

HBComposer - User Guide

Version 1.7.1 - © 2026 Marco Ristuccia



Contents

1	Introduction	1
2	Product installation and execution	1
2.1	Getting HBComposer	1
2.2	Installing HBComposer (Updated)	1
2.2.1	macOS (Apple silicon)	1
2.2.2	Windows (x86, 64-bit)	2
2.3	Installing Java runtime (Updated)	2
2.4	Running HBComposer	2
3	User interface overview	5
3.1	Operation	5
3.2	File Type	5
3.3	Frame Weight Shaping	6
3.4	Source	6
3.5	LCC / Flat-Field	6
3.6	Activity	6
3.7	Performance parameters	6
4	Sidecar files	7
5	Operations	9
5.1	Frame-Average	9
5.2	Frame-Median	10
5.3	Frame Alignment (Beta)	11
5.4	Frame Weight Shaping	11
5.5	Flat-Field Correction	12
5.5.1	Goals of flat-field correction	13
5.5.2	Operations and flat-field correction	16
5.5.3	Flat-field correction settings	16
5.5.4	Flat-Field (LCC) Presets	21
6	HBComposer Plugin for Adobe Lightroom®	22
6.1	Plugin installation	22
6.2	Plugin configuration	22
6.2.1	Configuration parameters	23
6.3	Plugin execution	24
6.4	Plugin uninstallation	27

List of Figures

1	HBComposer installation window (macOS)	2
2	HBComposer installation window (Windows, first step, press "More info")	3
3	HBComposer installation window (Windows, first step, press "Run anyway")	3
4	macOS Privacy & Security settings	4
5	HBComposer - Main window	5
6	HBComposer - Settings window	7
7	(a) Single frame (b) Frame-average of 160 RAW images	9
8	(a) Single frame (b) Frame-median of 34 RAW images	10
9	Frame weight shaping controls	11
10	(a) Frame-median with no oversampling – gaps are present in the clouds (b) Frame-median with 16X oversampling – gaps in the clouds are smoothed out	12
11	Histogram of a well-exposed flat-field (LCC) image	13
12	Example of Flat-Field (LCC) image	13
13	Flat-field application process	14
14	(a) Original image - horizontal PDAF banding and light falloff present (b) Processed image - horizontal PDAF banding and light falloff corrected	14
15	(a) Original image - sensor tiling, color cast and vignetting present (b) Processed image - sensor tiling, color cast and light falloff corrected	15
16	(a) Original image - dust spot and light falloff present (b) Processed image - dust spot and light falloff corrected	15
17	HBComposer - Flat-field structural settings (<i>Structural</i> panel)	16
18	HBComposer - Flat-field color cast correction settings (<i>Color</i> panel)	17
19	HBComposer - Flat-field light falloff correction settings (<i>Light Falloff</i> panel)	18
20	HBComposer - Flat-field correction settings (<i>Reference</i> panel)	18
21	A scheme indicating how to use the Ref Point sliders to place the LCC reference point.	19
22	HBComposer - Flat-field correction settings (<i>Thresholds</i> panel)	20
23	HBComposer - flat-field correction presets	21
24	HBComposer - Install Lightroom Plugin	23
25	HBComposer - Plugin Installation Result	23
26	HBComposer - Plug-in Manager menu item	24
27	HBComposer - Plug-in configuration window	25
28	HBComposer - Plugin actions	26
29	HBComposer - Uninstall Lightroom plugin	27
30	HBComposer - Plugin uninstallation result	28

1 Introduction

HBComposer is an unofficial tool for pre-processing *Hasselblad* RAW files. It is able to perform the following *3FR* to *3FR* operations in the RAW domain:

- Frame-Average
- Frame-Median
- Advanced Flat-Field correction

Since the produced files are still raw files, they can be further processed in any RAW developer, especially in *Hasselblad Phocus* **without losing the advantage of the *Hasselblad Natural Color Solution* (HNCS)**. The advanced flat-field correction is able to fix the infamous *Hasselblad PDAF banding* issue and any **sensor tiling** artifacts generated by digital backs when used in conjunction with technical cameras and shifted wide-angle lenses.

***** DISCLAIMER ***** *I am in no way connected to Hasselblad, nor does Hasselblad endorse this project. This tool is the result of my personal initiative and the files produced are not guaranteed to be 100% OEM compliant. Moreover, Hasselblad may change their RAW file specs at any moment causing the tool to stop working.*

2 Product installation and execution

2.1 Getting HBComposer

HBComposer is offered for free. You can download it from my web site under the following link:

<https://photography.marcoristuccia.com/hbcomposer-an-hasselblad-raw-file-composer/>

2.1.0.1 Support my work *Please consider donating to allow me to spend more time improving it and fixing any bugs. Thank you!*

2.2 Installing HBComposer (Updated)

HBComposer is available for Macs with Apple silicon and for 64-bit (x86) Windows.

2.2.1 macOS (Apple silicon)

Open the disk image (*hbcomposer-version.dmg*), review and accept the license and then drag *Disto.app* to Applications (see *Figure 1*).

The app is signed with an Apple Developer ID and *notarised*: Apple has checked it for malicious software and vouched for it, and macOS verifies this the first time the app opens. Because *disto* is not distributed through the Mac App Store, macOS only lets it open if your Mac accepts apps from identified developers as well as from the App Store. If macOS refuses to open the app, open **System Settings** → **Privacy & Security**, and under **Security** set **Allow applications from to App Store & Known Developers**. The first time you open *disto*, macOS reminds you that it was downloaded from the internet; click **Open**. It does not ask again.

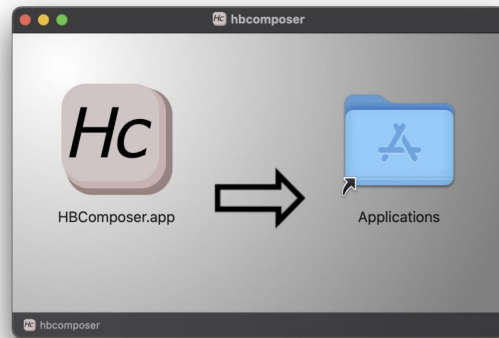


Figure 1: HBComposer installation window (macOS)

2.2.2 Windows (x86, 64-bit)

Run the installer (`HBComposer-version-setup.exe`). It installs disto for all users in `C:\Program Files\Disto`, adds Disto to the Start menu, and adds an entry under `Settings → Apps` from which you can remove it again. Installing needs administrator rights, so Windows asks you to confirm before the installer starts. To update disto, run the installer of the new version: it replaces the previous one. On Windows on Arm the x86 version runs under Windows' built-in emulation.

Note. I do not own a Windows code-signing certificate, so the installer is not signed. Windows therefore cannot tell who published it and warns you before running it. If you want to install disto, you have to accept that risk and tell Windows to go ahead anyway:

- Your browser may flag the download as uncommon or unsafe. Choose to keep it (in Microsoft Edge: `... → Keep`, then `Show more → Keep anyway`).
- When you run the installer, Microsoft Defender SmartScreen shows Windows protected your PC. Click `More info`, check that the file is the one you downloaded, then click `Run anyway`.
- The User Account Control prompt that follows shows the publisher as `Unknown`. Click `Yes` to install.

See *Figure 2* and *Figure 3*.

2.3 Installing Java runtime (Updated)

Starting from version 1.7.0, *HBComposer* is a self-contained application and **does not need an external *Java* runtime installed on the system anymore!**

2.4 Running HBComposer

Once installed, you will find *HBComposer* among the *macOS* applications. Just click on it. Please be patient, the application startup may take a few seconds since *macOS* needs to verify the application and instantiate the Java runtime environment.



Figure 2: HBComposer installation window (Windows, first step, press "More info")



Figure 3: HBComposer installation window (Windows, first step, press "Run anyway")

Since I'm officially part of the Apple Developer Program, I have digitally signed and notarized *HBComposer* as per *Apple* recommendations. Security wise, *macOS* should not complain about opening it, apart from warning you that the application has been downloaded from the *Internet* and asking for your authorization to proceed.

