darktable 1.2

darktable 1.2

Table of Contents

Preface to this manual

User manual version and applicable darktable version are listed below:

	version	date
user manual	1.2.0	March 2013
darktable	1.2	March 2013

Many thanks to all contributors to this user manual. Special thanks for proof reading, style improvement, and constructive criticism go to Colin Adams, Mark Garrow, Simon Harhues, Ammon Riley, Rob Z. Smith, and David Vincent-Jones.

Chapter 1. Overview

darktable is an open source photography workflow application and RAW developer, a virtual lighttable and darkroom for photographers.

It manages your digital negatives in a database, lets you view them through a zoomable lighttable and enables you to develop raw images and enhance them.

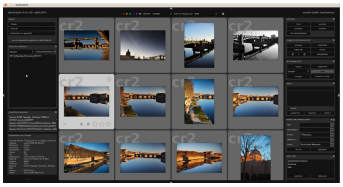
General Features

- darktable runs on GNU/Linux / GNOME, Mac OS X / macports and Solaris 11 / GNOME.
- Fully non-destructive editing.
- All darktable core functions operate on 4x32-bit floating point pixel buffers for high accuracy processing, preventing banding and color breaks.
- darktable makes heavy use of *Streaming SIMD Extensions 2* (SSE2) instructions of the CPU to speed up processing. In fact, darktable will only run on a CPU that supports SSE2.
- GPU acceleration via OpenCL (runtime detection and enabling).
- Most image processing is done in CIELab color space, which is much larger than the gamut of modern displays, printers or even human vision.
- Full color managed display with softproofing and gamut-check. Built-in ICC profile support for export: sRGB, Adobe RGB, XYZ and linear RGB.
- A collect module allows you to execute flexible database queries, search your images by tags, image rating (stars), color labels and many more. Filtering and sorting your collections within the base query or simple tagging by related tags are useful tools in your every-day photo workflow.
- Import a variety of standard, raw and high dynamic range image formats (e.g. JPG, CR2, OpenEXR, PFM, ...).
- darktable has a zero-latency fullscreen, zoomable user interface through multi-level software caches.
- Tethered shooting.
- The powerful export system supports Picasa webalbum, flickr upload, disk storage, 1:1 copy, email attachments and can generate a simple html-based web gallery. darktable allows you to export to low dynamic range (JPEG, JPEG2000, PNG, TIFF), 16-bit (PPM, TIFF), or linear high dynamic range (PFM, EXR) images.
- darktable uses both XMP sidecar files as well as its fast database for saving metadata and processing settings. All Exif data is read and written using libexiv2.
- darktable comes with 50 image operation modules which cover everything from basic operations, tonal value changes, color manipulation, correction of common image defects to artistic effects.
- Many darktable modules can be combined with blending operators for even more development options.
- A powerful blend mask feature gives you fine control on module's effect to different parts of an image steered by pixel values.

- Most modules can exist as multiple instances. Together with the blend mask feature, you can let an operation have different effects on different parts of the image.
- darktable introduces a highly efficient, yet simple “single-click” denoiser that always just works (tm). It's designed as a module where the denoising performance only depends on camera and ISO setting. A database of profiles contains parameters for well over 70 popular camera models.

1.1. User interface

This section describes the layout of the user interface.



1.1.1. Views

darktable consists of several views or modes. There are four available views as described in this section. You can switch between views by clicking the view name at the top of the right panel - the active view is highlighted - or by using one of the key accelerators:

l	switches to lighttable
d	switches to darkroom
t	switches to camera tethering
m	switches to map

1.1.1.1. Lighttable

The lighttable view is where images and filmrolls are managed. It's also where you rate images, add tags and colorlabels, and export images among other actions (see Chapter 2, *Lighttable*).

1.1.1.2. Darkroom

In the darkroom view you develop a single image using the available modules (see Chapter 3, *Darkroom*).

1.1.1.3. Tethering

This view is for shooting with the camera connected to the computer and remotely capturing images that will be downloaded and shown on computer screen (see Chapter 4, *Tethering*).

1.1.1.4. Map

This view shows images with geo-tag data on a map and allows manually geo-tagging new images (see Chapter 5, *Map*).

1.1.2. Screen layout

The general screen layout of all views is similar. There is a center area which contains most of the relevant information of that view. Then there are panels left, right, top and bottom to the center area. The left panel typically has an informational purpose. The right panel offers functions to modify an image. The top and bottom panel give access to several settings and shortcuts. Each of the panels can be collapsed or expaned by pressing a triangle like  , located close to the panel.

By pressing the *TAB* key all panels get collapsed, allowing the center area to occupy all available space. Pressing *TAB* again brings you back to the previous view.

Fullscreen view can be toggled by pressing *F11*.

darktable's contrast can be changed by using *F7* and *F8* and darktable's lightness by using *F9* and *F10*.

1.1.3. Filmstrip

The filmstrip along the bottom shows the same images as lighttable, with respect to filters and sort order. It is turned on/off with key accelerator *ctrl-f*. You can navigate along the filmstrip by scrolling with the mouse wheel. The filmstrip allows you to interact with images while you are not in lighttable mode. For example, you can, while developing an image in darkroom mode, switch to another image to develop, by double clicking the thumbnail in the filmstrip. You can also rate the images as you do in lighttable, copy/paste history stack, etc.

1.1.4. Preferences

A button located in the upper panel allows you to define various parameters which control darktable's behavior.

The options are fairly self-explanatory. If you need more information, hover the mouse cursor over the text label or entry box, to display a popup tool-tip. All configuration parameters are explained in Chapter 6, *Preferences and settings*.

1.2. darktable basic workflow

1.2.1. Importing images

To begin with darktable, you first need to import images. The import module is in the left pane of the lighttable view (Section 2.2.1, “Import”). You can either import from the filesystem or, if darktable supports your camera model, directly from camera.

1.2.1.1. Importing images from filesystem

When importing from disk, you can import either a single image or a folder. darktable will analyse its content, detect images that are already imported and only import new images.

1.2.1.2. Importing from camera

Connect your camera to your system. If your distribution tries to automount it, select the option to abort the mount operation. Otherwise the camera will be locked and not accessible from within darktable. If you don't see your camera in the import pane, hit the “scan for devices” button. Your camera will then appear in the same pane with additional choices: *import* and *tethering*.

1.2.2. Basic development steps

1.2.2.1. Introduction

This section will guide you through the basics of developing an image in the darkroom view.

To begin, open an image in darkroom mode by double clicking an image thumbnail on the lighttable. The darkroom mode is where the actual adjustments for an image are made, where an arsenal of modules are at hand to help you reach your goal.

Each change made on a module while developing an image is turned into a *history stack* item. The history is stored in a database and in an XMP sidecar file for the specific image. Changes are stored automatically without the need to press a "save" button, therefore you can safely leave darkroom mode or darktable, and come back later to continue your work.

On the left panel in darkroom mode is the *history stack*, showing changes starting from bottom, and building up with each change made to the image. You can select a point in this history to show how the image looked at that point, for comparison of changes. The stack can be compressed: it will be optimized and redundant changes will be discarded. When you think you are done and are happy with what you have done, just compress the history stack.

darktable ships with a number of modules, arranged into groups. These module groups are accessed via toggle buttons in the right panel, just under the histogram. There are also two special module groups named “active” and “favorites”, which only show modules enabled in the history for the current image, and modules selected as a favorite, respectively. Marking a module as a favorite is done in the *more modules* dialog (Section 3.3.7, “More modules”), at the bottom of the right panel, by clicking a module until a star is displayed in front of the icon.

1.2.2.2. White balance

The *white balance* module controls the white balance or color temperature of the image. It's always enabled and reads its default values from camera metadata embedded in the

image. The most common change is fine-tuning the white balance, which is done using the “temperature in” slider. Moving this slider left will make the color balance cooler, and moving it right will make it warmer.

1.2.2.3. Exposure correction

The *exposure* module is probably the most basic module of them all. Exposure is fine-tuned either by using the slider, or by dragging with the mouse in the *histogram*. You can also boost the black level to enhance contrast; but be careful: use small amounts, like steps of 0.005. There is also an auto-correct feature.

1.2.2.4. Noise reduction

The best starting point for noise reduction is *profiled denoise*. This module offers an almost “single-click” solution to fight noise. From a user perspective the effect only depends on camera type and ISO value, both derived from EXIF data. All other settings are taken from a database of noise profiles that the darktable team has collected - now covering already over 70 popular camera models. In addition you have several other options in darktable to reduce noise. There is *raw denoise*, *denoising based on bilateral filter*, *denoising based on non-local means*, and *equalizer*, which is based on wavelets. If your camera is not yet supported by *profiled denoise*, *denoising based on non-local means* is probably the most convenient, as it allows you to treat color and luminance noise separately.

1.2.2.5. Fixing spots

Sometimes you will need to remove spots caused by sensor dirt. The *spot removal* module is at hand and can also correct other disturbing elements like skin blemishes. If your camera has stuck pixels or tends to produce hot pixels at high ISO values, or longer exposure times, have a look at the *hot pixels* module for automatic correction.

1.2.2.6. Geometrical corrections

Quite frequently you want to only show part of the captured scene in your image, e.g. to take away some disturbing feature close to the frame. In other cases, the horizon in the image may need levelling, or there are perspective distortions. All this can be corrected in the *crop and rotate* module. If you need to correct typical camera lens flaws like cushion distortion, transversal chromatic aberrations or vignetting, there is a *lens correction* module.

1.2.2.7. Bringing back detail

Digital RAW images often contain more information than you can see at first sight. Especially in the shadows of an image, there are lots of hidden details. The *shadows and highlights* module helps bring these details back into visible tonal values. Structural details in fully blown-out highlights, by nature of the digital sensor, can not be recovered. However, you can correct unfavorable color casts in these areas with the *highlight reconstruction* module.

1.2.2.8. Adjusting global contrast

Almost each workflow will cover as one basic element the adjustment of global contrast. You will want to control how the image tonal values cover the available range of your output medium. darktable offers several alternative modules to take care of that. In one of them, the *tone curve* module, tonal values are adjusted by constructing a gradient curve. The *levels* module offers a much simpler interface, with just three sliders. In addition, there is a *zone system* module which allows control over tonal values by zones, inspired by the work of Ansel Adams.

1.2.2.9. Enhancing local contrast

Local contrast enhancement can emphasize detail and clarity in your image. Carefully used, it can give your photograph the right pop. darktable offers several modules for this task. The *local contrast* module is easy to handle, with just a few parameters. A much more versatile, but also more complex technique, is offered by the *equalizer* module. Have a look at its presets, to get a feeling for how it works. Equalizer is darktable's "Swiss Army Knife" for many adjustments where spatial dimension plays a role.

