}{Flat\ -\ Bias}$$

which is the more common way of doing image calibration.

The need of scaling Dark frames arises when the Darks and the Lights have different features (exposure duration and temperature), so a simple subtraction would not provide the desired result. Suppose that the Master Dark has a 600-second exposure time, while the Light frames have been exposed for 300 seconds. It is reasonable to subtract only half of the Dark, so k = 0.5.

However, you have to note that Dark frames are affected by read noise, so they "incorporate" a Bias frame. In the last formula we subtracted the Master Dark from Light frames, so we automatically subtracted the Bias too. But if we scale the Master Dark, for instance with k=0.5, that is no longer true. That's why PixInsight, by default, subtracts the Bias from the Lights before applying the scaled version of (Dark – Bias) (from the above citation, only the thermal noise must be scaled, so the Bias noise must be removed from the Dark before scaling).

That's all well and good for calibration of images gathered through a CCD sensor. But with CMOS sensors the story can be different and in the case of cameras where the dark frames are darker than bias frames, scaling simply does not work because the hypothesis at its foundations (linearity of output) is simply false for these cameras.

HOW TO CALIBRATE WITH PIXINSIGHT (STEP BY STEP)

Let's start with a FITS image that I recently shoot with the QHY163M and a 6-nm H-alpha filter (it's the well-known Heart nebula, or IC 1805, in Cassiopeia). The camera is monochrome, so it's a grey scale image, and the exposure was 10 minutes at f/5.3 and a temperature of -20°C. It's a pure raw file, I only applied the default screen transfer function (STF) of PixInsight:

In this picture it is easy to overlook the area where amp glow is more prominent (it's indicated by the arrows). **To calibrate my light frames I took several dark frames at the same identical temperature (-20°C).** I also used a library of 200 Bias frames to build a Master Bias and 20 flat frames at the same temperature (-20°C) with an exposure of 9 seconds.

In the following pages I will show how to integrate the light frames in order to remove any hint of amp glow.

The most important things to consider are the following:

- We learned that, with CMOS cameras that present amp glow and black point calibration, **it is mandatory to use dark frames with the same temperature and exposure duration of light frames**. Also, you must **avoid any form of dark frame scaling**.
- **Calibrate flats with Master Bias only**. If you choose to calibrate with the Master Dark too, it will be scaled and, again, we don't want that to happen. If you're paranoid about calibration and you want to remove thermal noise from flats, then you need to build a different Master Dark (i.e. a Master Flat Dark) for that purpose (in my opinion, if you're using a light panel for flats and your flat exposures are short, flat darks are really overkill).

As a consequence, we will calibrate using this formula:

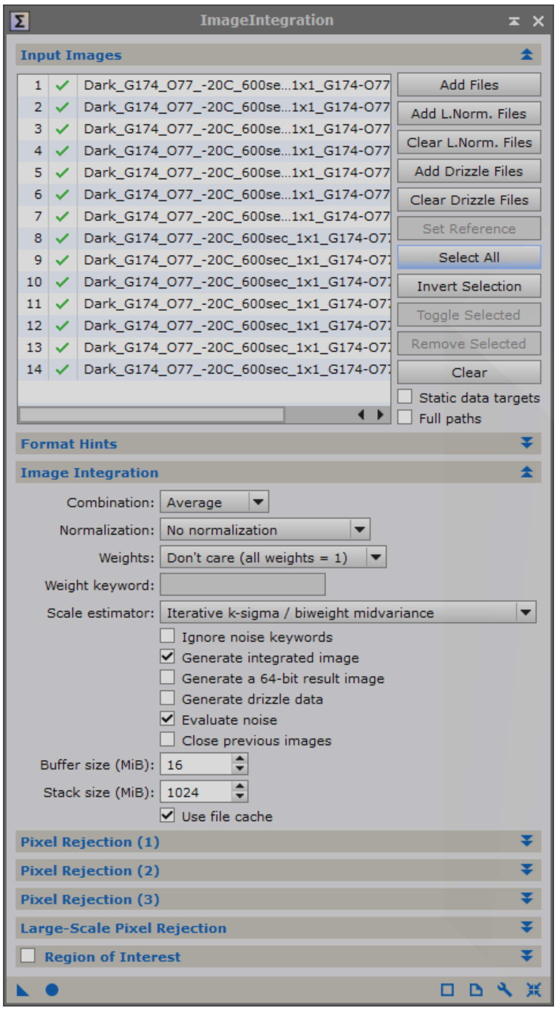
$$Calibrated\ light = \frac{Light - Dark}{Flat - Bias}$$

INTEGRATION OF THE MASTER BIAS AND THE MASTER DARK

As the first step, we need to integrate the Bias frames into a Master Bias and the dark frames into a Master Dark. The procedure is the same for the two cases and its performed through the ImageIntegration process. We have to change some of the default values (Peris):

- Don't normalize the images, because the bias pedestal must be preserved. Both normalization methods in the Image Integration and Pixel Rejection sections must be disabled (No normalization setting).
- Disable the image weighting feature. We want to reject only clear outliers.
- Use Winsorized Sigma Clipping algorithm to reject outliers (try to use at least 10-15 frames).