Since the application will not be downloaded from the official *App Store*, however, you may need to slightly relax the security settings by allowing applications from “**App Store & Known**

Developers”.

Such an option can be found under **System Settings -> Privacy & Security** as shown in *Figure 4*.

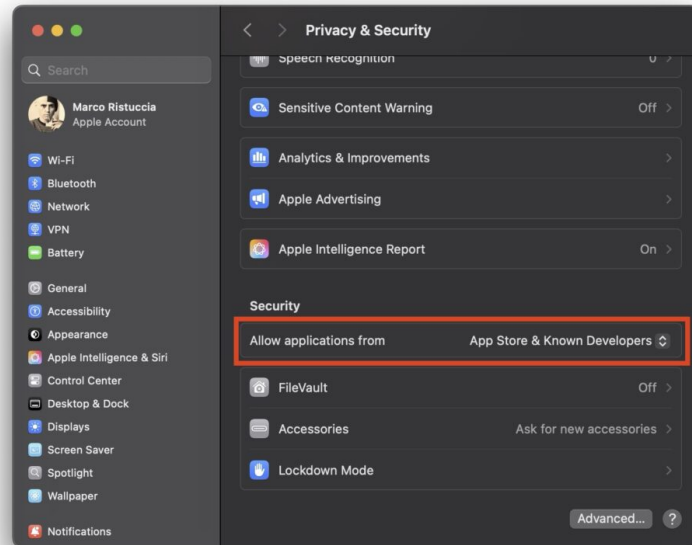


Figure 4: macOS Privacy & Security settings

3 User interface overview

In *Figure 5*, a screenshot of the main application window is shown.

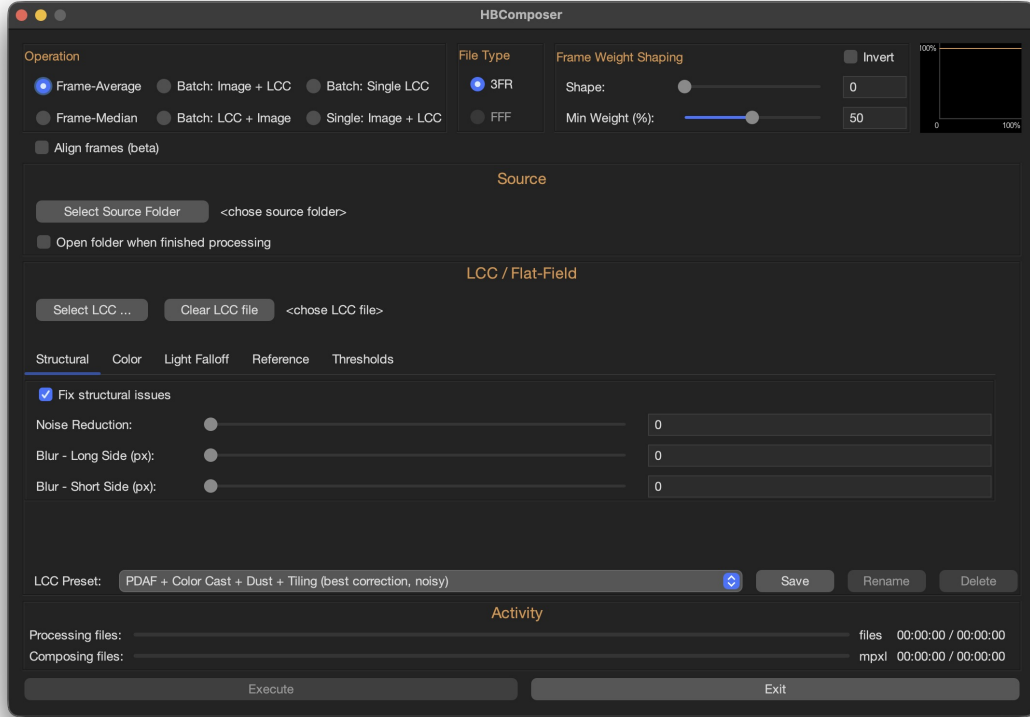


Figure 5: HBComposer - Main window

In the following sections, the general configurations are described first. Then, each operation will be fully explained in dedicated sections.

3.1 Operation

This area shows available operations to choose from.

See the dedicated *section 5* below in this document for more details on each operation.

3.2 File Type

The tool has been initially designed to operate on both *3FR* and *FFF Hasselblad* RAW files.

Due to technical reasons though, only *3FRs* are supported at the moment. For this reason, the *FFF* option is grayed out and not selectable.

3.3 Frame Weight Shaping

This area allows you to shape how frames are weighted during a frame-average or a frame-median operation.

See the dedicated *subsection 5.4* below in this document for more details.

3.4 Source

Depending on the chosen type of operation, a button is presented here to select the working file(s).

- **Files**

Used for batch operations. All images to be processed must be selected. You have the option to automatically open the containing folder after the processing is finished.

***NOTE:** starting from version 1.6.0, the source of batch operations has been switched from entire folder to files selection. From now on, all files to be processed need to be selected instead of the containing folder. This should give more flexibility. It's important to note that the order in which images are selected matters for the "Batch: Image + LCC" and "Batch: LCC + Image" operations. If a range is selected, images are processed from the first to the last as they are shown in the select window. If you select images one by one, the processing order will follow your selection order.*

- **File**

A single image to be processed. You have the option to automatically open the file folder after the processing is finished.

3.5 LCC / Flat-Field

In this area an LCC file to be applied to the processed image can be chosen. A set of parameters controls how the LCC correction is performed. See the dedicated section below for more details.

3.6 Activity

In the bottom area of the application window there are progress bars that are updated while processing. The composition operations will process files first, then compose them, so both progress bars are updated. The batch *LCC* operations only update the first progress bar.

3.7 Performance parameters

Performance parameters can be accessed by selecting the menu bar *HBComposer -> Settings....*

These parameters only apply to the *frame-average* and *frame-median* operations.

There are a couple of them for each of the two operations:

- **Bucket Size**

Represents the sliding memory window which buffers the part of the image that is kept in memory while processing the frame compositions. It is represented in bytes. The default value is a good compromise for the majority of systems. If your system happens to have a large amount of RAM (> 36GB) you can try increasing this value (keep it a multiple of 4096). This may speed up the composition operation. Be aware however that you may experience memory

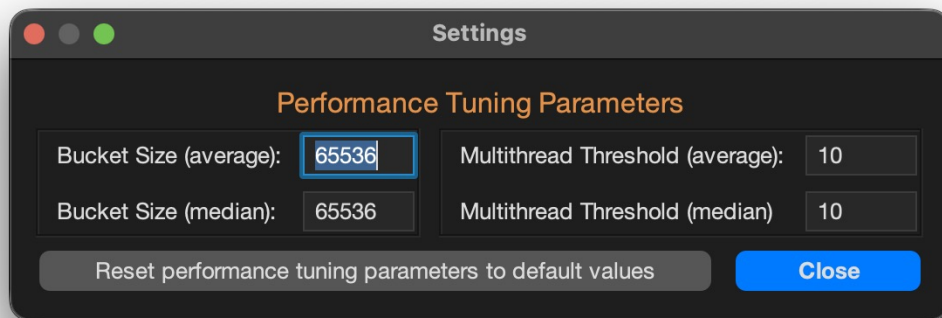


Figure 6: HBComposer - Settings window

over-allocations, in such case the application may slow down or even crash with an unexpected exception. If this happens, just restart *HBComposer* and reset to the default performance values. If you decide to venture into changing this parameter, you'll need to find the sweet spot.

- **Multithread Threshold**

Represents the maximum number of parallel processing threads. The default value is a good compromise for the majority of systems. If your system has a large number of CPU cores (> 14) you can try increasing it. This may speed up the composing operation. If you decide to venture into changing this parameter, you'll need to find the sweet spot.

NOTE: all performance parameters are remembered after closing and reopening the application. They can be reset to their default values by clicking the dedicated button.