1.2.2.10. Color adjustments

darktable offers many modules for adjusting colors in an image. A very straightforward technique is implemented in the *color correction* module. Use it to give an image an overall tint or to adjust overall color saturation. The *color zones* module offers a much finer control to adjust saturation, or lightness, and even hue, in user defined zones. darktable's *tone curve* module - in addition to the classical adjustment of tonal values - gives you fine control over the colors in an image. Finally, if you intend to convert an image into black & white, a good starting point, with an easy to use and intuitive user interface, is offered by the *monochrome* module. Alternatively, you might consider using darktable's *channel mixer*.

1.2.2.11. Sharpening

If you start your workflow from a RAW image, you will need to have your final output sharpened. The *sharpen* module can do this with the classical USM (unsharp mask) approach, available in most image processing software. Another very versatile way to enhance edges in an image is offered by the *highpass* module, in combination with darktable's rich set of blending operators.

1.2.3. Exporting images

Changes to an image are not saved as in a regular image editor. darktable is a non-destructive editor, which means all changes are stored in a database, and the original image is untouched. Therefore, you need to export images to bake the processing options into an output file that can be distributed outside of darktable.

Images are exported from the lighttable view, using the *export selected* dialog in the right panel (Section 2.2.13, “Export selected”). In general, export means: save my developed RAW image as a JPEG.

The export is modularized into *storage* and *format*. darktable ships with several storage modules such as *save on disk*, *picasa* and *flickr webalbum* and more. Format modules are the actual image formats such as JPEG, PNG, TIFF, OpenEXR and more.

Select images on the lighttable, choose the target storage and format, and set the maximum width and height image restraints. This means that none of the images will be bigger than any of the width/height restraints and hit the export button. Leave the width and height restraints at zero, if you want the original resolution.

Chapter 2. Lighttable

The lighttable is where you manage all your images, ratings, export and much more.

2.1. Overview

2.1.1. Lighttable

In the central view, your images are shown as thumbnails, surrounded by a frame. When the mouse is over an image, its rating and color labels are shown in the frame, along with an indicator of whether the image has already been altered in darkroom. Also, when the mouse hovers over an image frame, image information (EXIF data, metadata) is shown in the *image information* panel in the bottom left.

While the mouse is over an image frame, there are a number of actions you can perform on the image. Here is a table of keyboard shortcuts and assigned actions.

0 – 5	set the rating of the image; if an image has 1 star and you hit the *1* key, the image will be unrated. Pressing *r* rejects the image.
F1 – F5	set a color label
ctrl-c	copy the history stack
ctrl-v	paste the copied history stack
d	open in darkroom view for developing
z	fully zoom into the image while the key is pressed

At the bottom you have an option to choose between zoomable lighttable view or filemanager view of the thumbnails. In zoomable lighttable view, scroll with your mouse wheel to zoom in and out. Moving the mouse while *pressing the left mouse button* allows you to navigate through your collection. In filemanager view, you can can change the number of images in each row, using the slider next to the filemanager option, or by using *ctrl-(mouse wheel)*. Use your mouse wheel to navigate through your collection.

While in filemanager mode, you can scroll (not select) up and down through your collection using $\uparrow/\downarrow$. In zoomable lighttable $\leftarrow/\rightarrow/\uparrow/\downarrow$ allow you to move left/right/up/down through your collection. Pressing *g* goes to the top, *shift-g* to the bottom.

To locate where you are in a collection, there are indicators at the extreme borders of the window: left/right for your position when you are in filemanager mode, left/right and top/bottom for your vertical and your horizontal position, respectively, when you are in zoomable lighttable view.

2.1.2. Filtering and sort order

The filtering and sort order of images in the lighttable are accessed and changed in the top bar. You can select among a few predefined filters, and ordering like “date”, “color labels” or “rating”.

2.1.3. Image grouping

Grouping images helps improve structure and clarity of your image collection when displayed in lighttable view.

You can combine images into a group by selecting them, and clicking the “group” button in the *selected image(s)* panel (Section 2.2.7, “Selected image(s)”), or by typing *ctrl-g*. Likewise, you can remove selected images from a group by clicking the “ungroup” button, or typing *shift-ctrl-g*. Images generated by duplicating an existing one, are automatically grouped. If you import images from the file system or camera, images with the same base name, but different extensions (eg. IMG_1234.CR2 and IMG_1234.JPG), will form a group. Images which are members of a group are labeled with a “G” symbol in their thumbnails.

The group button G in the top panel of the lightroom view toggles grouping mode on and off. If grouping is off, each image is displayed as an individual thumb. If grouping is on, images of a group are collapsed, which means they are represented by one thumbnail. The image you see is called the group head. If you press “G” symbol in the thumbnail of a group, only this group gets expanded; if another group was expanded at that time, it gets collapsed. To collapse an expanded group again, just click on the “G” symbol of its group head.

You can define which image constitutes the group head, while in an expanded view of a group, clicking on the “G” symbol of the desired image.

If you are in collapsed view, and enter darkroom mode with an image group (eg. by double-clicking on the thumbnail), only the group head will be opened.

Image groups are a convenient way to protect an existing history stack against unintentional changes. Suppose you have just finalized an image; all you need to do now is generate a duplicate, make sure grouping is switched on, and the group collapsed. Whenever you open the image group again in darkroom, only the group head will be altered. The underlying duplicate remains unchanged. Please note that duplicating images only means that a second copy of your history stack, and a second small XMP file, is generated. There still is only one RAW file, so you don't waste a lot of disk space.

2.2. Lighttable panels

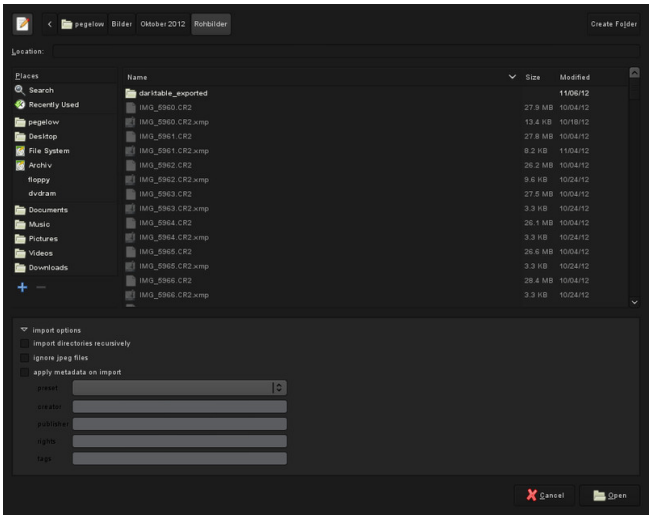
2.2.1. Import

This panel is for importing images into filmrolls. You can either import a complete folder, by pressing “import filmroll”, or a single image with “import single image”. You can also import directly from a connected camera.

All filmrolls are available in a list, which can be filtered using the editbox to fast find the filmroll of interest. Double click on a filmroll in the list and it will open in the lighttable. You can also click the items in *recently used collections* (see Section 2.2.4, “Recently used collections”) to open the latest ones you have worked with.

2.2.1.1. Import from filesystem

A filmroll is analogous to a folder in the filesystem. You can import either a single image, or a folder. If a folder is imported, the images in that folder will show up in a filmroll with the same name as the imported folder. If single images are imported, they will show up in the filmroll called “single images”.



Clicking on “import single image” or “import filmroll” opens a file selector dialog. Navigate through the filesystem, and select the item to import. On the lower part of the dialog, are some further import options.

As the name implies, checking “import directories recursively” will import images in the currently selected directory, and all subdirectories. It is not recommended, and wastes resources, to do this on an exhaustive list of images. darktable would generate thumbnails for all of them, but in the end only keep the recent ones in its cache. It is better to import images in smaller chunks, making logical filmrolls.

Checking “ignore jpeg files” is a good choice if there are JPEG images in the same folder that you do not want to process; eg. if the camera stores RAW+JPEG, you often only want to work on the RAWs, leaving the JPEG images alone.

You can also apply some metadata during import; see Section 2.2.11, “Metadata editor” for more details.

Importing a folder does not mean that darktable copies your images into another folder. It just means that the images are visible in lighttable and thus can be developed. If you delete an image or a folder from disk after having imported them, darktable cannot access

them anymore. Importing an image or folder in darktable is not a backup of your filesystem! Moreover, darktable does not watch for changes in the filesystem. Thus, if you add an image to a folder, after having imported that folder in darktable, the new image will not be shown until explicitely imported.

2.2.1.2. Import from camera

When a camera is detected, it will show up in the device panel. If you hover your mouse over the camera tab label, a tooltip will pop up with information about the camera, such as model, firmware version, and more. Depending on the camera's support, buttons with actions will be available such as “import images” and “tethering”.

Import images

This will bring up an import dialog, showing the images on camera that can be selected for import into a filmroll in darktable.

Tethering

Tethering is used to integrate darktable with your camera. While you take images with your camera, they are automatically imported into darktable, so you can review the result of the shoot. You can also setup remote capture jobs, controlling the number of images and time between captures, along with camera settings such as exposure time, aperture and more.

If supported by your camera, tethering will take you into the capture view for tethered shooting. Read more about tethering in Chapter 4, *Tethering*.

2.2.1.3. Supported file formats

darktable is focused on managing and developing camera RAW files. It supports a huge number of file formats from various camera manufacturers. In addition darktable can read specific *low dynamic range* and *high dynamic range* images - mainly for data exchange between darktable and other software.

In order for darktable to consider a file for import, it must have one of the following extensions (case independent): 3FR, ARW, BAY, BMQ, CAP, CINE, CR2, CRW, CS1, DC2, DCR, DNG, ERF, FFF, EXR, IA, IIQ, JPEG, JPG, K25, KC2, KDC, MDC, MEF, MOS, MRW, NEF, NRW, ORF, PEF, PFM, PNG, PXN, QTK, RAW, RAW, RDC, RW1, RW2, SR2, SRF, SRW, STI, TIF, TIFF, X3F.

If darktable was compiled with JPEG2000 support, these extensions are also recognized: J2C, J2K, JP2, JPC.

If darktable was compiled with GraphicsMagick support, the following extensions are recognized in addition to the standard ones: BMP, DCM, GIF, JNG, JPC, JP2, MIFF, MNG, PBM, PGM, PNM, PPM.

Camera RAW files

darktable reads RAW files using two open source libraries: RawSpeed (developed by Klaus Post) and - failing that - with LibRaw. The number of supported cameras and file formats is constantly increasing. It is beyond the scope of this manual to give an exhaustive list. Most modern camera models are supported, and new ones tend to get added very quickly. darktable does not decode images from cameras with non-Bayer sensors (e.g. Fuji X-Pro1 or Sigmas with the Foveon X3 sensor).

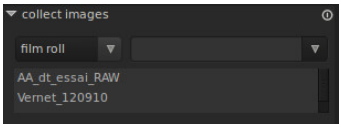
LDR image files

darktable natively reads “ordinary” images in JPEG, 8-bit/16-bit PNG and 8-bit/16-bit TIFF format. JPEG2000 is also supported if the required libraries are built into darktable at compile time. Similarly, if darktable was compiled with GraphicsMagick support, there are further import formats, like GIF, Dicom DCM, additional exotic TIFF formats, and some of Sun's “portable xyz-map” family.