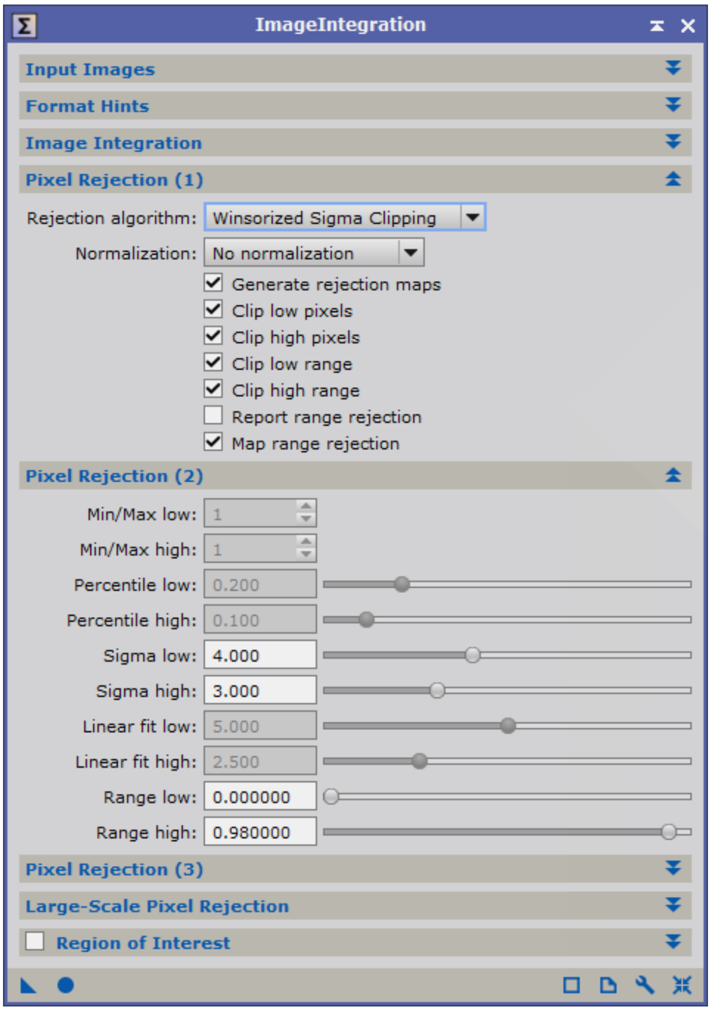
Here is how my ImageIntegration process window is set (on the left the details of the Pixel Rejection parameters.



1 ✓ Dark_G174_O77_-20C_600se...1x1_G174-O77
2 ✓ Dark_G174_O77_-20C_600se...1x1_G174-O77
3 ✓ Dark_G174_O77_-20C_600se...1x1_G174-O77
4 ✓ Dark_G174_O77_-20C_600se...1x1_G174-O77
5 ✓ Dark_G174_O77_-20C_600se...1x1_G174-O77
6 ✓ Dark_G174_O77_-20C_600se...1x1_G174-O77
7 ✓ Dark_G174_O77_-20C_600se...1x1_G174-O77
8 ✓ Dark_G174_O77_-20C_600sec_1x1_G174-O7
9 ✓ Dark_G174_O77_-20C_600sec_1x1_G174-O7
10 ✓ Dark_G174_O77_-20C_600sec_1x1_G174-O7
11 ✓ Dark_G174_O77_-20C_600sec_1x1_G174-O7
12 ✓ Dark_G174_O77_-20C_600sec_1x1_G174-O7
13 ✓ Dark_G174_O77_-20C_600sec_1x1_G174-O7
14 ✓ Dark_G174_O77_-20C_600sec_1x1_G174-O7

☐ Static data targets
☐ Full paths

Combination: Average
Normalization: No normalization
Weights: Don't care (all weights = 1)
Weight keyword:
Scale estimator: Iterative k-sigma / biweight midvariance
☐ Ignore noise keywords
☑ Generate integrated image
☐ Generate a 64-bit result image
☐ Generate drizzle data
☑ Evaluate noise
☐ Close previous images
Buffer size (MiB): 16
Stack size (MiB): 1024
☑ Use file cache