4 Sidecar files

RAW files are never altered by RAW processors. Instead, changes are either stored in a database (e.g. in *Lightroom*) or in a sidecar file (e.g. *Phocus* and *Lightroom*). *Sidecar* files created by RAW processors are associated with any edited RAW image and contain all its editing history.

Starting from version 1.4.9 of *HBComposer*, any *sidecar* file associated with the source image(s) is replicated and associated with the processed one. This ensures that any edits made to the source image are preserved.

The currently supported *sidecar* types are:

- **XMP** (.xmp): created by *Adobe Lightroom* when the flag for creating them is enabled in LR settings.
- **PHOS** (.phos): created by *Hasselblad Phocus* starting from version 4 when editing *3FR* RAW images instead of *FFFs*.

*NOTE: for frame-average and frame-median operations, the sidcar files of **the first image of the source sequence** are the ones that will be replicated and associated to the final processed image.*

5 Operations

5.1 Frame-Average

This operation will perform a composition of all Source images, producing a single RAW image which represents a pixel-by-pixel average of all processed images. The output file will take the name **FA.3FR**. If it already exists, it will be rewritten.

In a typical situation, you place your camera on a tripod and use the featured internal intervalometer to shoot a series of images of the same scene at a specified time cadence.

Figure 7 presents an example of the results that can be achieved with *frame-average*.

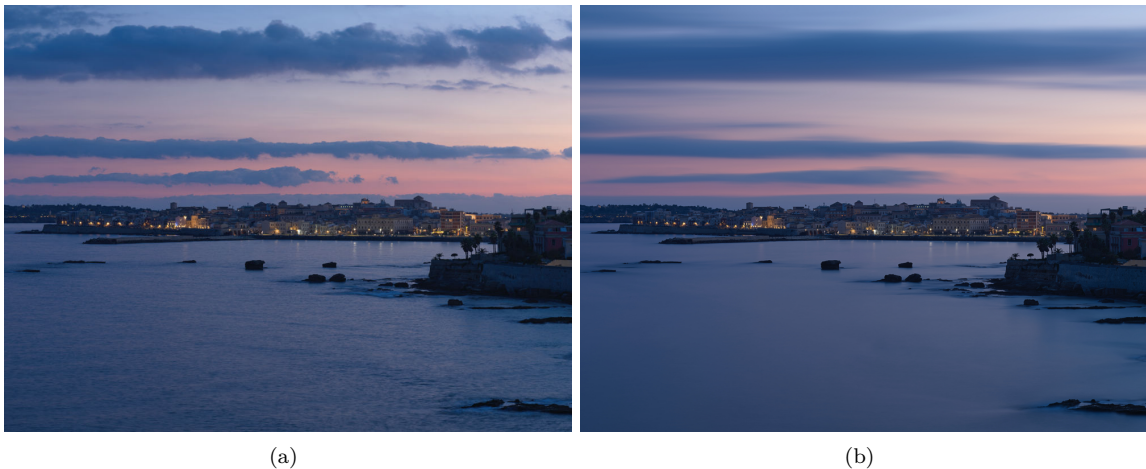


Figure 7: (a) Single frame (b) Frame-average of 160 RAW images

It shows the result of a frame-averaged composition of 160 frames, each one shot at ISO 64, f8 and 0.8s exposure time without any ND filter. It produced a RAW image which is roughly equivalent to a 2-minute-long exposure taken with an 8-stop ND filter on.

Frame-Average may be useful for the following purposes:

- **It reduces the overall image noise, especially in the shadows.**
Noise is reduced by the square root of the number of frames averaged. So, if you average 4 images, noise will be reduced to 1/2 of the noise you'd have in a single frame.
- **On moving subjects, it produces an effect similar to a long-exposure.**
An advantage to a real long-exposure is that you can operate in daylight without the need of an ND filter.
A disadvantage is that *Hasselblad* cameras do not currently allow shooting images at a speed greater than one frame every two seconds. Thus, **fast moving objects may not be represented as a continuous trail** in the final composed image. To try to partially compensate for this issue, a **frame weight shaping** function has been added to the tool (see below for more info on this).

Another tip to reduce gaps between frames to the minimum is **using a remote cable trigger with a lock button**, this will **avoid the 2-second gap** the camera intervalometer would introduce.

5.2 Frame-Median

This operation will perform a composition of all Source images, producing a single RAW image which represents a pixel-by-pixel median of all processed images. The output file will take the name ***FM.3FR***. If it already exists, it will be rewritten.

In a typical situation, you place your camera on a tripod and use the featured internal intervalometer to shoot a series of images of the same scene at a specified time cadence.

Figure 8 presents an example of the results that can be achieved with *frame-median*.



Figure 8: (a) Single frame (b) Frame-median of 34 RAW images

It shows the result of a frame-median composition of 34 frames, each one shot at ISO 64, f8 and 1/10s exposure time without any ND filter.

As shown in the sample image, on scenes containing moving objects like **cars, persons, etc...** *frame-median* may be useful to **cancel them out from the final image**. A crowded place where many people pass by can be turned into an empty space. This can be useful for example in architectural photography.

Like with *frame-average*, **frame weight shaping** could help in smoothing out some gaps, like on moving clouds for example, although it will increase the calculation time (see an example of weight shaping results below in the dedicated section).

Another tip to reduce gaps between frames to the minimum is **using a remote cable trigger with a lock button**, this will **avoid the 2-second gap** the camera intervalometer would introduce.

5.3 Frame Alignment (Beta)

Option *Align frames* only applies to *frame-average* and *frame-median* operations. When activated, it tries to align frames before they get composed. This may be useful if vibrations due to the wind or a flimsy tripod+head combo may have caused a slight misalignment of the captured frames.

Since the alignment occurs in the RAW domain, only displacements with an even number of pixels can be recovered. Currently, the algorithm can correct displacements of up to 32 pixels in the RAW domain.

Due to the RAW nature of the treated image, only horizontal and vertical shifts are treated. Fixing rotational displacements would break the RGGB alignment and would require pixel interpolation/reconstruction in the RAW domain, which could be more detrimental than beneficial. Better to use a sturdy tripod!

Please note that frame alignment will slow down the composition process. All displacements are calculated at the beginning of the composition process. So, you will notice a delay when the pre-processing progress bar is near the end.

NOTE: this feature is still in beta version.

5.4 Frame Weight Shaping

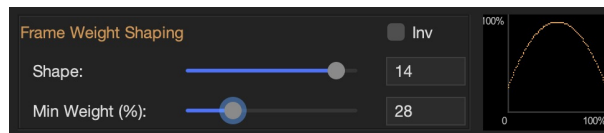


Figure 9: Frame weight shaping controls

You'll find this option in the top-right area of the application window. This feature shapes the weight of the frames in different ways while performing frame-average and frame-median. This feature may help in smoothing out some gaps that may arise in presence of moving objects in the scene when the time interval between captured frames is long.

5.4.0.1 Shape selects the curvature of the shaping function.

5.4.0.2 Weight Perc (%) selects the minimum weight value in percentage (maximum is always 100%).

5.4.0.3 Inv. inverts the shaping function.

The graph on the right shows a live preview of the shaping function while its parameters are changed.

In *Figure 10* you'll find a result comparison based on a *frame-median* composition of **100 images shot at a frequency of 1 every 10 seconds**. Without *weight shaping* (left), gaps in the clouds are visible in the upper and central right part of the image. *Weight shaping* succeeded in smoothing things out (right).



Figure 10: (a) Frame-median with no oversampling – gaps are present in the clouds (b) Frame-median with 16X oversampling – gaps in the clouds are smoothed out

5.5 Flat-Field Correction

HBComposer features a *flat-field* correction operation and offers an advanced set of fine-tune parameters.

This technique consists in taking an additional frame, called *flat-field* or *LCC*, while holding a thick piece of white semi-transparent plexiglas in front of the lens. This special image should be acquired under the very same conditions as the photo it will correct. Aperture, focus, shift/tilt levels, and light conditions need to match. Shutter duration can be adjusted to get a correct exposure. A good *LCC* exposure is key to the effectiveness of this technique.