HDR image files

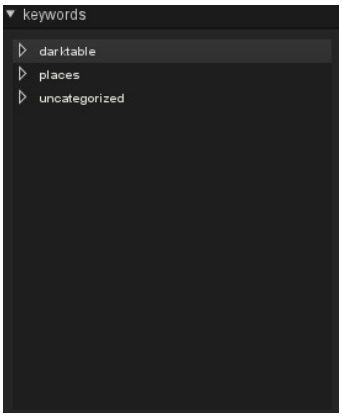
darktable reads high dynamic range images in OpenEXR, RGBE and PFM format.

2.2.2. Collect images



The current view in lighttable is called a collection. With this panel, you can manage the collection by filtering with various criteria. The left combo box lets you choose from different filters, such as “filmroll”, “history stack”, “date”, “camera” and much more. The table below the combobox lists all available data for the selected filter. You can then select the data of your choice by double-clicking.

2.2.3. Keywords

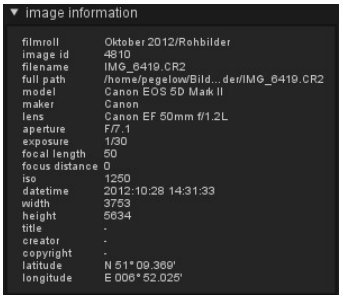


This module shows a hierarchial tree of your tags. You can arrange tags within this tree using drag-n-drop one tag onto another.

2.2.4. Recently used collections

This panel keeps tracks of the latest collections you have used, so you can jump within recently used collections without remembering what rules were specified in the collection.

2.2.5. Image information



This panel shows information embedded within an image's EXIF data. When hovering with the mouse over thumbnails, darktable will update this view, displaying information of the image currently under the mouse cursor. This panel is also available in darkroom, tethering and map view.

2.2.6. Select

This panel allows for a quick selection of images, according to some common criteria.

select all

Select all images in the current view (collection), with respect to the filters.

select none

De-select all images.

invert selection

Select all images that are not currently selected.

select film roll

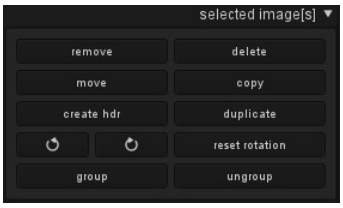
Select all images that are in the same filmroll as the currently selected images.

select untouched

Select all images that have not yet been developed.

2.2.7. Selected image(s)

This panel provides some actions that operate on selected images.



remove

Remove the selected images from the darktable database. Those images will not be shown in lighttable anymore, but remain on the filesystem. As darktable stores XMP files with your development parameters on disk, you can later fully reconstruct your work by just re-importing the images. When backing up your RAWs make sure to also save the XMP files!

delete

Physically delete selected images from filesystem. See also preference option “ask before erasing images from disk” (Section 6.1, “GUI options”). If this option is not active, darktable will delete the file(s) without further question! This is irreversible, and will also erase your development work of these images.

move

Physically move selected images (parent file plus accompanying .xmp sidecar file) to another filesystem folder.

copy

Physically copy selected images (parent file plus accompanying .xmp sidecar file) to another filesystem folder.

create hdr

Create a high dynamic range image from the selected images, and store it as a new source file in DNG format. Images need to be properly aligned, which implies that they have been taken on a sturdy tripod. You can also generate HDRs with programs like *Luminance HDR* [http://qtpfsgui.sourceforge.net/], and later import them into darktable for further processing (see Section 2.2.1.3, “Supported file formats”).

duplicate

Create a virtual copy of selected images within darktable, but not physically on your filesystem. It allows testing different developments for the same image, for example. Duplicate images share the same parent input file, but each have their own .xmp sidecar file.

rotation

Perform a counter-clockwise or clockwise rotation on selected images. The third button resets the image rotation to the value in the EXIF data.

group

Create a new group from selected images (see Section 2.1.3, “Image grouping”).

ungroup

Remove selected images from the group (see Section 2.1.3, “Image grouping”).

2.2.8. History stack

This panel allows manipulating the history stack (development) of images. For each image, development is written in a sidecar file (.xmp), and is fully non-destructive.

copy

Copy the history stack of the selected image. You will be prompted for which items are to be include. If more than one image is selected, the history stack is taken from the image that has been selected first.

copy all

Copy the complete history stack of the first selected image; all items will be included. If more than one image is selected, the history stack is taken from the image that has been selected first.

discard

Physically delete the history stack of the selected images. Beware, this action can not be undone!

overwrite/append

Describes how a new history stack behaves when pasted on an image that already has a history stack. “Overwrite” will delete the previous history stack, whereas “append” will concatenate the two history stacks.

paste

Paste a previously copied history stack onto all selected images. You will be prompted for which items to include. This button is greyed out, until a history stack is copied from another image.

paste all

Paste all previously copied items of a history history stack onto all selected images. This button is greyed out, until a history stack is copied from another image.

load sidecar file

Opens a dialog box to select an XMP file, thus loading a history stack that you can paste on images.

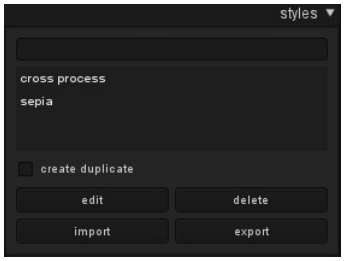
Files that were exported by darktable contain the full history stack if the file format supports embedded metadata (see Section 2.2.13, “Export selected”). You can load an exported image as a sidecar file in the same way as you do with an XMP file. This feature allows you to recover all parameter settings in case you have accidentally lost or overwritten the XMP file. All you need is the source image, typically a RAW, and the exported file.

write sidecar files

Write XMP sidecar files for all selected images. By default darktable will do this automatically whenever you work on an image and change the history stack. You can disable automatic sidecar file generation in the preferences dialog (see Section 6.2, “Core options”), which is not recommended.

2.2.9. Styles

This panel provides a powerful functionality in darktable: storing a history stack as a style, and applying it to other images. Styles are created in the darkroom using a button placed below the history panel (see Section 3.3.3, “History”). They are managed within this lighttable panel, which allows you to apply and delete styles.



create duplicate

When applying a style to selected images, activating this box creates a duplicate of the image before applying the style. Disable this option if you want to try various styles without creating multiple duplicates.

edit

Styles are a collection of history stack items. After pressing “edit”, you are prompted with a dialogue to include or exclude specific items from the stack. Check option “duplicate” if you want to create a new style, instead of overwriting the existing one; you need to provide a new style name in this case.

delete

This deletes the selected style, without further question.

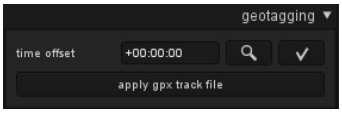
import

You can import a style which has been previously saved by darktable. darktable stores styles as XML files with the extension ".dtstyle".

export

This option saves a selected style to disk as a .dtstyle file. This allows sharing styles with other users.

2.2.10. Geotagging



Use this panel to import and apply GPX track data on a selection of images. You can add a time offset to existing GPX tracks, to correct time differences between your camera and GPS receiver. Alternatively, you can manually geotag images within the *Map* view (see Chapter 5, *Map*).

2.2.11. Metadata editor

Edit metadata of an image, like *title*, *description*, *creator*, *publisher*, or *rights*. You can define your own presets, if you want to apply specific settings frequently.

clear

Delete existing metadata from the selected image(s).

apply

Apply new settings, as defined in the fields above, to the selected image(s).

2.2.12. Tagging

This panel is for managing tags on images. Tags are stored in both, sidecar files (.xmp), and within the darktable database for a faster access. The panel is divided into two parts: the upper part contains the tag(s) currently set for the image under mouse (if the mouse is over an image) or the selected image (if the mouse is outside the lighttable). The lower part contains all available tags, which can be filtered in the upper text box.

attach

Attach the selected tag(s) from the list below to all selected images.

detach

Detach selected tag(s) from the list above from all selected images.

new

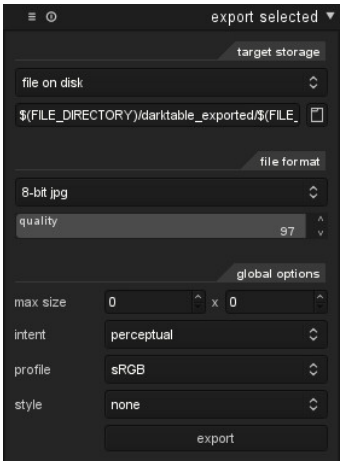
Create a new tag for the list.

delete

Delete a tag from the list.

2.2.13. Export selected

Each workflow ends in this panel: the export of your developed images. You can export either to a file on disk, or to various on-line storage places. Tip: you can use *ctrl-e* from within darkroom mode to export.



target storage

Where to store your selected images. Different back-ends are implemented, including file on disk, a LaTeX book template and various web albums. Depending on the selected target, you will be asked to give additional information, like filenames, or account name and password.

filename template

You can define filenames that darktable generates for export. Several pre-defined variables can be used as placeholders:

$(ROLL_NAME)	roll of the input image
$(FILE_FOLDER)	folder containing the input image
$(FILE_NAME)	basename of the input image
$(FILE_EXTENSION)	extension of the input image
$(SEQUENCE)	a sequence number within export job
$(YEAR)	year at date of export
$(MONTH)	month at date of export

$(DAY)	day at date of export
$(HOUR)	hour at time of export
$(MINUTE)	minute at time of export
$(SECOND)	second at time of export
$(EXIF_YEAR)	exif year
$(EXIF_MONTH)	exif month
$(EXIF_DAY)	exif day
$(EXIF_HOUR)	exif hour
$(EXIF_MINUTE)	exif minute
$(EXIF_SECOND)	exif second
$(STARS)	star rating
$(LABELS)	colorlabels
$(PICTURES_FOLDER)	pictures folder
$(HOME)	home folder
$(DESKTOP)	desktop folder

output directory

Pressing button opens a dialog to select the parent directory for export.

file format

darktable can export to various file formats. For some of them you need to define the desired bit depth of the exported image. If you export to a JPEG file you can define an output quality. Higher values will lead to larger file sizes.

If the file format supports embedded metadata, like JPEG, JPEG2000 and TIFF, darktable will store the history stack as XMP tags within the output file. This information can later be used to reconstruct your parameters and settings that have produced the exported image (see Section 2.2.8, “History stack”). If you don't want to distribute history stack data with your images, there are various tools to delete embedded XMP tags. As an example you can use the program *exiftool* [http://www.sno.phy.queensu.ca/~phil/exiftool/] with:

```
exiftool -XMP:all=   image.jpg
```

max size

Set the maximum width and height of the output images in pixels. Set both to a value of "0" to export with full resolution. darktable currently can only do down-scaling; the maximum output resolution is defined by the parent image.