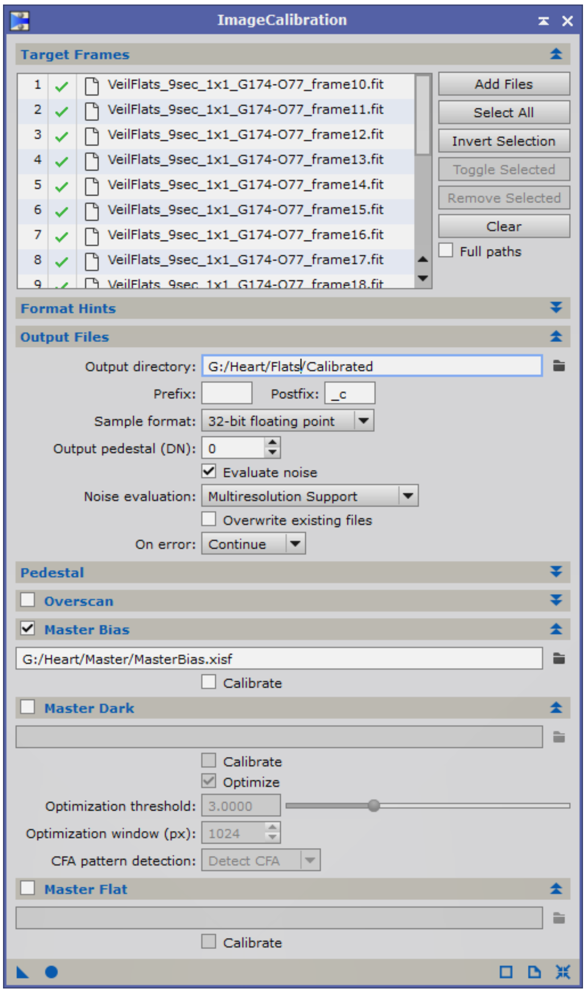
Rejection algorithm: Winsorized Sigma Clipping
Normalization: No normalization
☑ Generate rejection maps
☑ Clip low pixels
☑ Clip high pixels
☑ Clip low range
☑ Clip high range
☐ Report range rejection
☑ Map range rejection

Min/Max low: 1
Min/Max high: 1
Percentile low: 0.200
Percentile high: 0.100
Sigma low: 4.000
Sigma high: 3.000
Linear fit low: 5.000
Linear fit high: 2.500
Range low: 0.000000
Range high: 0.980000

The output of the process is an image that we'll rename **Master Bias** or **Master Dark** depending on which type of frame we integrated. Now we are ready to calibrate the Flats.

FLAT CALIBRATION

As the next step, we calibrate flat frames. As discussed above, we will only subtract the Master Bias by selecting it under the Master Bias tab. Be sure to disable Master Dark and Master Flat and to choose a suitable destination directory for your calibrated flats.



1 ✓ VeilFlats_9sec_1x1_G174-O77_frame10.fit
2 ✓ VeilFlats_9sec_1x1_G174-O77_frame11.fit
3 ✓ VeilFlats_9sec_1x1_G174-O77_frame12.fit
4 ✓ VeilFlats_9sec_1x1_G174-O77_frame13.fit
5 ✓ VeilFlats_9sec_1x1_G174-O77_frame14.fit
6 ✓ VeilFlats_9sec_1x1_G174-O77_frame15.fit
7 ✓ VeilFlats_9sec_1x1_G174-O77_frame16.fit
8 ✓ VeilFlats_9sec_1x1_G174-O77_frame17.fit
9 ✓ VeilFlats_9sec_1x1_G174-O77_frame18.fit

☐ Full paths

Output directory: G:/Heart/Flats/Calibrated
Prefix:
Postfix: _c
Sample format: 32-bit floating point
Output pedestal (DN): 0
☑ Evaluate noise
Noise evaluation: Multiresolution Support
☐ Overwrite existing files
On error: Continue