From now on, the words *flat-field* and *LCC* will be used interchangeably to indicate this special frame.

Figure 11 shows a typical histogram of a well-exposed *flat-field* image. To minimize noise transferred to the processed image, it must be exposed to the right without clipping the highlights (ETTR).

Figure 12 shows a typical example of a flat-field image taken with a technical camera and a shifted (raised) lens.

Figure 13 explains how the *LCC* correction process works. The *flat-field* capture is applied to the

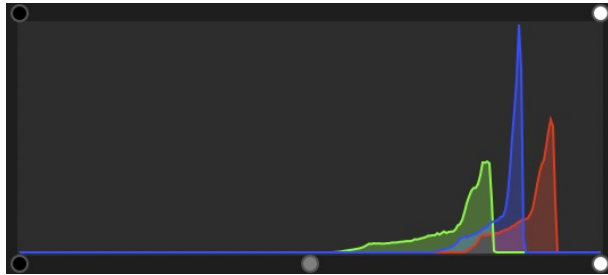


Figure 11: Histogram of a well-exposed flat-field (LCC) image



Figure 12: Example of Flat-Field (LCC) image

original image to normalize it.

5.5.1 Goals of flat-field correction

Flat-field correction allows the reduction and often the full elimination of the following image issues:

- **Color cast** when taking images with **shifted lenses**. The issue is particularly evident on non-BSI sensors when used in conjunction with symmetrical wide-angle and ultra-wide-angle lenses, such as the *Schneider-Kreuznach* technical ones, and some *Rodenstocks* as well.
- **Light Falloff** when taking images with **shifted lenses**. The issue is particularly evident if using symmetrical wide-angle and ultra-wide-angle lenses, such as the *Schneider-Kreuznach* technical ones, and some *Rodenstocks* as well.
- **PDAF sensor banding**, an issue that has been discovered by *GetDPI* forum member *@diggles* while using the most recent *Hasselblad CFV-100c* digital back together with a *Schneider-*



Figure 13: Flat-field application process

Kreuznach Apo-Digital 5.6/35 XL lens. It shows up when shifting this lens, especially on uniform areas like the sky. In *Figure 14* an example of PDAF banding fix performed by *HBComposer's flat-field* correction is presented.



Figure 14: (a) Original image - horizontal PDAF banding and light falloff present (b) Processed image - horizontal PDAF banding and light falloff corrected

You can follow the forum discussion under the following link (click on it):

<https://www.getdpi.com/forum/index.php?threads/hasselblad-100c-and-35xl.75781/>

- **Sensor tiling**, an issue which makes the different tiles a digital sensor is typically composed of evident in the final image. The issue is particularly evident on non-BSI sensors when used in conjunction with symmetrical wide-angle and ultra-wide-angle lenses, such as the *Schneider-Kreuznach* technical ones, and some *Rodenstocks* as well. In *Figure 15* an example of sensor tiling fix performed by *HBComposer's flat-field* correction is presented.
- **Dust spots** present on the sensor glass which are visible in the final image as darker circular or elongated shapes. In *Figure 16* an example of dust spot fix performed by *HBComposer's*



Figure 15: (a) Original image - sensor tiling, color cast and vignetting present (b) Processed image - sensor tiling, color cast and light falloff corrected

flat-field correction is presented.



Figure 16: (a) Original image - dust spot and light falloff present (b) Processed image - dust spot and light falloff corrected

IMPORTANT NOTE: unfortunately, in case of extreme light falloff or CA the flat-field algorithm will not be able to correct everything. High noise and residual color casts need to be corrected later on in the image processing chain, in the RAW developer or

in post-production.

5.5.2 Operations and flat-field correction

You can apply a *flat-field* correction to the following operations:

- **Frame-Average and Frame-Median**

In such cases, the *flat-field* correction is applied to the final composed image before saving it to disk.

Select the *LCC* file and adjust all the parameters before executing the composition.

- **Batch: Image + LCC or LCC + Image**

In this case, images in *Source Folder* need to be arranged in couples so that, when alphabetically sorted, each *LCC* frame will immediately follow (Image + LCC) or precede (LCC + Image) its target image.

The corrected images produced by *HBComposer* will keep the original name with the additional *__HBC* suffix.

- **Batch: Single LCC**

A single *LCC* is applied to all files present in *Source Folder*. In such case, you need to manually select the *LCC* image as you would do with *frame-average* and *frame-median*.

The corrected images produced by *HBComposer* will keep the original name with the additional *__HBC* suffix.

- **Single: Image + LCC**

A chosen *LCC* is applied to a chosen image. The corrected images produced by *HBComposer* will keep the original name with the additional *__HBC* suffix.

5.5.3 Flat-field correction settings

HBComposer offers a set of fine-tuning parameters for *flat-field* correction. The most useful combinations are provided as factory presets.

You don't need to learn all of the following parameters if you just want to use the tool. Just choose a factory preset and you're good to go. For exceptional cases though, you have the ability to tailor the parameters and save them as a custom preset.

Parameters are organized in different panels. Here below a brief description of each setting is given:

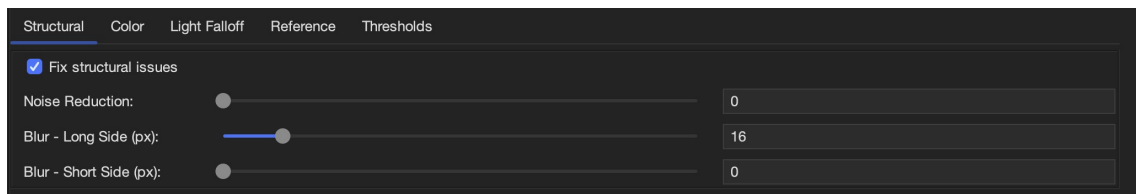


Figure 17: HBComposer - Flat-field structural settings (*Structural* panel)

5.5.3.1 Fix structural issues (Updated) Located in the *Structural* panel (see *Figure 17*). When enabled, it will fix structural issues like PDAF banding, dust spots, sensor tiling. This fix might introduce a bit of noise in the final image. You can leverage the *Noise Reduction* and/or the *Blur* controls to tame the noise.

5.5.3.2 Noise Reduction (Updated) Located in the *Structural* panel (see *Figure 17*). Applies a bilateral noise reduction to the final RAW image which is proportional to the level of light falloff recovery performed by the *flat-field* correction. This clever way of applying noise reduction will be more aggressive in the areas that need it most, while excluding the areas untouched by *LCC* correction.

This setting might help when enabling the structural issues fix (flag “*Fix structural issues*”) because this fix might introduce some noise in the processed image. A radius of 2 or 3 is usually enough for light to medium falloffs. A value larger than this may be necessary for extreme falloffs.

5.5.3.3 Blur (long / short side) (Updated) Located in the *Structural* panel (see *Figure 17*). Sets the pixel radius of *LCC* blur along each side of the frame. The blur is applied to the high-frequency LCC used to correct structural issues. Applying it helps reduce the noise introduced by this type of correction, but it will reduce the effectiveness of the correction. A clever use might help a lot. For example, for PDAF banding a bit of blur along the same direction of the bands will still be effective in correcting the banding while reducing the noise in the processed image.

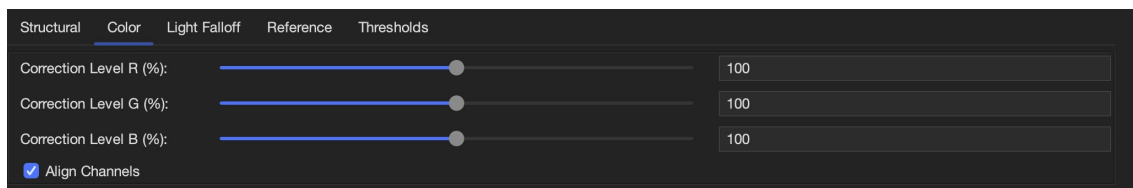


Figure 18: HBComposer - Flat-field color cast correction settings (*Color* panel)

5.5.3.4 Correction Levels (R, G, B) (%) (Updated) Located in the *Color* panel (see *Figure 18*). Sets the percentage of *LCC* color cast correction to be applied on each color channel. Normally it should be kept at 100%. You may increase it together with the other channels to over-correct, or decrease to under-correct. If the three sliders are disjoined and set differently from each other, this correction can help in fine-tuning any global color cast, especially if the chosen reference point is not perfectly neutral (see below). If unsure, leave the three levels aligned and at their default value (100%).