Caution: it's a frequent pitfall to accidentally put low values, like 1 or 10, in these fields, causing darktable to produce miniature output files. You might think darktable's output is broken, but in fact it only generated what you asked for.

intent

This option lets you define the intent, i.e. the way darktable will deal with out-of-gamut colors. See Section 3.4.8.3, “Output color profile” for a more detailed description of the available options.

profile

This defines the output color profile. Select “image settings” if you want the settings in the *output color profile* module of the individual images to take precedence.

style

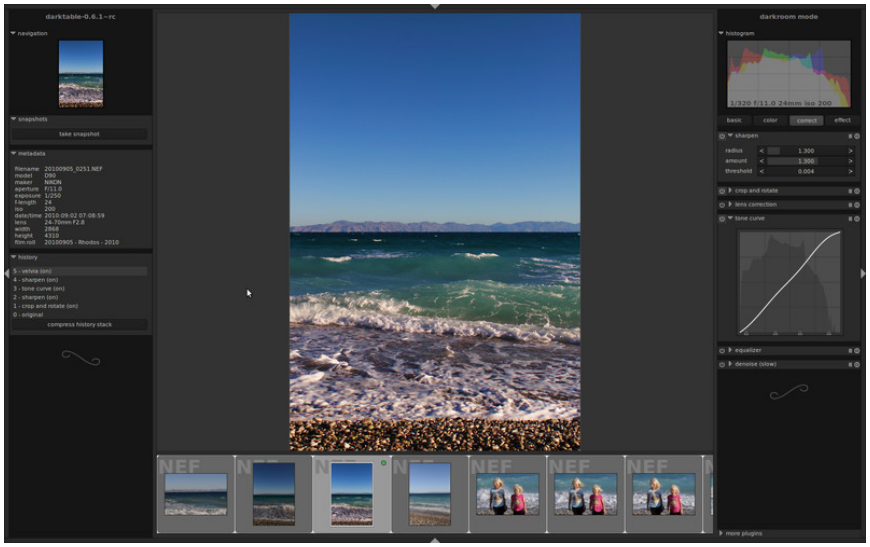
This option lets you choose a style, i.e. a collection of history stack items, which darktable concatenates with the existing history stack to generate the output image. These history items are only added temporarily; the original history stack is not overwritten. You can use this feature to add processing steps and parameters that you want to be applied specifically to exported images, e.g. you may define a style that adds a stronger level of sharpening when you produce scaled-down JPEG files for the internet. Learn more about styles in Section 2.2.9, “Styles”, and Section 3.3.3, “History”.

export

Pressing this button starts a background job to export all selected images. A bar at the bottom of the left side panel displays the progress. Whenever a file is successfully exported, a notification message pops up for a few seconds. You may click on the pop-up to make it disappear.

Chapter 3. Darkroom

The darkroom view is where you develop your image.



3.1. Overview

Darkroom mode is for photograpic development of the image you selected from the lighttable. Lots of tools, named modules, are available for processing that image.

On the left hand side you have navigation, snapshot and history panels, described in Section 3.3, “Darkroom panels”. In the right hand panel you can see the histogram and a list of modules available for working with the image. At the bottom of the right hand panel you can enable/disable view of individual modules.

Modules are organized into five functional groups: basic, tone, color, correction and effect as described in Section 3.3.6, “Module groups”. You either view all modules in one long list or instead click on a group to just display modules belonging to that group.

For those interested in how it works, modules are applied on images following a bottom-to-top order. That means, that if you activated the view of all modules (by selecting no groups), the first module to be applied will frequently be *invert* at the bottom of the right panel, whereas the last one will be *dithering* at the top of the right panel. That order can not be changed by user, so modules are always applied following the same order.

You can use *middle-click* to zoom 1:1. A double *middle-click* takes you to 2:1.

You normally export multiple images from the lighttable view but you can also export the current image directly from the darkroom by using the shortcut *ctrl-e*. Export parameters are then those currently selected in the lighttable.

3.2. Filmstrip

The optional filmstrip can be used to quickly switch between images while remaining in the darkroom view. The images viewed are the same as the ones in the lighttable view.

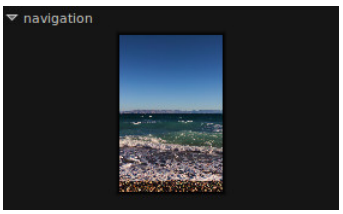
The filmstrip can be switched on and off using the shortcut *ctrl-f*. You can scroll with your mouse to quickly navigate through the images and change the height of the filmstrip panel by dragging its top.

3.3. Darkroom panels

This section contains documentation for panels that are specific to the darkroom view.

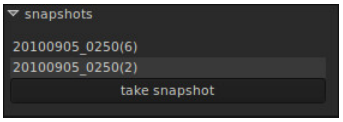
3.3.1. Navigation

The navigation panels shows a full preview of your image with a rectangle showing the currently visible zoom area. Drag the rectangle around to pan the zoomed-in view.



3.3.2. Snapshots

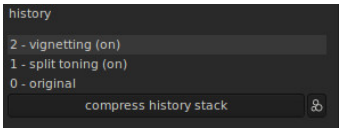
You can take snapshots of images as you process them. A snapshot is stored as a bitmap of the current center view and is kept as long as you stay in the darkroom. A snapshot can then be selected and overlayed in the current center view to help you with a side by side comparison (left: snapshot, right: active) when you are tuning parameters of a module. This can also be combined with history (see Section 3.3.3, “History”) to compare the snapshot against different stages of development.



You can control the split view by moving the splitline back and forth. If you hover with the mouse over the splitline, a small rotation icon will appear on the center of the line. Click it to change between vertical and horizontal split view.

3.3.3. History

The history stack lists every change of state (activate/de-activated) for all modules. Here you can *select* a point in stack to return to that point of development history. If you then activate a new module or change a module parameter, all modules above the current point will be discarded.



Caution: activating any module action using key accelerators will discard all modules above the currently selected one. It is easy to lose all development work on an image this way!

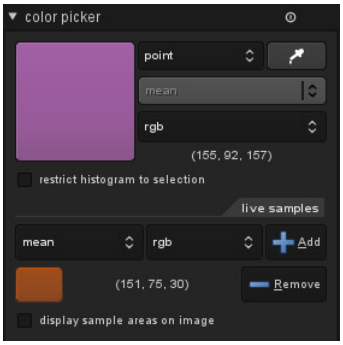
Hitting “compress history stack” generates the shortest history stack that produces the current image, i.e. suppressing all de-activated modules. This also will discard all modules above the currently selected one.

The button to the right lets you create a new style for applying your history stack to other images. Use the first line of the popup dialog window to name your style and the second to add a searchable description. You are prompted for which of the current history stack modules to include in the style.

Once created styles are then managed and applied to other images in the lighttable's *styles* panel (see Section 2.2.9, “Styles”).

3.3.4. Global color picker

Using the global color picker you can take color samples from your image, display their values in multiple ways and compare colors from different locations. The color picker is activated by pressing the icon. There are multiple parameters for controlling how the color picker works, whose settings remain in effect until you leave the darkroom mode.



Besides the global color picker described here there are also local color pickers in some of the modules (eg. *tone curve*). Global and local color pickers are different. The global color picker works in monitor color space and takes samples after the complete pixelpipe has been processed. The local color pickers run in the color space of the individual module, which is usually Lab; they reflect the input and output data of that specific module within pixelpipe.

The global color picker can be run in point or area mode. When in point mode only a small spot under your cursor is taken as a sample. In area mode you can draw a rectangle and darktable samples the area within that rectangle. The combobox to switch between point and area mode can also be used to toggle the mode of local color pickers.

If samples are taken in area mode, darktable will calculate mean, min and max color channel values. A combobox allows you to select which of those are displayed. For obvious statistical reasons mean, min and max are identical for the single sample of point mode.

A color swatch representing the sampled point or area is displayed. Numerical values are shown as well. As said before global color picker works in monitor RGB color space. You can also let darktable translate these numerical values into Lab color space. Beware that Lab values are approximated here; depending on monitor color profile there can be some deviations from the real values.

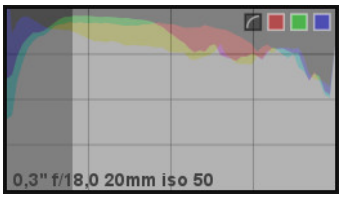
When the checkbox “restrict histogram to selection” is ticked, only the values of your selected area or point are taken into account by the main histogram at the top of the right hand panel (see Section 3.3.5, “Histogram”). This is a way to show which tonal values are present in a specific area.

The sampled colors in either area or point mode can be “stored” as live samples by pressing the “add” button. darktable will then show a color swatch and numerical values for each stored sample. You can once again select which numerical value (mean, min, max) is to be displayed and if this is to be done in RGB or Lab color space.

Newly created live samples are not locked. If you change your image the changes will be reflected in your live samples. Use this if you want see how changing parameters effects different parts of an image. Clicking on a live sample's color swatch locks it and a lock symbol is displayed. Further image changes will then no longer affect the sample. You can for example take two live samples from the same location and lock just one of them to provide a before and after sample comparison.

Live sample locations are indicated in your image if you check option “display sample areas on image”.

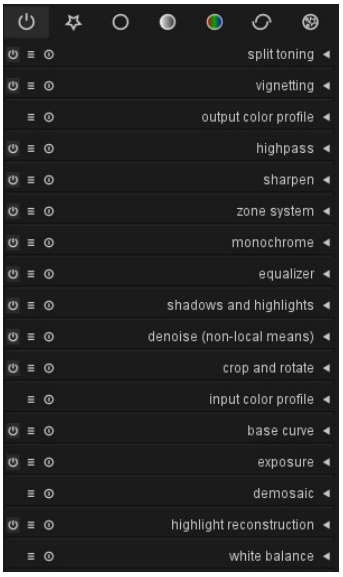
3.3.5. Histogram



This shows a histogram of the developed image's light levels. In its default state curves for all three RGB color channels are displayed. You can toggle the colored squares to enable or disable specific color channels. A curve button is also provided to toggle between linear and logarithmic view.

The histogram is directly linked to the *exposure* module described in Section 3.4.6.4, “Exposure”, and you can operate some of the exposure module's controls from the histogram. You can left-click towards the right hand side of the histogram and then drag right to increase or drag left to decrease the exposure. In a similar manner you can control the black level by clicking and dragging in the left hand side.

3.3.6. Module groups



The module groups button bar gives you quick access to darktable's processing modules.

Here follows a description of the module groups available:

	Active	Modules you have activated and are using on the current image.
	Favorites	Modules you have marked as favorites using *more modules* (see Section 3.3.7, “More modules”).
	Basic	Modules that are frequently used, such as exposure, temperature etc. (see Section 3.4.6, “Basic group”).
	Tone	Modules for working with the image's tonal values, e.g. levels, tonemap etc. (see Section 3.4.7, “Tone group”).
	Color	Modules for processing colors, such as color correction, vibrance etc. (see Section 3.4.8, “Color group”).