☐ Overscan

☑ Master Bias
G:/Heart/Master/MasterBias.xisf
☐ Calibrate

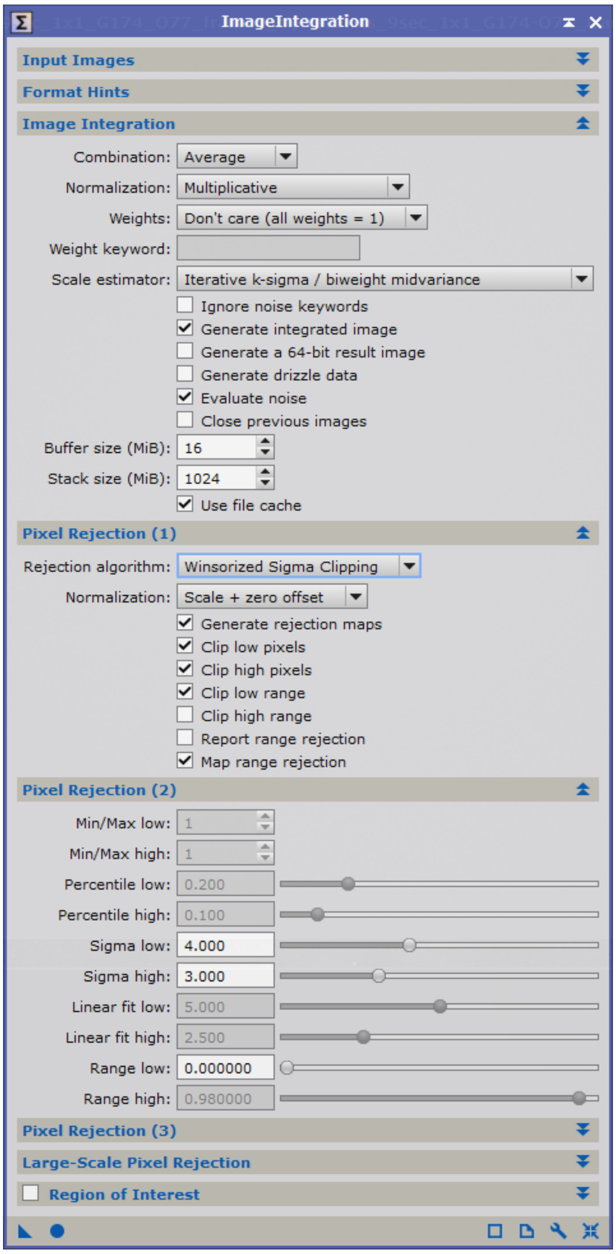
☐ Master Dark
☐ Calibrate
☑ Optimize
Optimization threshold: 3.0000
Optimization window (px): 1024
CFA pattern detection: Detect CFA

☐ Master Flat
☐ Calibrate

INTEGRATION OF THE MASTER FLAT

Now we are ready to integrate Flat frames. There is only a couple of things to change with respect to the integration of Master Bias and Master Dark:

- We select Multiplicative Normalization to generate a Master that will not change the overall brightness of our Lights.
- If the number of flats is small or you have used the “Sky Flat” technique, then it is better to use the Percentile Clipping method for rejection of outliers. I used an iPad as a “light panel”, so in this example I will stick with Winsorized Sigma Clipping.



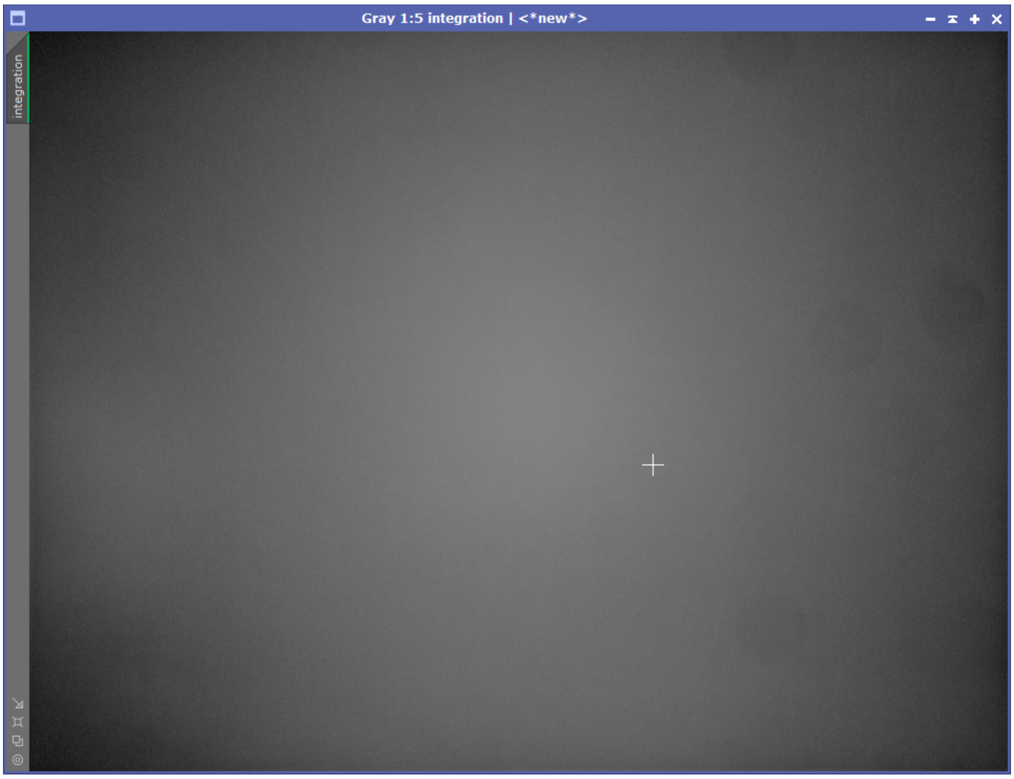
Combination: Average
Normalization: Multiplicative
Weights: Don't care (all weights = 1)
Weight keyword:
Scale estimator: Iterative k-sigma / biweight midvariance
☐ Ignore noise keywords
☑ Generate integrated image
☐ Generate a 64-bit result image
☐ Generate drizzle data
☑ Evaluate noise
☐ Close previous images
Buffer size (MiB): 16
Stack size (MiB): 1024
☑ Use file cache