5.5.3.5 Light Falloff (%) (New) Located in the *Light Falloff* panel (see *Figure 19*). Sets the percentage of *LCC* light falloff correction to be applied. Normally it should be kept at 100%. You may lower it to under-correct. This is often done to keep a bit of vignetting around the subject. If



Figure 19: HBComposer - Flat-field light falloff correction settings (*Light Falloff* panel)

unsure, leave it at its default value (100%).

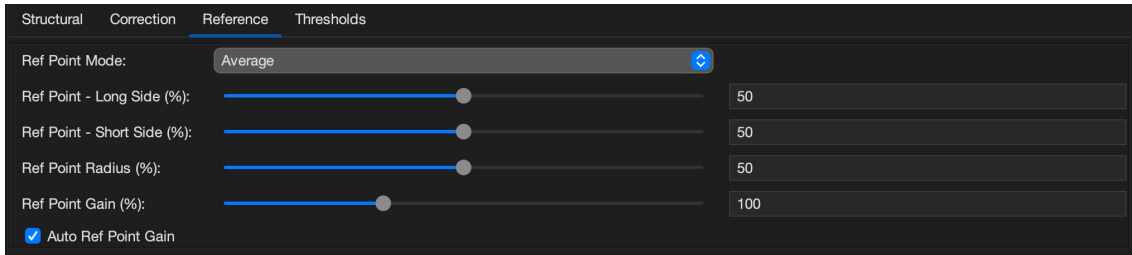


Figure 20: HBComposer - Flat-field correction settings (*Reference* panel)

5.5.3.6 Ref Point Mode Located in the *Reference* panel (see *Figure 20*). The *flat-field* correction algorithm needs a reference value taken from the *LCC* image which will be used to normalize the final image in terms of colors and brightness. It sets the values (one for each R_GB channel) where the amplification factor is considered to be 1. To over-simplify the matter, you can consider it as the white balance point. This setting allows you to choose different ways of calculating it:

- **Average**

The average of the full *LCC* image is used.

In presence of high light falloff, it may produce a dark image, which you can try to counterbalance by using the *Ref Point Gain* setting (see below).

When a high color cast is present in the *LCC* frame, the full average may leave some residual global color cast in the final image. In this eventuality, you can disjoin the *Correction Levels* (*R*, *G*, *B*) % sliders and try to counterbalance the color cast.

If some (under) clipping in the final image happens (usually pink areas), try to increase the *Ref Point Gain* (see below).

In presence of any of the above problematic cases however it is advisable to switch to *Custom Selection* mode and manually choose a good neutral *LCC* point/area.

This is the default selection.

- **Custom Selection**

Offers the option of selecting a specific point of the *LCC* (*Ref Point Radius* = 0) or a circular averaged area (*Ref Point Radius* is > 0). This kind of selection works best in cases where a significant degradation in a zone of the *LCC* image is present. Just place the reference point/area so that it keeps the degraded zone excluded (see *Ref Point (long/short side) (%)* and *Ref Point Radius (%)* below).

5.5.3.7 Ref Point (long / short side) (%) Located in the *Reference* panel (see Figure 20). These sliders are active when *LCC Custom Selection* mode is selected. They help position the reference point along the X-Y plane in percentage values, as shown in Figure 21.

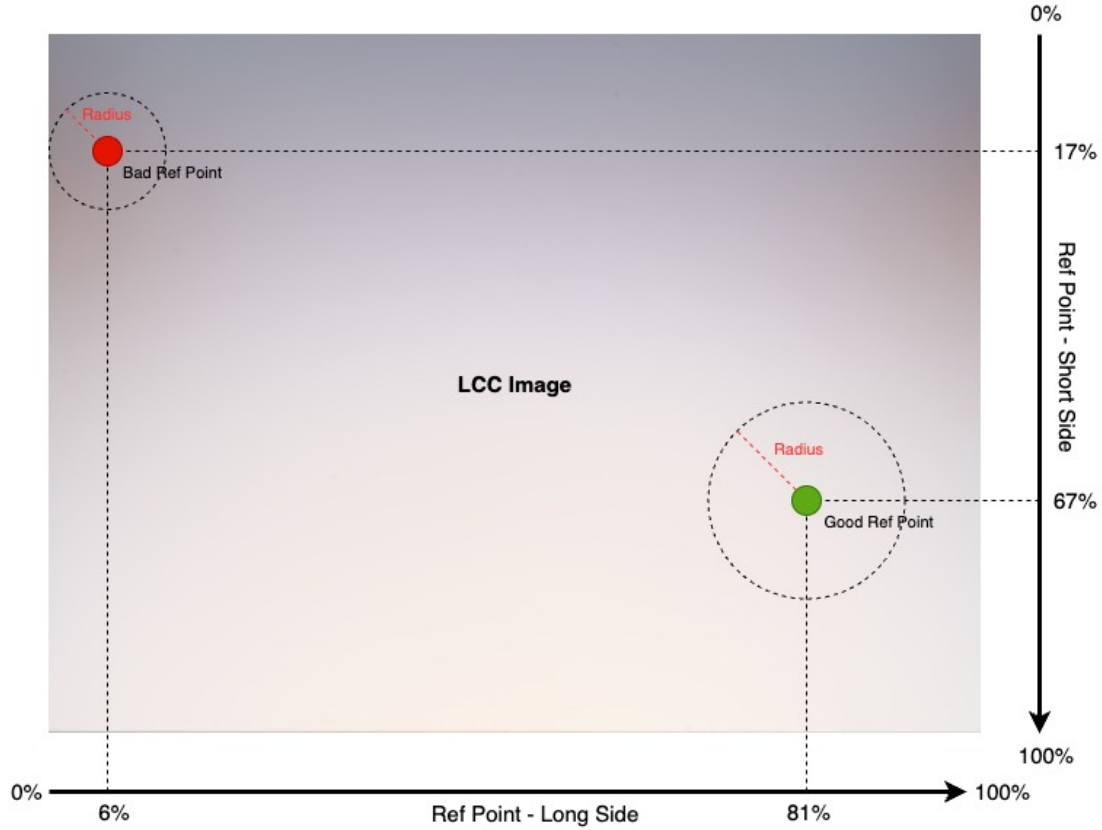


Figure 21: A scheme indicating how to use the Ref Point sliders to place the LCC reference point.

If unsure, leave the *Ref Points* at their default values (50%, 50%). This places the reference point at the very center of the *LCC* frame, which works well on average use-cases that usually only have a slight amount of peripheral aberrations.

NOTE: unfortunately, the tool does not feature a live preview of the reference point/area placement nor of the LCC correction results. Multiple attempts may be needed to nail the right spot.

5.5.3.8 Ref Point Radius (%) Located in the *Reference* panel (see Figure 20). When *Ref Point Mode* is set to *Custom Selection*, *HBComposer* offers the possibility of increasing the area around the chosen reference point. The area will cover a circle of a specific radius (percentage) around it. A radius of 0% corresponds to the single reference point. The maximum radius (100%) corresponds to a circle around the reference point having a radius of half the diagonal of the *flat-field* image. Values within the area will be averaged.

If unsure, leave the radius at its default value (50%).

5.5.3.9 Ref Point Gain (%) Located in the *Reference* panel (see *Figure 20*). In some situations, usually in presence of a highly degraded *flat-field* image, and especially when including severely vignetted areas in the *LCC* reference area, the produced image may turn out dark. Increasing **Ref Point Gain (%)** may help bring the produced image to the right exposure level. This is particularly important **in presence of clipped highlights in the source image**. Such areas **must remain clipped** also in the processed image, otherwise they will turn pink due to the absence of color information. If you encounter a similar situation, either manually increase this parameter until highlights no longer show a pink cast in the processed image (even when lowering down exposure in RAW developers) or enable the **Auto Ref Point Gain** (see below) to let *HBComposer* automatically set the gain level for you. If unsure, leave it at its default value (100%) and enable **Auto Ref Point Gain**.