	Correction	Modules making corrections to the image, e.g. denoise, CA correction etc. (see Section 3.4.9, “Correction group”).
	Effect	Modules with a more artistic output, such as vignetting, softening etc. (see Section 3.4.10, “Effect group”).

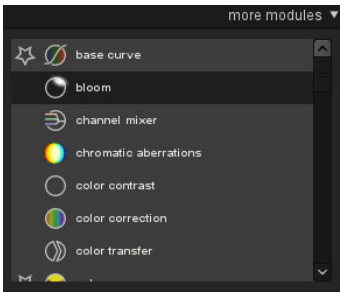
Clicking on one of the group symbols will show the modules in that group. If you once again click on the symbol, grouping will be de-activated and all non-hidden modules will be shown in one long list. This list shows the sequence in which modules are applied from bottom to top. As a general principle darktable applies modules in a pre-defined sequence.

For those who are interested here is some information about darktable's internals: The long list helps you to figure out in which color space a specific module acts. In fact there are only three modules which convert from one color space to another: *demosaic*, *input color profile* and *output color profile*.

up to *demosaic*	Image is in raw data format with only latent colors. Each pixel carries lightness and color information for only one base color. Please mind that some of the modules in this part can also act on non-RAW input images in RGB format.
between *demosaic* and *input color profile*	Image is in RGB format within the color space of the specific camera or input file.
between *input color profile* and *output color profile*	Image is in Lab format. This is a very huge universal color space which covers all colors visible to the human eye (and even more). As darktable processes images in 4x32-bit floating point buffers, we can handle the Lab color space without risking banding or tonal breaks.
after *output color profile*	Image is in RGB format as defined by the selected display or output ICC profile.

3.3.7. More modules

More modules at the bottom of the right panel is used to show the less frequently used modules. By default only standard modules are shown to the user but you can use this function to make the extra modules visible, or alternatively to hide away modules you don't typically use.



Each module is shown with a small icon next to its name. Left-click with your mouse to toggle the status between visible, hidden and favorite. Favorite modules are indicated by a star in front of the icon and in addition to appearing in their normal module group will also be visible in the module group *favorites*. This is a good way to get fast access to modules that you use very frequently. Visible modules are indicated in the list by a light grey background whilst hidden modules have a dark grey background and do not display any of their controls.

Hiding or un-hiding modules is not meant to be part of your daily workflow, you should only occasionally need to review the modules you typically use.

3.3.8. Bottom panel

The bottom panel provides quick access to apply presets and styles to your image and allows to activate the over/underexposure warning.

3.3.8.1. Quick access to favorite presets

Clicking the icon opens a combobox that gives you quick access to your favorite module's presets. Click on the preset name to apply it to the image.

3.3.8.2. Quick access to styles

Clicking the icon opens a combobox with your styles. Hovering with the mouse over a style name opens a tooltip showing the involved modules. Click on a style name to apply that style to the image.

3.3.8.3. Over/underexposed warning

By clicking the icon an over/underexposed warning is toggled on or off. Pixels outside the dynamic range, close to pure white or close to pure black, are prominently displayed in a signal color. You can also activate the over/underexposure warning with the keyboard shortcut “o”.

Right-clicking on the icon opens a dialog with configuration parameters.

color scheme

In the default color scheme underexposed pixels are shown in blue and overexposed pixels in red. These colors are easy to identify in most cases. In some cases you may want to change the color scheme to “black & white” or “purple & green”, eg. if you experience overexposed highlights in red blossoms.

lower threshold

Sets the threshold for underexposure warning, expressed as a percentage of the maximal brightness.

upper threshold

Sets the threshold for overexposure warning, expressed as a percentage of the maximal brightness.

3.4. Modules

Each correction, enhancement or effect is implemented as a module. This section includes documentation of each module and its specific parameters.

A module has an expander bar 

. Clicking on the name of the module expands the module's GUI with all parameters. If you expand too many GUIs you may easily lose track - it's up to the user to do the housekeeping. Alternatively you can expand the module's GUI with *shift-click*, which expands just the desired GUI while collapsing all others.

Expanding a module does not activate it. You need to click the icon to turn a module on or off.

Icon accesses the module's available presets or creates a new preset from your current settings (see Section 3.4.4, “Module presets”).

The icon is used to reset the module parameters to their default values.

Many of darktable's modules can have multiple instances, each with different settings. Click on the icon to generate new instances and control existing ones (see Section 3.4.5, “Multiple instances”).

3.4.1. Interacting with modules

The most frequently used control elements are sliders, comboboxes and curves.

Sliders

For each slider, you can interact in four different ways, depending on the level of control you need.

1. Triangular marker

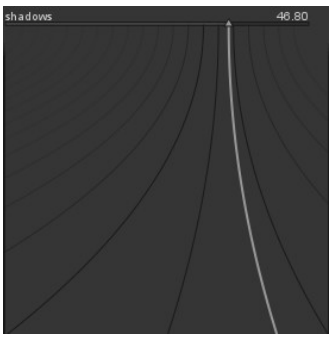
   Left-click the slider's triangular marker and drag it to the left or right.

2. Mouse wheel

   Hover over any place on the slider with your mouse, then use your mouse wheel to adjust the value step by step.

3. Right-click

   When your mouse is over a slider right-click gives you a multi-functional pop-up below the slider for fine control with your mouse or numerical entry using the keyboard, there is no cursor, you just start typing.



darktable's innovative input method: for both coarse and fine value adjustments in a single control element.

A bent line extending from the triangular marker moves as you move your mouse. The closer your mouse pointer is to the triangular marker the coarser the control; the further away from the triangular marker the finer is your control. Left-click with your mouse to accept the current value and go back to normal control.

4. Double-click

   You can double-click on a parameter label to reset its value to default.

Comboboxes

Clicking on a combobox will open a list of available options. Click on the item you want to select. Sometimes the selection list opens close to the bottom or top of the screen and only part of the items are visible; scroll with your mouse wheel to bring up the full list.

Curves

Some modules are controlled by adjusting curves. More detail is given later in this chapter when the respective modules are explained.

3.4.2. Blending operators

Blending is a functionality that works on a per-module basis by reprocessing a module's output with its input in various ways.

Modules that support blending inherit additional controls at the bottom of the module's user interface.

3.4.2.1. Usage

blend mode

There are several blend modes implemented and more might be added in future. For now all the common ones are there and you will recognize a few of them from other imaging software. A good introduction on many common blend modes is given in *The Gimp Manual (Chapter 8.2, "Layer Modes")* [http://docs.gimp.org/2.8/en/gimp-concepts-layer-modes.html]. Therefore we only discuss a few blend modes here in more detail.

off

If blend mode is set to its default "off" value no blending will be done and all other blending related controls are hidden.

normal

This will probably be the most used blend mode. It just mixes input and output and therefore reduces the strength of a module's effect.

inverse

This blend mode acts similarly to blend mode "normal", only that the role of input and output of the current module are reversed. Tip: This can be used together with conditional blending (Section 3.4.3, "Conditional blending") to get the effect of an inversed blend mask.

unbounded

This blend mode acts similarly to blend mode “normal”, except that input and output data are not clamped to a particular min/max value range unlike all other blend modes which do clamp their input and output. In some cases (e.g. highly color saturated extreme highlights) it is important to let unbound values travel through the pixelpipe in order to properly deal with them at the right place (e.g. in module *output color profile*).

lightness

This blend mode mixes lightness from the input and output images. Color data (chroma and hue) are taken unaltered from the input image.

chroma

This blend mode mixes chroma (saturation) from the input and output images. Lightness and hue are taken unaltered from the input image.

hue

This blend mode mixes hue (color tint) from the input and output images. Lightness and chroma are taken unaltered from the input image. Caution: When modules drastically modify hue (e.g. when generating complementary colors) this blend mode can result in strong color noise.

color

This blend mode mixes color (chroma and hue) from the input and output images. Lightness is taken unaltered from the input image. Caution: When modules drastically modify hue (e.g. when generating complementary colors) this blend mode can result in strong color noise.

coloradjustment

Some modules act predominantly on the tonal values of an image but also perform some color saturation adjustments, e.g. module *levels* and *tone curve*. This blend mode takes the lightness only from output data and mixes colors from input and output enabling control of the module's color adjustments.

opacity

This slider controls the amount of blending. A value of 100% gives the module's full effect into the pixelpipe; a value of 0% means no effect of this module. With blend mode “normal” opacity controls the strength of a module's effect.

blend if

Most modules offer additional control with a user-defined blend mask. In order to activate these controls you need to switch from “blend uniformily” to “blend only, if..”. More details on conditional blending are given in a separate chapter (Section 3.4.3, “Conditional blending”).

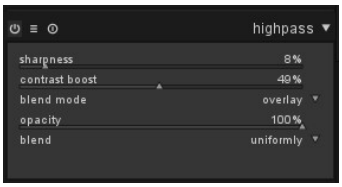
3.4.2.2. Examples

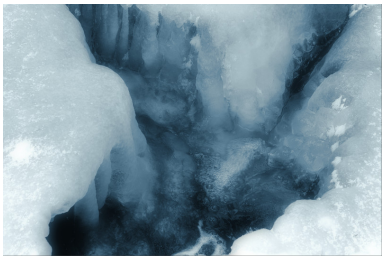
Texturing an image

The watermark module supports SVG files with embedded images that can be used as a texture source. Blending operators then allow control of how that texture is expressed.

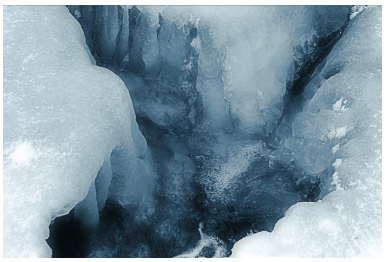
Gritty details

When blending operators were introduced into darktable, a new module named *highpass* (see Section 3.4.10.7, “Highpass”) was added. It provides a highpass filter of the image to be implicitly used with blending. It allows to produce a gritty detailed image and is a widely used workflow with other imaging softwares.





This is the original image, pretty heavily processed: first *monochrome*, then some blue *splittoning* but as you see it lacks pop in details and is a bit out of focus...



Here we applied the highpass filter with the values shown above. You can now see that the details are greatly boosted and we now have a really gritty detailed image.

3.4.3. Conditional blending

Conditional blending offers fine-grained selective control over how individual pixels are blended. It does so by automatically generating an intermediate blend mask from user defined parameters. These parameters are color coordinates not the geometrical coordinates used in conventional hand drawn masks.

Conditional blending is a powerful tool with a certain level of complexity.

3.4.3.1. Working principle

For each data channel of a module (Lab, RGB) and additionally for several virtual data channels (e.g. hue, saturation) users can construct a per-channel opacity function. Depending on the pixel's value for this data channel this function determines a blending factor between 0 and 1 (or 100%) for that pixel.