Rejection algorithm: Winsorized Sigma Clipping
Normalization: Scale + zero offset
☑ Generate rejection maps
☑ Clip low pixels
☑ Clip high pixels
☑ Clip low range
☐ Clip high range
☐ Report range rejection
☑ Map range rejection

Min/Max low: 1
Min/Max high: 1
Percentile low: 0.200
Percentile high: 0.100
Sigma low: 4.000
Sigma high: 3.000
Linear fit low: 5.000
Linear fit high: 2.500
Range low: 0.000000
Range high: 0.980000

☐ Region of Interest

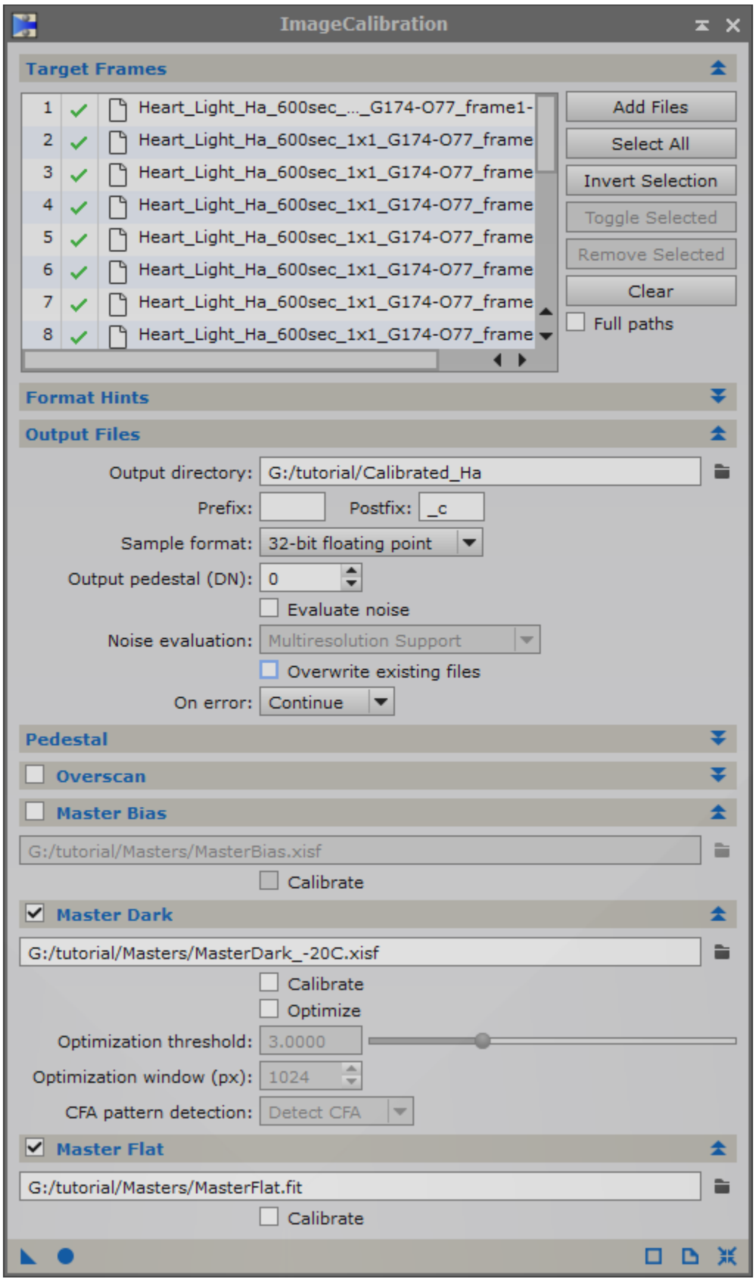
After running the process with obtain our **Master Flat**:



CALIBRATION OF LIGHT FRAMES

We are finally ready to perform the calibration of Light Frames. The options to choose are:

- ★ Disable Master Bias (we will not subtract the Bias from the Lights, while Flats have already been Bias-subtracted in their calibration).
- ★ Enable Master Dark and select the Master Dark file we created above.
- ★ Under Master Dark, be sure to **disable both Calibrate and Optimize**.
- ★ Enable Master Flat, and select the Master Flat we created above; disable Calibrate.