5.5.3.10 Auto Ref Point Gain Located in the *Reference* panel (see *Figure 20*). When on, this flag automatically calculates the **Ref Point Gain (%)** value described above so that the top image highlight value matches the one of the original image. **This is very important in presence of source images with clipped highlights**, as such highlights must stay clipped in the processed image. If they don't, they will get a pink cast due to the loss of color information. After processing the image, the gain calculated automatically will be reported to the **Ref Point Gain (%)** slider, so that it can be fine-tuned manually if necessary. As long as this flag remains on, the value reported on the slider is ignored. Any manual intervention to the slider will automatically disable the flag. This flag is active by default, as it is also useful for images without clipped highlights because it keeps them at almost the same exposure of the original image.

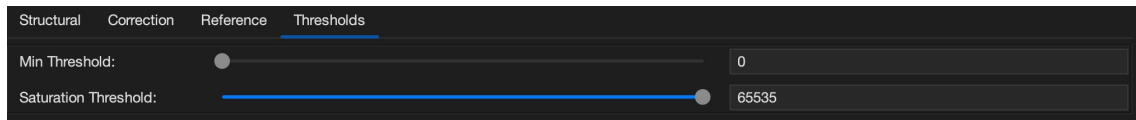


Figure 22: HBComposer - Flat-field correction settings (*Thresholds* panel)

5.5.3.11 Min Threshold Located in the *Thresholds* panel (see *Figure 22*). Sets the minimum threshold value to consider a pixel of the *LCC* image valid for correction (0-65535). A value below this threshold leaves the matching pixel of the final image uncorrected. Normally, *Hasselblad* RAW files have a black level set to 4096, so anything below that would perform a full correction. You may need to increase it above 4096 if the sensor has some defective pixels that would create problems in the final image. This is really rare though. If unsure, leave it at its default value (0).

5.5.3.12 Saturation Threshold Located in the *Thresholds* panel (see *Figure 22*). Sets the maximum threshold value to consider a pixel of the *LCC* image valid for correction (0-65535). A value above this threshold leaves the matching pixel of the final image uncorrected. Normally *Hasselblad* RAW files have a maximum pixel level set to 65535. You may need to decrease this setting below that if the sensor has some defective pixels that would create problems in the final image. This is really rare though. If unsure, leave it at its default value (65535).

5.5.4 Flat-Field (LCC) Presets

Starting from version 1.5.0, *HBComposer* supports LCC presets.

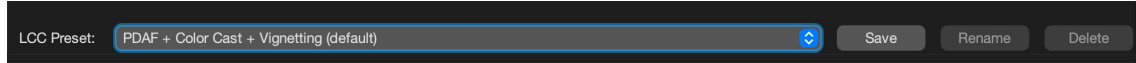


Figure 23: HBComposer - flat-field correction presets

In order not to lose your latest settings, when this new version is opened for the first time, it will save them as a custom user profile named *Custom LCC Preset 1* and select it as current profile.

From that moment on, you can create (save), rename, and delete profiles to your liking.

HBComposer comes with a predefined set of factory profiles which cover the most important use-cases. Here below is the list of the factory presets with a brief explanation:

5.5.4.1 PDAF + Color Cast + Dust + Tiling (best correction, noisy)

Noise Reduction: 0 | Blur Long Side: 0 | Blur Short Side: 0
Corrections: color cast, light falloff, PDAF banding, sensor tiling, dust spots

This is the most effective preset for correcting all issues. Since the LCC image is not blurred at all, this setting will introduce a bit of noise to the processed image(s).

5.5.4.2 PDAF + Color Cast + Dust + Tiling (with noise reduction, softer)

Noise Reduction: 2 | Blur Long Side: 0 | Blur Short Side: 0
Corrections: color cast, light falloff, PDAF banding, sensor tiling, dust spots

Like the preceding one, but with a bit of noise reduction applied to the processed image. This reduces the noise introduced by the application of the unblurred LCC. The noise reduction is applied proportionally to the light falloff magnitude, it will be stronger on the areas that receive the biggest exposure compensation. There is no such thing as a free lunch though: the noise reduction will introduce a slight loss of micro-details.

5.5.4.3 PDAF + Color Cast + Vignetting (default)

Noise Reduction: 0 | Blur Long Side: 16 | Blur Short Side: 0
Corrections: color cast, light falloff, PDAF banding

It applies an LCC blurring along the long side, the direction of the PDAF banding. This is still very effective for correcting PDAF banding while minimizing the noise introduced to the processed image. Results on dust spots and sensor tiling may vary. This preset represents the **sweet spot** for correcting PDAF banding together with color cast and light falloff.

5.5.4.4 Color Cast + Vignetting

Noise Reduction: 0 | Blur Long Side: 64 | Blur Short Side: 64
Corrections: color cast, light falloff

Applies a blurred LCC image just like any other RAW developer would do. It corrects color cast and light falloff, does not introduce noise, but it is not effective against PDAF banding, sensor tiling and dust spots.

5.5.4.5 Only Structural Issues

Noise Reduction: 0 | Blur Long Side: 64 | Blur Short Side: 64
Corrections: PDAF banding, sensor tiling, dust spots

This preset activates a special algorithm which only corrects structural issues but does not correct color cast and light falloff. The LCC must be re-applied in your preferred RAW developer to correct them as one would normally do without *HBComposer*. The idea behind this preset is that commercial RAW developers may do a better job and may have more options than *HBComposer* in doing the usual LCC correction. So, *HBComposer* can only do what they won't: correct the structural issues. LCC blur is set to 64px in this case, but it has another meaning/behavior in this special case (see *paragraph 5.5.3.1* and *paragraph 5.5.3.3*).

NOTE: Factory presets cannot be renamed nor removed.

6 HBComposer Plugin for Adobe Lightroom®

Starting from *version 1.5.2*, *HBComposer* comes with a plugin companion for Adobe Lightroom Classic® which supports the following actions:

- Frame-Average
- Frame-Median
- Flat-field application on *Image* -> *LCC* sequence
- Flat-field application on *LCC* -> *Image* sequence

To perform such actions, the plugin communicates with the main *HBComposer* application, but there is no need to keep *HBComposer* running. The plugin will invoke it behind the scenes when necessary.

6.1 Plugin installation

The plugin is installed via *HBComposer*:

1. Launch *HBComposer*.
2. Open the *Help* menu and select *Install Lightroom Plugin...* (see *Figure 24*).
3. The plugin will be automatically installed. If an older version is already present, it will be updated. A dialog box will confirm the installation/update and will indicate the folder where it has been installed (see *Figure 25*).
4. To complete the installation, a *Lightroom* restart is needed if it was already running while installing the plugin.

6.2 Plugin configuration

You can enable, disable and configure the plugin in *Lightroom* by going to *File* -> *Plug-in Manager...* as indicated in *Figure 26* and *Figure 27*.