Each pixel of an image thus has different blending factors for each of its data channels (real and virtual). All blending factors are finally pixel-wise multiplied together with the value of the global opacity slider (see Section 3.4.2, “Blending operators”) to form a blend mask for the image.

If for a given pixel the blend mask has a value of 0, the input of the module is left unchanged. If for a pixel the blend mask has its maximum value of 1 (or 100%), the module has full effect.

3.4.3.2. Usage

When conditional blending is activated with option “blend only, if..” an additional set of tabbed controls is shown.

Channel tabs

Each tab selects a data channel - real or virtual. Modules acting in Lab color space have data channels for L, a, b, C (chroma of LCh) and h (hue of LCh). Modules acting in RGB color space have data channels for g (gray), R, G, B, H (hue of HSL), S (saturation of HSL), and L (lightness of HSL). Consult for example Wikipedia's article on color spaces [http://en.wikipedia.org/wiki/Color_space] for a deeper look.

Each tab provides two sliders for its data channels: one for the input data that the module receives and one for the output data that the module produces prior to blending.

Color channel sliders

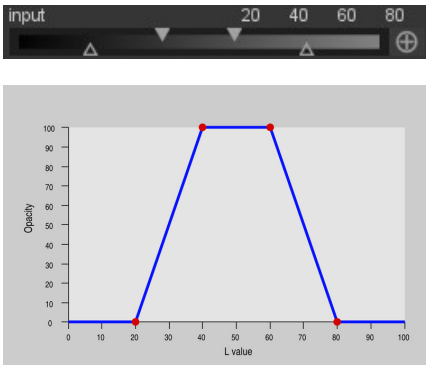
With the color channel slider you construct a trapezoidal opacity function. For this purpose there are four markers per slider. Two triangles above the slider mark the range of values where opacity is 1. Two triangles below the slider mark the range values where opacity is zero. Intermediate points between full and zero opacities are given a proportional opacity.

The filled triangles, or inside markers, indicate the closed (mostly narrower) edge of the trapezoidal function. The open triangles, or outside markers, indicate the open (mostly wider) edge of the trapezoidal function.

By default the markers are placed at the ends of the slider. The sequence of the markers always remains unchanged: they can touch but they can not switch position.

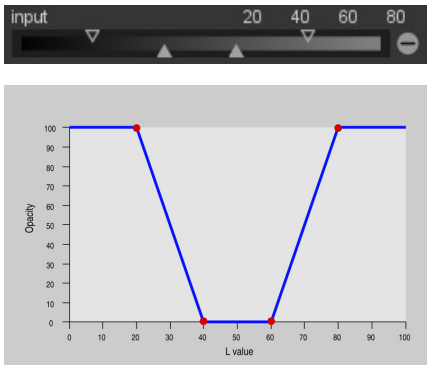
A polarity button to the right of the slider switches between include and exclude function modes with visual confirmation provided by exchanging the upper and the lower triangle markers. These two types of trapezoidal functions are represented graphically in the following images.

Include function



A trapezoidal function that selects a narrow range of values for blending.

Exclude function



A trapezoidal function that excludes a narrow range of values from blending.

Control buttons

Control buttons help you when designing a blend mask. With the eye button you can temporarily deactivate the blend mask; blend mode and global opacity slider remain in effect. Switch this button on and off to see if the module is acting on the image as intended. With the mask button you can directly see the blend mask as a yellow overlay over a black and white copy of the image; the stronger the yellow color, the higher the blend mask value. With the color picker button you can select a probe from your image. The corresponding values for the real and virtual data channels are then displayed within each color channel slider. With the reset button you can put all conditional blending settings back to their default state.

Mask blur

Blurring the mask creates a softer transition between blended and unblended parts of an image and avoids artifacts. The mask blur slider controls the radius of a gaussian blur applied to the final blend mask. The higher the radius, the stronger the blur or set to 0 for an unblurred mask.

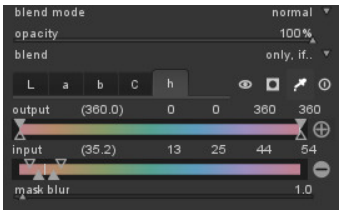
3.4.3.3. Examples

Colorkey effect

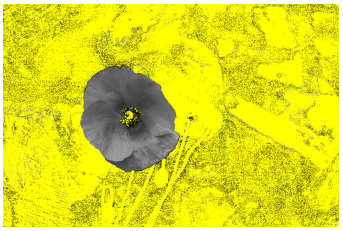
To create a colorkey effect with this poppy blossom in red and the remainder of the image in monochrome, we could apply module *monochrome* to all parts of the image except for of the saturated red petals.

We choose the hue channel to control our mask as hue provides good separation between the petals and background.

These settings in hue channel construct a blend mask that excludes the red petals. The small white bar in the gradient was obtained by using the color picker on one of the petals and the markers then closely centered on the indicated hue to increase the selectivity of our mask.



The resulting blend mask.



The final image after module *monochrome* is applied.

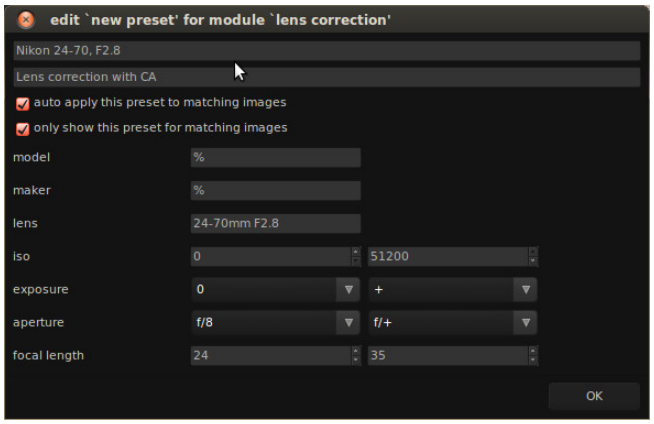
3.4.4. Module presets

Presets are stored configurations for a module's parameters. Some modules already have internal pre-defined presets but you can also define your own. Both internal and user-defined presets are displayed by clicking the icon with the currently activated preset shown in bold text.

The preset system also supports automatic preset selection based on image data such as focal length, ISO, camera model and other fields.

3.4.4.1. Creating a new preset

First configure the module's parameters then click the icon and select “store new preset”. The following dialog will be shown for configuring the preset:



The first two fields are used to name and describe the preset.

In the example above we have also checked the auto apply option. This brings up additional selection fields where you can define a filter used to decide if the preset should be automatically applied when opening other similar images in darkroom for the first time. The example dialog above sets up following rules: if lens name matches and aperture is greater or equal to f/8 and focal length is between 24 and 35mm the preset will be automatically applied. Also the second checkbox is clicked so this preset will only be shown in the preset list if the image matches the rule.

darktable finds this data in your image's EXIF information. If you want a preset to be applied to all images from a specific camera leave all fields at default values except for the model field.

Tip: The *image information* panel for your image displays your model name, use this to ensure you have the correct spelling (see Section 2.2.5, “Image information”).

3.4.4.2. Managing Presets

Both user created and pre-defined presets can be viewed and managed from within the presets menu (Section 6.4, “Presets”) in the preferences dialog (see Chapter 6, *Preferences and settings*).

3.4.5. Multiple instances

Many of darktable's modules can be applied multiple times. Each instance of that module behaves like any other module, taking its input from the module below in the pixelpipe delivering its output to the module above.

3.4.5.1. Typical use cases

There are many occasions where it makes sense to have an operation act more than once in the pixelpipe. Here are a few use cases.

Most of our modules are highly versatile and depending on parameters can deliver quite varying effects. For example the *fill light* module (Section 3.4.7.1, “Fill light”) allows local modification of lightness based on pixel values. You might want to do two lightness corrections in your image at the same time - one for dark tones and another one for lighter tones.

You might want to apply a denoising module like *denoise (profile)* (Section 3.4.9.3, “Denoise - profiled”) with two different parameter sets. One to do luma denoising and another set of parameters to do chroma denoising. You could do so by generating two instances and use the first one only on luma by selecting blend mode “lightness” and use the second one just for chroma by selecting blend mode “color” (see Section 3.4.2, “Blending operators”).

In an even more elaborate case you could have a module act on different parts of your image. As an example you might want to apply a certain gradation curve with module *tone curve* (Section 3.4.7.3, “Tone curve”) to your complete image and have a second curve being applied specifically to skin tones. All the controls offered by *conditional blending* (Section 3.4.3, “Conditional blending”) are at your hand to select those parts of an image where each of the module instances is applied.

Please be aware that of course each instance also adds to the workload of your pixelpipe. Generating too many instances - especially of the more demanding modules - will certainly cause some noticable slow-down.

3.4.5.2. Managing instances

When clicking on the  icon a drop-down menu will appear.

Selecting “new” generates a new module instance below any existing ones. Each new instance gets its own complete set of GUI controls and a number appended to the base module name for distinction.

To delete an instance just press “delete” from the drop-down menu.

darktable applies all modules in a defined order according to their type. Therefore all instances of a particular module will occur together in the pixelpipe. You can however decide on the relative order in which the different instances of a module are applied by selecting “move up” or “move down” to shift the position of the instance among its peers.

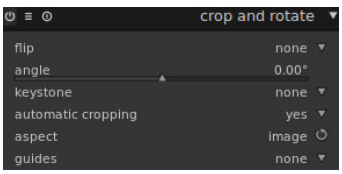
3.4.6. Basic group

The basic group of modules contains the modules for basic development. These are ones you probably will use most often, such as exposure, temperature etc.

3.4.6.1. Crop and rotate

Overview

This module is used to crop, rotate and correct perspective distortions of your image. You can overlay your image with various helpful guidelines that assist you using the tools.



Some of the tools of this module, namely adjustment of angle and corrections of perspective distortion, will require the original image data to be interpolated. For best sharpness results set “lanczos3” as pixel interpolator in *core options* (see Section 6.2, “Core options”).

Usage

Whenever the user interface of this module is in focus, you will see the full uncropped image overlayed with handles and guiding lines.

First off, select what aspect ratio you want and size the crop boundaries by dragging border and corner handles. Use the button right of the aspect box, to swap between portrait and landscape mode. You can move around the crop rectangle by holding down left mouse button and move around. When you are done and want to execute the crop, just give focus to another module. You can at any time change your crop area by just revisiting this module.

flip

This tool is used to flip the image on the horizontal,vertical or both axis.

angle

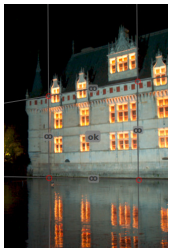
This tool corrects the rotation angle helping you level an image. You can either set a numerical value or use your mouse directly on the image. To use your mouse, right-click, hold it down and draw a line along the horizon; as soon as you release the mouse button the image is rotated so the line you drew matches the horizontal axis.

keystone

This tool is used to correct perspective distortions in your image. Useful for example when you shoot a high building from ground with a short focal length, aiming upwards with your camera. The combobox lets you select the type of correction you want to use :

vertical	if you want to limit the correction to vertical lines
horizontal	limit the correction to horizontal lines
free	if you want to correct horizontal and vertical lines

Depending on the selected correction type you will see two or four straight adjustment lines overlaid to your image. Two red circles on every line let you modify the line positions with your mouse. Each line additionally carries a “symmetry” button. If activated (and highlighted in red) all movements of the affected line will be mirrored by the opposite line.