1 ✓ Heart_Light_Ha_600sec_..._G174-O77_frame1-
2 ✓ Heart_Light_Ha_600sec_1x1_G174-O77_frame
3 ✓ Heart_Light_Ha_600sec_1x1_G174-O77_frame
4 ✓ Heart_Light_Ha_600sec_1x1_G174-O77_frame
5 ✓ Heart_Light_Ha_600sec_1x1_G174-O77_frame
6 ✓ Heart_Light_Ha_600sec_1x1_G174-O77_frame
7 ✓ Heart_Light_Ha_600sec_1x1_G174-O77_frame
8 ✓ Heart_Light_Ha_600sec_1x1_G174-O77_frame

☐ Full paths

Output directory: G:/tutorial/Calibrated_Ha
Prefix:
Postfix: _c
Sample format: 32-bit floating point
Output pedestal (DN): 0
☐ Evaluate noise
Noise evaluation: Multiresolution Support
☐ Overwrite existing files
On error: Continue

☐ Overscan

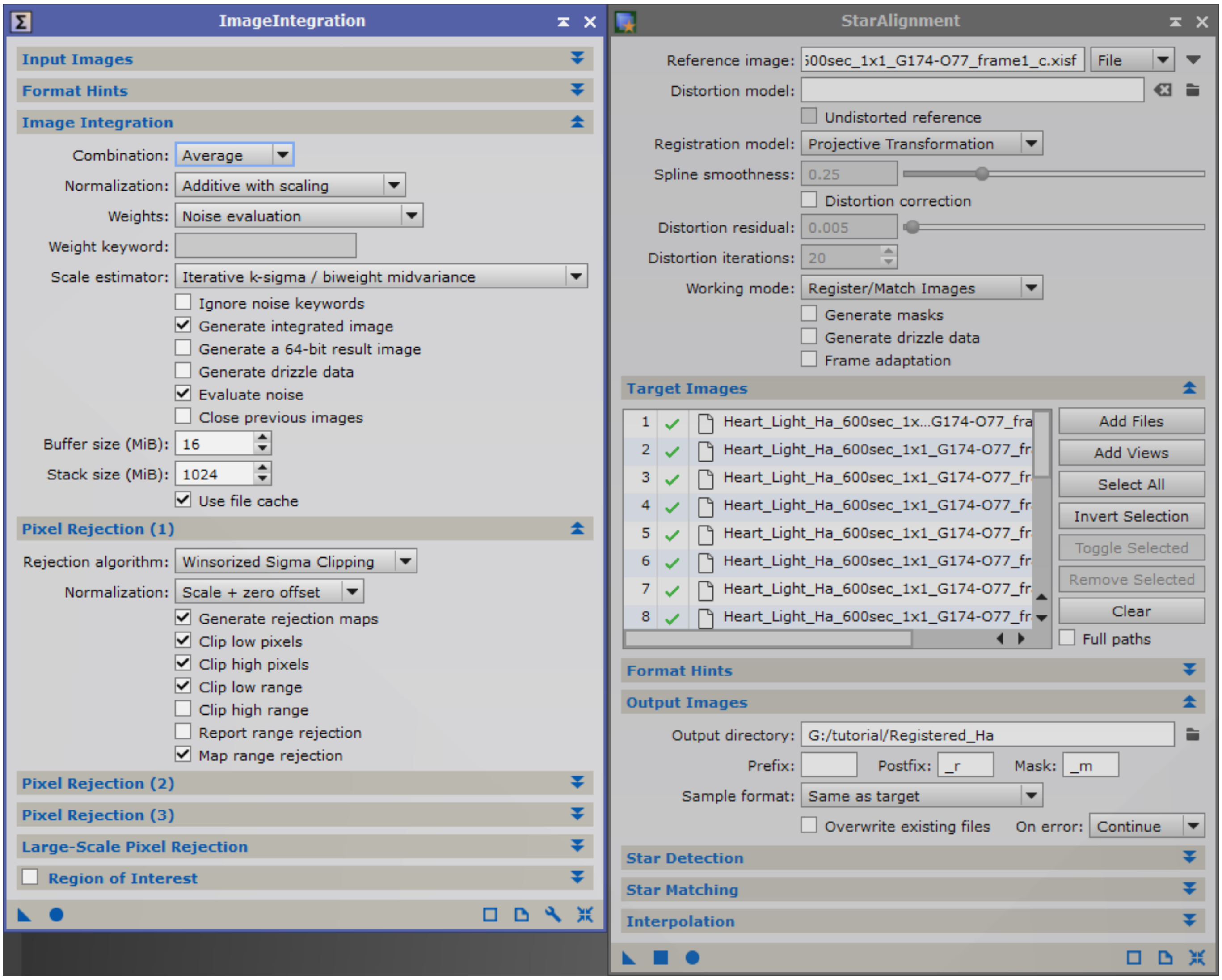
☐ Master Bias
G:/tutorial/Masters/MasterBias.xisf
☐ Calibrate

☑ Master Dark
G:/tutorial/Masters/MasterDark_-20C.xisf
☐ Calibrate
☐ Optimize
Optimization threshold: 3.0000
Optimization window (px): 1024
CFA pattern detection: Detect CFA

☑ Master Flat
G:/tutorial/Masters/MasterFlat.fit
☐ Calibrate

REGISTRATION AND INTEGRATION OF LIGHT FRAMES

Now that we have our calibrated Light, we can proceed with registration and integration. Just go on as you usually do. For this tutorial I used default values and I selected Winsorized Sigma Clipping as the rejection algorithm in integration.