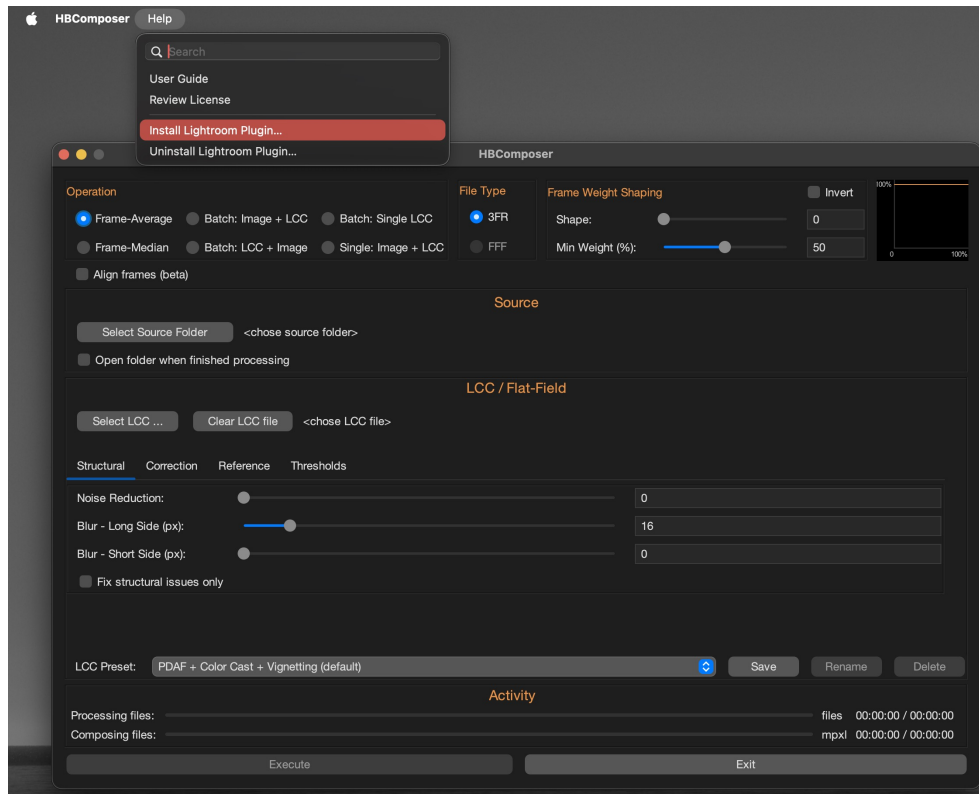


Figure 24: HBComposer - Install Lightroom Plugin



Figure 25: HBComposer - Plugin Installation Result

6.2.1 Configuration parameters

6.2.1.1 Executable path

This parameter points to the installed HBComposer application. There is no need to change this

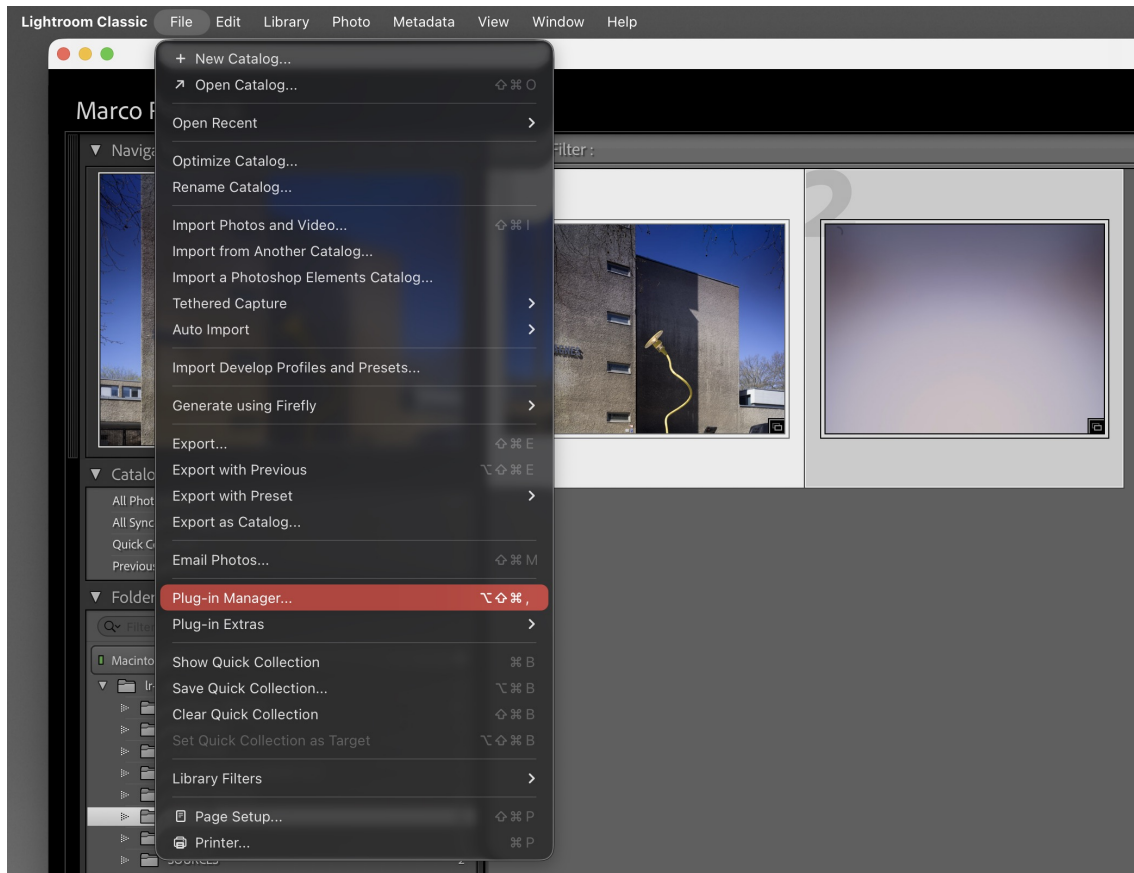


Figure 26: HBComposer - Plug-in Manager menu item

value, unless your *HBComposer* has been relocated elsewhere.

6.2.1.2 LCC preset

A drop-down list containing all LCC presets defined in *HBComposer*, including your custom ones. The selected preset will be the one used for the plugin's LCC actions. More on this in *subsection 5.5.4*.

6.2.1.3 Align frames

This checkbox enables or disables the automatic frame alignment for the frame-average and frame-median plugin actions. More on this in *subsection 5.3*.

6.3 Plugin execution

In Lightroom, the plugin actions can be accessed via menu *Library -> Plug-in Extras* as shown in *Figure 28*.

Before executing any action, the files that the action applies to must be selected from the Lr Library

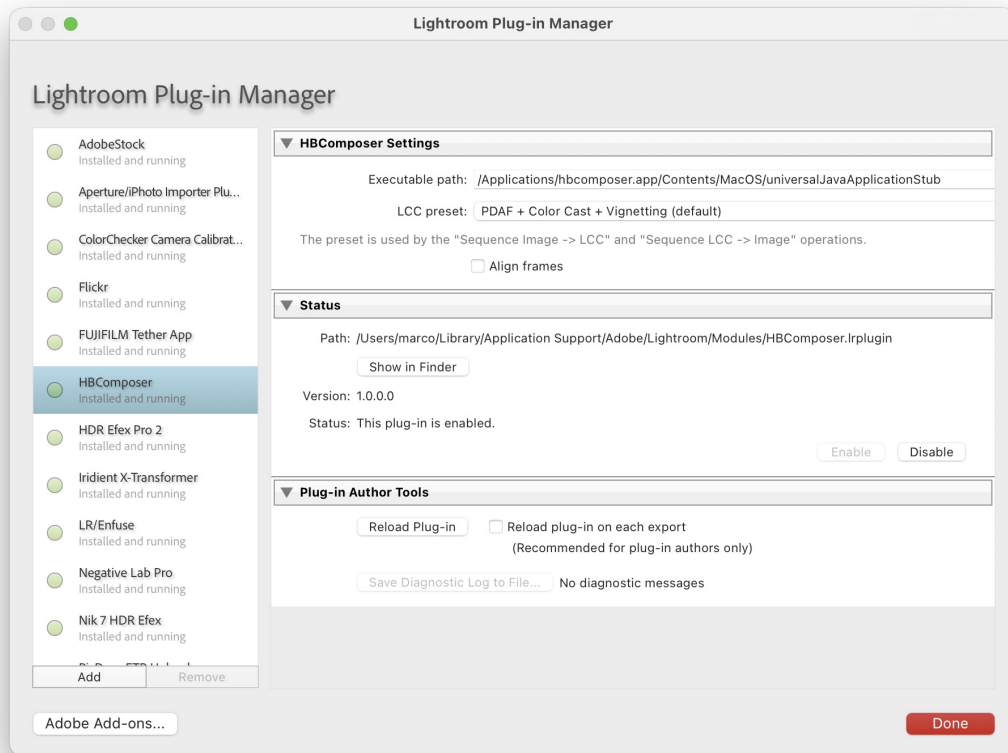


Figure 27: HBComposer - Plug-in configuration window

Grid.