In order to correct perspective distortions, you need to find suitable horizontal and/or vertical features in your image and align the adjustment lines with them. When done, press the “OK” button, which is located close to the center of your image. The image will be corrected immeditely. You can at any time come back and refine your corrections by selecting “correction applied” in combobox keystone.

automatic cropping

Use this options to avoid black edges on the image borders. Useful when you rotate the image.

aspect

Here you can change what aspect ratio you want to have on the result, thus constraining the ability to drag and crop rectangle out of the aspect ratio of your choice. Many common numerical ratios are pre-defined. You can also select any other ratio after opening the combobox and typing it in the form of “x:y”. A few special aspect ratios deserve explanation:

free	free forming the rectangle without any ratio restrictions
image	this option constrains the ratio to be equal to image ratio
golden cut	this option constrains the ratio to be equal the golden number
square	this option constrains the ratio to be 1

guides

Many self-explaining guides are available to help you compose your image.

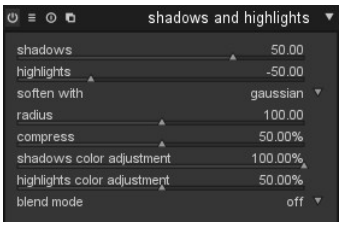
Examples

A cropped image in center view when module crop and rotate is in focus. The cropped area is visible as well as some guiding lines.

3.4.6.2. Shadows and Highlights

Overview

The shadows and highlights module allows adjustment to the tonal range of darker parts of an image (shadows) and lighter parts (highlights); it can bring back details in shadows and highlights by enhancing local contrast.



Usage

shadows

This slider controls the effect on shadows; positive values will lighten up shadows while negative values will darken them.

highlights

This slider controls the effect on highlights; negative values will darken highlights while positive values will lighten them up.

soften with

This combobox choses the underlying blurring filter, gaussian or bilateral. Try bilateral filter if you experience halos with gaussian blur.

radius

This slider controls the radius of the involved blurring filter. Higher values give softer transitions between shadows and highlights but might introduce halos. Lower values will reduce the size of halos but may lead to an artifical look. As said, bilateral filter is much less prone to halo artifacts.

compress

This slider controls how strong the effect extends to midtones; high values reduce the effect to the extreme shadows and highlights; low values cause strong adjustments also to midtones. You normally only need to touch this parameter if you want to limit the effects to the extreme shadows and highlights; increase the value in this case. At 100% this module has no visible effect any longer as only absolute black and abolute white are affected.

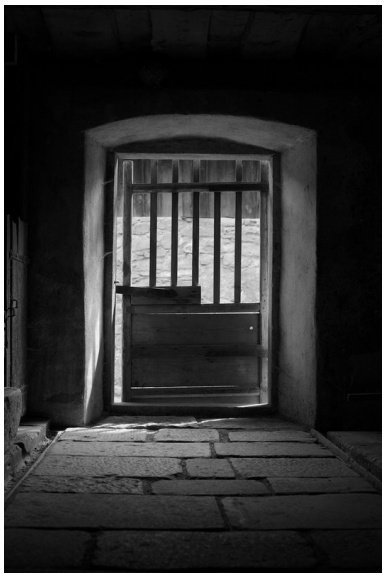
shadows color adjustment

This slider controls the color saturation adjustment made to shadows; high values cause saturation enhancements on lightened shadows; low values cause desaturation on lightened shadows. It is normally safe to leave this at its default of 100%. This gives a natural saturation boost on shadows - similar to the one you would also expect in nature if shadows would receive more light.

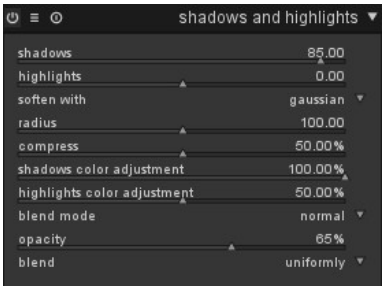
highlights color adjustment

This slider controls the color saturation adjustment made to highlights; high values cause saturation enhancements on darkened highlights; low values cause desaturation on darkened highlights. Often highlights do not contain enough color information to give convincing colors when darkened. You might need to play a bit with this parameter in order to find the best fitting value depending on your specific image; but be aware that sometimes results still might not be fully satisfying.

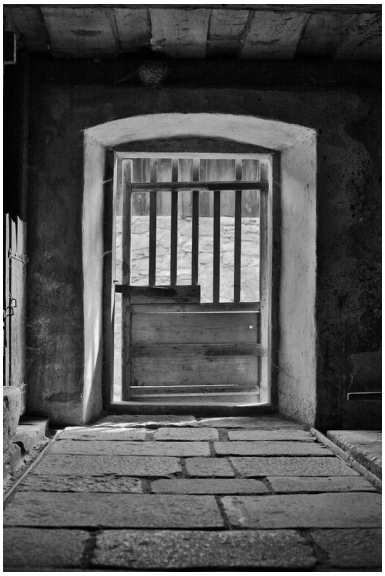
Examples



Original image exposed for the outer sunlit wall to avoid clipped highlights. As a consequence the interior of the barn has pitch black shadows.



Shadows get lightened; highlights are untouched; overall effect toned down a bit by *blend mode* “normal” and an opacity of 65%.

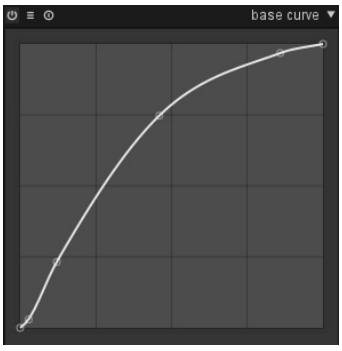


Resulting image.

3.4.6.3. Base curve

Overview

Camera sensors provide data in linear RGB format, the original image appears flat and dull. That's the reason why camera manufacturers apply their characteristic base curves to the RAW data when they generate in-camera JPEG images with better colors and contrast. darktable comes with base curve presets that mimic the curves of various manufacturers. These are automatically applied to RAW images according to the manufacturer ID found in EXIF data.



Usage

You can adjust an existing base curve or create a new one. The base curve is defined by two or more nodes. You can click on any node and drag to modify it. You can also create additional nodes by clicking on a curve segment between two nodes. In order to remove a node drag it outside of the widget area.

Tip: If you intend to take full manual control of the tonal values with the *tone curve* module or the *zone system* module (see Section 3.4.7.3, “Tone curve” and Section 3.4.7.4, “Zone system”) it may be easier to leave the image in linear RGB. Disable the *base curve* module in this case.

3.4.6.4. Exposure

Overview

This module is used to tweak the exposure. It is directly linked to the histogram panel. Indeed, if you correct exposure graphically, using the *histogram* (see Section 3.3.5, “Histogram”), you automatically activate the exposure module. The histogram simply acts as a view for the exposure module.

Usage

This module is responsible for one of the most basic steps in each raw image development. An exposure adjustment value allows you - within certain limits - to correct for under- or overexposure. A shift by 1EV is equivalent to a change of exposure time by a factor of 2.

Positive exposure corrections will make the image brighter. As a side effect noise level gets higher. Depending on the basic noise level of your camera and the ISO value of your image, positive exposure compensations with up to 1EV or 2EV still give reasonable results.

Negative exposure corrections will make the image darker. Given the nature of digital images this can not correct for blown out highlights (see also Section 3.4.6.6, “Highlight reconstruction”).

A black level adjustment is a basic tool to increase contrast and pop of an image. The value defines at what threshold dark gray values are cut off to pure black. Use with care as the clipped values can not be recovered in other modules further down the pixelpipe. Please also have a look at the *tone curve* module (see Section 3.4.7.3, “Tone curve”) and the *levels* module (see Section 3.4.7.2, “Levels”) which can produce similar results with less side effects as they come later in pixelpipe.

black

Adjust the black level.

exposure

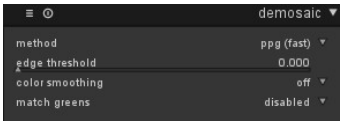
Adjust the exposure correction [EV].

auto

Calculate a correct exposition for the rectangular view appearing in the centre of the image. You can draw your own selection using your mouse. An adjustment slider right to the auto exposure checkbox lets you define what percentage of bright values are to be clipped out in the calculation.

3.4.6.5. Demosaic

Overview



This module allows you to control how the demosaic is processed.

Usage

Demosaic is an essential step of any raw image development process.

A detailed description would be beyond the scope of this manual. In a nutshell, the sensor cells of a digital camera are only able to record different levels of lightness, not different color. In order to get a color image, each cell is covered by a color filter, either in red, green or blue. Due to the color sensitivity of the human vision, there are two times more green cells than red or blue. Filters are arranged in a certain mosaic, called Bayer pattern. Therefore each pixel of your image originally only has information about one color channel. Demosaic reconstructs the missing color channels by interpolation with data of the neighboring pixels.

As all this is prone to produce artifacts, various different demosaic algorithms have been developed in the past. Artifacts would typically be visible as Moiree-like patterns when you strongly zoom into your image. Currently darktable supports PPG and AMAZE. Both algorithms produce high quality output with a low tendency to artifacts. AMAZE is reported to sometimes give slightly better results. However, as AMAZE is significantly slower, darktable uses PPG as a default.

Some further parameters of this module can activate additional averaging and smoothing steps. They might help to reduce remaining artifacts in special cases.

Demosaic is always applied when exporting images. Demosaic is done on monitor display only when zoom is greater than 50% or when the according preference setting “demosaicing for zoomed out darkroom mode” (see Section 6.2, “Core options”) is set accordingly. Else color channels are taken from neighboring pixels without an expensive interpolation.

method

Set the demosaic method. darktable currently supports PPG and AMAZE.

edge threshold

Set the threshold for an additional median pass. Defaults to “0” which disables median filtering.

color smoothing

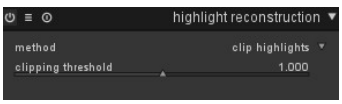
Activates a number of additional color smoothing passes. Defaults to “off”.

match greens

In some cameras the green filters have slightly varying properties. This parameter adds an aditional equilization step to suppress artifacts. Available options are “disabled”, “local average”, “full average” and “full and local average”.

3.4.6.6. Highlight reconstruction

Overview



This module tries to reconstruct color information that is usually clipped because of incomplete information in some of the channels. If you do nothing, your clipped areas are often toned to the not clipped channel. For example, if your green and blue channels are clipped, then your image will appear red in the clipped areas.