Combination: Average

Normalization: Additive with scaling

Weights: Noise evaluation

Weight keyword:

Scale estimator: Iterative k-sigma / biweight midvariance

☐ Ignore noise keywords
☑ Generate integrated image
☐ Generate a 64-bit result image
☐ Generate drizzle data
☑ Evaluate noise
☐ Close previous images

Buffer size (MiB): 16

Stack size (MiB): 1024

☑ Use file cache

Rejection algorithm: Winsorized Sigma Clipping

Normalization: Scale + zero offset

☑ Generate rejection maps
☑ Clip low pixels
☑ Clip high pixels
☑ Clip low range
☐ Clip high range
☐ Report range rejection
☑ Map range rejection

Reference image: 00sec_1x1_G174-O77_frame1_c.xisf File

Distortion model:

☐ Undistorted reference

Registration model: Projective Transformation

Spline smoothness: 0.25

☐ Distortion correction

Distortion residual: 0.005

Distortion iterations: 20

Working mode: Register/Match Images

☐ Generate masks
☐ Generate drizzle data
☐ Frame adaptation

1	✓	Heart_Light_Ha_600sec_1x...G174-O77_fra
2	✓	Heart_Light_Ha_600sec_1x1_G174-O77_fr
3	✓	Heart_Light_Ha_600sec_1x1_G174-O77_fr
4	✓	Heart_Light_Ha_600sec_1x1_G174-O77_fr
5	✓	Heart_Light_Ha_600sec_1x1_G174-O77_fr
6	✓	Heart_Light_Ha_600sec_1x1_G174-O77_fr
7	✓	Heart_Light_Ha_600sec_1x1_G174-O77_fr
8	✓	Heart_Light_Ha_600sec_1x1_G174-O77_fr