Below is a brief description of the actions provided.

- **Average:** a frame-average of the selected images will be produced. If the configuration checkbox *Align frames...* is enabled, an automatic frame pre-alignment is performed before composing the files. See more in *subsection 5.1*.
- **Median:** a frame-median of the selected images will be produced. If the configuration checkbox *Align frames...* is enabled, an automatic frame pre-alignment is performed before composing the files. See more in *subsection 5.2*.
- **Image -> LCC:** For each image, the LCC **immediately following** it in the Lr grid is applied. Therefore, the number of selected images must be even and the images must be organized in the grid so that each LCC frame **follows** its matching image. More on this in *subsubsection 5.5.2*.
- **LCC -> Image:** For each image, the LCC **immediately preceding** it in the Lr grid is applied. Therefore, the number of selected images must be even and the images must be

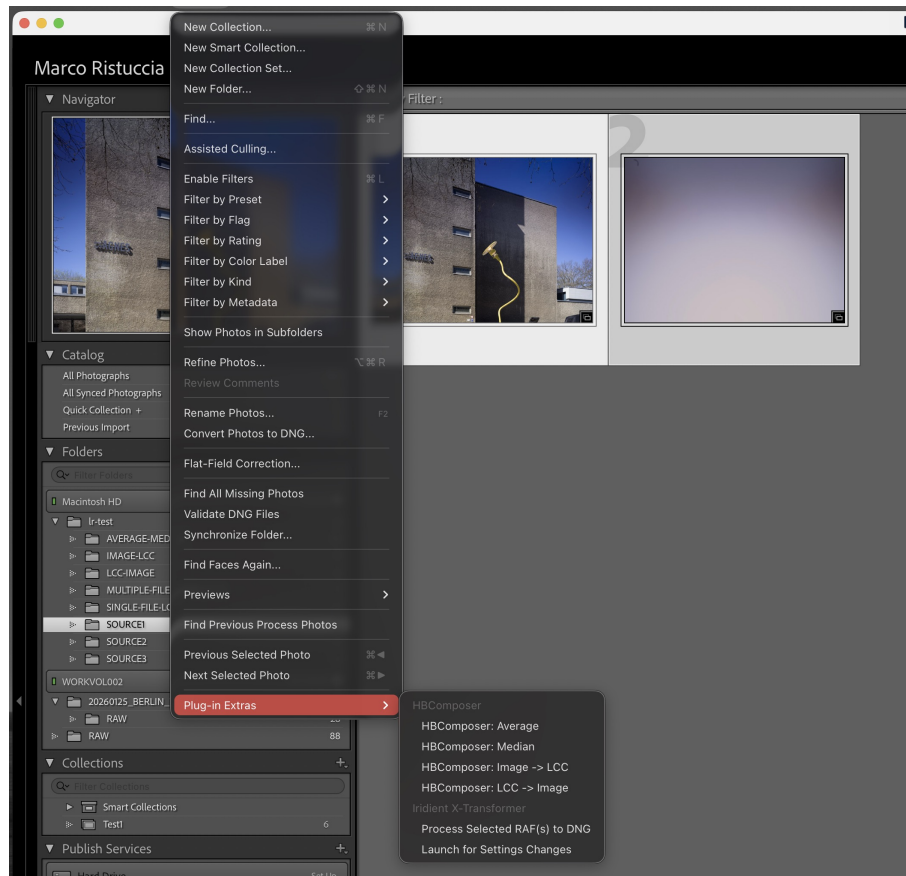


Figure 28: HBComposer - Plugin actions

organized in the grid so that each LCC frame **precedes** its matching image. More on this in *subsubsection 5.5.2*.

Some caveats:

- The plugin works on both real folders and collections. The processed images appear automatically. If they don't, an old one is probably already there. In such case, a folder sync in Lr followed by an inspection and possibly a removal of the old file may be needed before running the action again.
- When working in a collection with *Average* and *Median* actions, the produced image (*FA.3FR* / *FM.3FR*) will be physically placed in the same folder as the first image in the selected series.
- When working in a collection with the *Image -> LCC* action, each corrected image produced (*..._HBC.3FR*) will be physically placed in the same folder as the corresponding original image.
- When working in a collection with the *LCC -> Image* action, each corrected image produced

(..._HBC.3FR) will be physically placed in the same folder as the corresponding LCC image.

6.4 Plugin uninstallation

The plugin is uninstalled via *HBComposer*:

1. Launch *HBComposer*.
2. Open the *Help* menu and select *Uninstall Lightroom Plugin...* (see *Figure 29*).

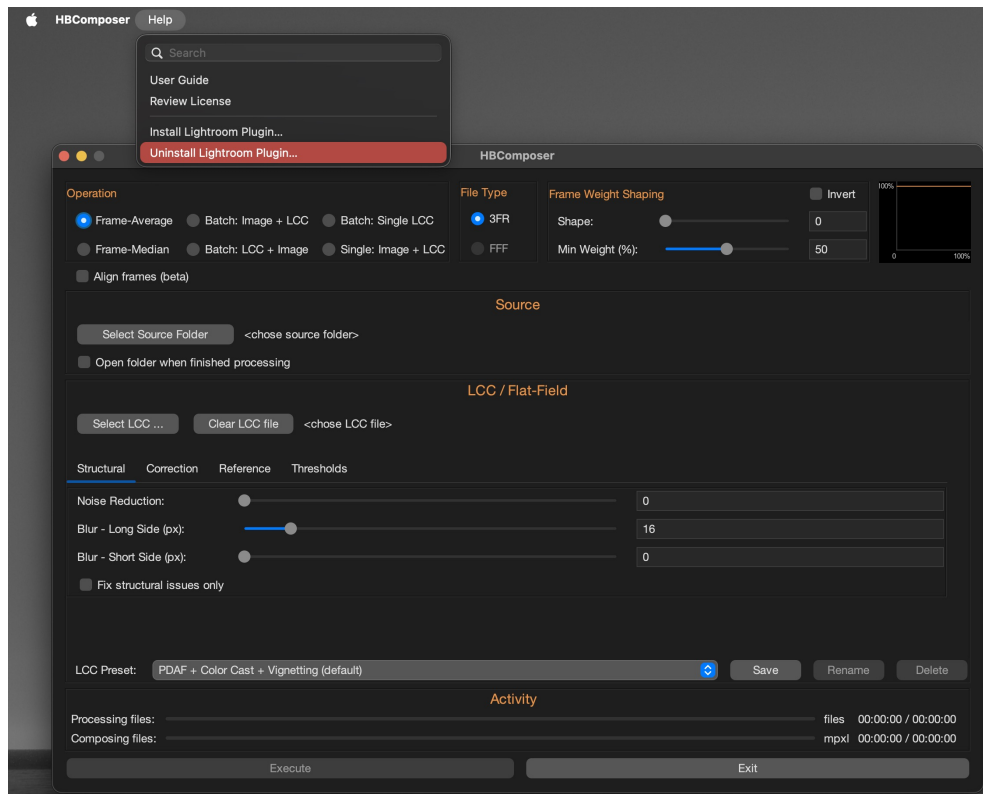


Figure 29: HBComposer - Uninstall Lightroom plugin

3. After a confirmation request, the plugin will be automatically removed. A dialog box will confirm the removal (see *Figure 30*).
4. To complete the removal, a *Lightroom* restart is needed if it was already running while uninstalling the plugin.

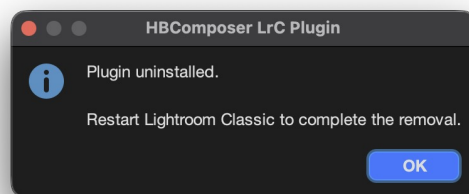


Figure 30: HBComposer - Plugin uninstallation result