Output directory: G:/tutorial/Registered_Ha

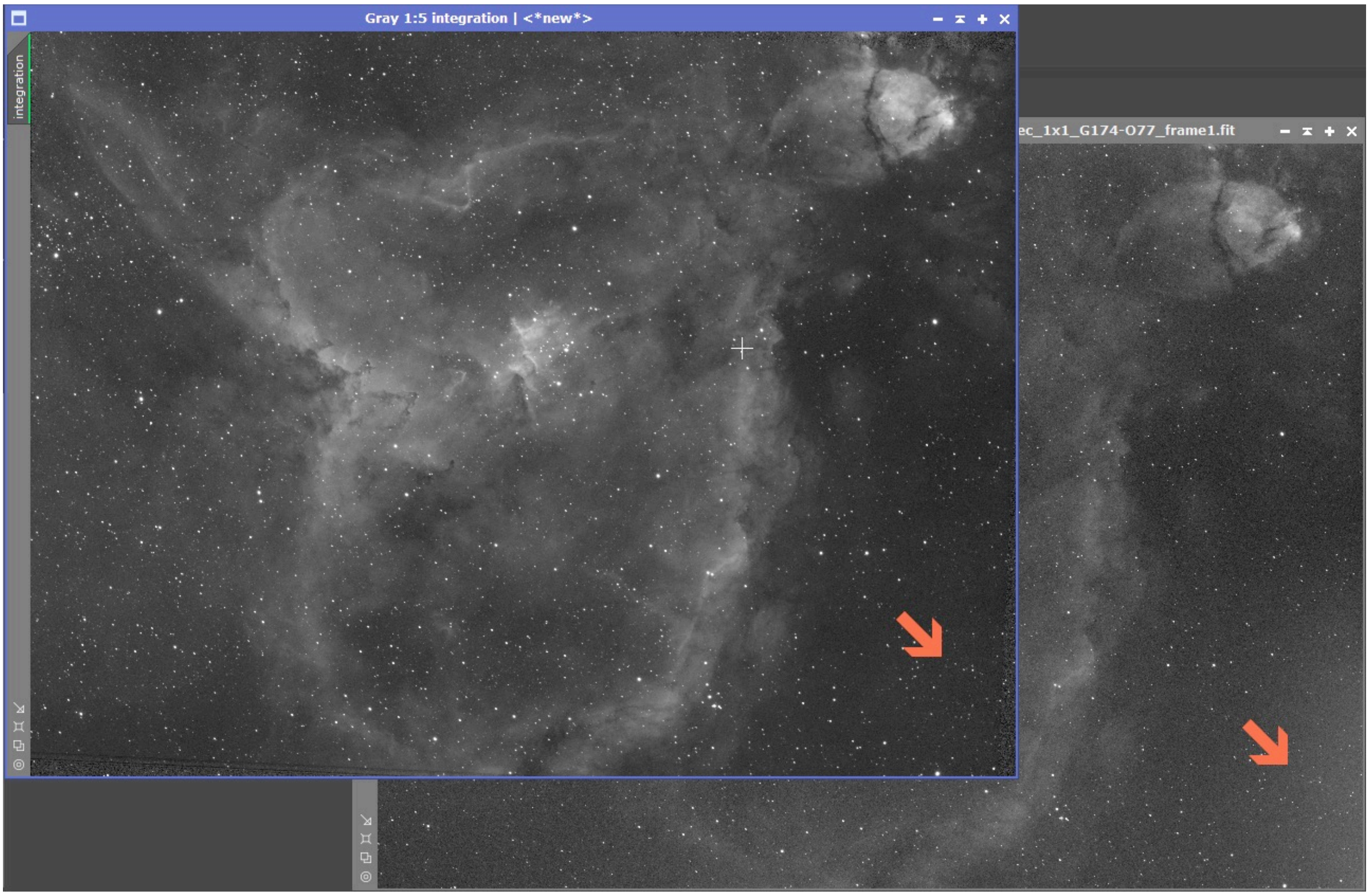
Prefix: Postfix: _r Mask: _m

Sample format: Same as target

☐ Overwrite existing files On error: Continue

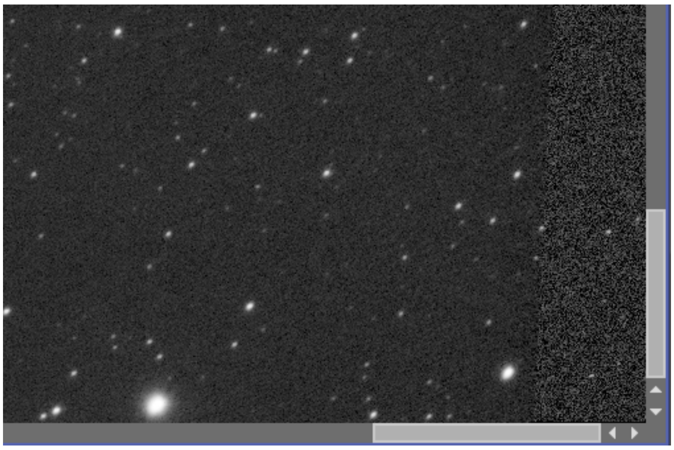
THE FINAL RESULT

In the following images you can appreciate the final result of the integration of 18 exposures of 10 minutes each, for a total of 3 hours of integration.



There no processing whatsoever in the above screenshot. To the integrated H-alpha image (on the left) and a single Light frame (on the right) I only applied a default screen stretch (STF – ScreenTransferFunction) and it is easy to see how the amp glow has been completely eliminated by the calibration of light frames.

The artifacts that you may notice in the bottom right corner and in the lower edge are due to image registration (I imaged the Heart Nebula in two different nights, so there is a slight rotation between the light frames). This is what you can see at 100% in the bottom right corner, where the “ghost” of frames with different orientation becomes clearly visible:



Still, the sky background is nice and black (or better, dark gray) as it should be. Just compare it with the corner of the single light frame, where amp glow shows up:

But, in general, the procedure works really well with this camera. Here is a 100% close-up of Melotte 15 at the center of the nebula: just take a look and judge for yourself if the techniques I presented are good enough for your purposes.

Part 3

BIBLIOGRAPHY

Beltrame, A. (2017). *QHY163M review.*
Retrieved from www.alessiobeltrame.com:
http://www.alessiobeltrame.com/qhy163m

Beltrame, A. (2017). *PixInsight - BasicCCDParameters: flaw in calculation of dark current?*
Retrieved from https://pixinsight.com/forum/index.php?topic=11800.msg72604 - msg72604

Buil, C. (2017). *CMOS vs CCD*.
Retrieved from http://www.astrosurf.com/buil/CMOSvsCCD/index.html

Hongyun, Q. (2017). *Why Dark Field Is Darker Than Bias Field*.
Retrieved from QHYCCD:
http://qhyccd.com/bbs/index.php?action=dlattach;topic=6159.0;attach=9024

Hornsey, R. (n.d.). *Noise in Image Sensors.*
Retrieved from https://ece.uwaterloo.ca/~ece434/Winter2008/Noise.pdf

Martinec, E. (n.d.). *Noise, Dynamic Range and Bit Depth in Digital SLRs.*
Retrieved from theory.uchicago.edi:
http://theory.uchicago.edu/%7Eejm/pix/20d/tests/noise/

Nakamura, J. (2006).
Image sensors and signal processing for digital still cameras.
Taylor & Francis.

Peris, V. (n.d.). *Master Calibration Frames: Acquisition and Processing.*
Retrieved from www.pixinsight.com:
https://www.pixinsight.com/tutorials/master-frames/index.html

Stark, C. (n.d.). *Pre-Processing images in Nebulosity.*
Retrieved from www.stark-labs.com:
http://www.stark-labs.com/help/nebulositytutorials/files/Processing_HowTo.pdf