Usage

method

You can choose between two methods: clipping highlight or reconstructing in LCh. Clipping highlight analyses each pixel having at least one channel clipped. Then it sets all channels to the minium value found among the channels. Reconstruct in LCh analyses each pixel having at least one channel clipped and transforms the information in LCh color space to linearly mix the channels.

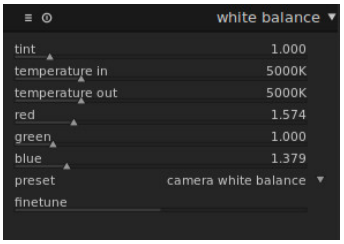
clipping threshold

Manually adjust the clipping threshold against magenta highlights. The default is usually satisfactory without any need for additional adjustments.

3.4.6.7. White balance

Overview

This module is used to set the white balance. You have three ways to interact with it: (a) Set up tint, temperature in and temperature out, (b) define the value of each channel, or (c) choose from predefined white balances.



Usage

tint

Alter the colour tint of the image, from magenta (value < 1) to green (value > 1). The channel sliders will be updated when you adjust this parameter.

temperature in/out

Set the temperature in and temperature out (in Kelvin). The channel sliders will be updated when you adjust this parameter.

red, green and blue channels

Set the channel values on a scale from 0 to 8.

preset

Select a preset white balance.

camera white balance (default)	White balance reported by the camera.
spot white balance	Select a square area in your image containing mostly grey pixels. The white balance is calculated based on the selected area.
passthrough	Show without adjusting for white balance.
camera presets	Camera specific white balance presets. Examples: direct sunlight, flash, cloudy, shade and a number of indoor lighting options.

finetune

Some cameras offer additional finetuning parameters if one of the camera presets is selected. Depending on camera white balance, can be adjusted in steps within a certain range. The adjustments are usually towards yellow (value < 1) or blue (value > 1).

3.4.6.8. Invert

Overview

The main purpose of this module is to invert scanned negatives.

Usage

color of film material

The only control element of this module is a color selector which allows to adjust for different colors of your film material. Clicking on the colored field will open a color selector dialog which allows to define a color in HSL or RGB color space. You can also activate a color picker by pressing and take a color probe from your image - preferrably from the unexposed border of your negative.

3.4.7. Tone group

This group contains modules that operate on the tonal values of an image, modulating brightness while leaving color values intact.

3.4.7.1. Fill light

Overview

This module allows local modification of the exposure based on pixel lightness.

Usage

Pushes exposure by increasing lightness with a Gaussian curve of a specified width, centered on a given lightness.

exposure

Sets fill-light exposure in [EV].

center

Sets the median lightness impacted by the fill-light. A color picker is activated by pressing . It shows the picked lightness value in the gradient bar, which helps find the desired center value.

width

Sets the width of the Gaussian curve. This number is expressed in zones, with the whole dynamic range being 10 zones. As the Gaussian curve is symmetric, only even numbers can be entered.

3.4.7.2. Levels

Overview

A tool for adjusting black, white, and mid-gray points. This module is especially useful if the histogram of an image does not span the whole horizontal range, from pure black to pure white.

Usage

The levels tool shows a histogram of the image, and displays three bars with handles. Dragging the handles modifies the tones in the image. Those bars control the black, middle gray and white points.

You can move the black and white bars to match the left and right border of the histogram, which will make the output image span the full available tonal range. A previously flat looking image will get more contrast and pop.

Moving the middle bar will modify the middle gray tones. Shifting it left will make the image look brighter, shifting it right will make it darker. This is often referred to as a change of image gamma.

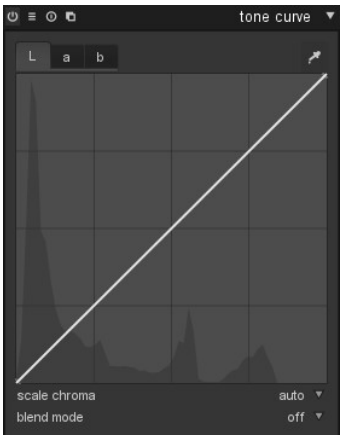
There are three color pickers in black, gray, and white, available by pressing the respectively colored icon. You can use them to sample the corresponding level directly from the image.

The “auto” button autoadjusts the black and white point and puts the gray point exactly in the mean between them.

3.4.7.3. Tone curve

Overview

This module is a classic digital photography tool. Unlike other image manipulation software, however, darktable's tone curve acts in Lab color space. Thus, is offers three independent curves for L, a, and b channels.



Usage

In its default state, curves will be straight lines, defined by few anchor nodes. You can move the nodes with your mouse to modify the curve. You can also generate new nodes

by clicking on the curve. Up to 20 nodes per curve can be defined. To remove a node, move it out of the widget area. A color picker is activated by pressing and will show the picked values in the graph. Numerical Lab values of input and output (see below) at the selected spot are shown on top left of the widget.

L-channel curve

The tone curve in L-channel works on Lightness. For a better overview a lightness histogram is displayed in the diagram.

The horizontal line represents the input image pixels' lightness. The vertical line represents the lightness of the output image pixels. A straight line does not change anything. A point above the default diagonal increases the lightness, whereas a point under decreases it. Shifting the center of the curve upwards will lighten the image, shifting it downwards will darken the image. An S-like curve will enhance the contrast of the image.

a/b-channel curves

The curves in the a and b channels work on color values. They are only displayed and active if the *scale chroma* combobox is set to “manual”. The horizontal line represents the color channel value of the input image pixels. The vertical line represents the color channel value of the output image pixels. Positive a-values correspond to more magenta colors; negative a-values correspond to more greenish colors. Positve b-values correspond to more yellowish colors; negative b-values correspond to more blueish colors.

A straight line does not change anything. Shifting the center of the curve will give the image a color tint: shifting a-channel upwards gives a magenta tint; shifting b-channel upwards gives a yellow tint; shifting a-channel downwards gives a green tint; shifting b-channel downwards gives a blue tint.

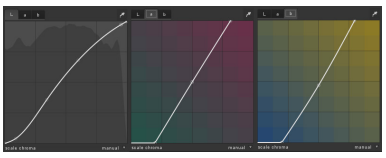
Increasing/decreasing the steepness of the curves, without shifting its center, will increase/decrease the color saturation of the respective channel. With properly defined curves you can exert fine control on color saturation, depending on the input pixel's colors.

scale chroma

darktable does an automatic adjustment of color saturation, if this combobox is set to “auto”. The level of this adjustment depends on the pixel's color values and its L-value modification by the L-channel tone curve. It is designed to give an overall boost in color saturation, if the L-curve gives a contrast boost. Look at blend mode “coloradjustment” to adjust the strength of this effect (see Section 3.4.2, “Blending operators”). If this combobox is set to “manual”, you can modify color saturation using the curves in channels a and b.

Examples

Original image



Tone curve settings. Please note how the center node of our b-curve was shifted down to negative values. This gives the image its blue tint.

Resulting image

3.4.7.4. Zone system

Overview

This module is another way to change the lightness of your image, based on Ansel Adams' system. It allows modification of a zone's lightness taking into account the effect on the adjacent zones. It divides the lightness range into a user-defined number of zones.

Usage

The lightness is processed on the L channel from Lab.

The center view shows the image broken down in zones.

When hovering above a zone on the lightness scale, that zone is highligthed on the preview. The number of zones can be changed by mouse-scrolling on the lightness scale.

Left click and drag a handle in the zonebar to modify the zonemapping, use right click to remove a controlpoint.

Examples

The original image.

Here, the darker and lighter zones were compressed to increase contrast, then the upper parts of darker zones were expanded to increase their visual impact.

3.4.7.5. Local contrast

Overview

This module allows enhancing local contrast. It uses the unnormalized bilateral filter, and works on the L channel from Lab.

Usage

Local contrast boosts details of your image, much like the *equalizer* does (see Section 3.4.9.2, “Equalizer”). However, it is easier to use as it does not require you to work on different frequency bands.

coarseness

Make the details you want to adjust finer or coarser.

contrast

How strongly the algorithm distinguishes between brightness levels. Increasing the value results in a more contrasty look.

detail

Add or remove detail. Higher values will increase local contrast.

Example

Before

After, a little overdone to demonstrate the effect. Use this sparingly, to avoid a cheap, overprocessed look.

3.4.7.6. Tonemapping

Overview

This module compresses the tonal range of HDR images, so they fit into the limits of a normal, low dynamic range image, using Durand's 2002 algorithm. darktable can import HDR images if they come in OpenEXR, RGBE or PFM format or as a DNG generated by darktable's HDR creation mechanism (see Section 2.2.7, “Selected image(s)”).

Usage

The underlying algorithm uses a bilateral filter to decompose an image into a coarse base layer and a detail layer. The contrast of the base layer is compressed, while the detail layer is preserved, then both layers are re-combined.

contrast compression

Sets the compression level of the contrast layer to fit the lower dynamic range.

spatial extent

Sets the spatial extent of the bilateral filter. Lower values cause the contrast compression to have stronger effects on image details.

3.4.7.7. Global tonemap

Overview

This modules implements another approach to compressing the tonal range of an HDR image into the limited tonal range of a typical LDR output file. It offers several implementations of global tonemap operators.

Usage

Global tonemapping processes each pixel of an HDR image, without taking the local surrounding into account. This is generally faster than local tonemapping, as implemented in the *tonemapping* module but might lead to less convincing results with very high dynamic scenes. As an enhancement to the original operators, darktable can preserve detail of the input image, and transfer it back to the output image.

operator

Reinhard, Filmic and Drago global tonemap operators are available for use. Depending on the selected operator, different parameters can be adjusted. Some operators are fully self-adjusting, and do not require specific controls.

bias

Only offered for the *Drago* operator. This parameter influences the contrast of the output image. It is an essential parameter for adjusting the compression of high values and the visibility of details in dark areas. According to the original paper, a value of 0.85 is recommended as a starting point.

target

Only offered for the *Drago* operator. This is a scale factor to adjust the global image brightness to the brightness of the intended display. It is measured in cd/m^2, and should match the according value of your output device. Higher values lead to a brighter image, while lower values lead to a darker image.

detail

Offered as an addition to all operators. This parameter controls how much detail is preserved and transfered back into the output image after tonemapping.

3.4.8. Color group

Modules for working specifically on your image's color are found here in the color group.

3.4.8.1. Velvia

Overview

The velvia module enhances image saturation. Its effect is tailored to increases saturation less on lower saturated pixels than on highly saturated pixels.

Usage

strength

This slider controls the strength of the effect.

mid-tones bias

Velvia selectively reduces its effect for mid-tones to avoid unnatural skin tones. This slider controls that selectivity, reducing its value reduces mid-tone protection and makes the overall velvia effect stronger.

3.4.8.2. Channel mixer

Overview

This module is a powerful tool to manage channels. It accepts red, green and blue channels as an input. As output it provides red, green, blue, gray, hue, saturation and lightness channels.

Usage

First select your output channel and then set the amount each input channel feeds into that output channel.

Examples

For skin tones the blue channel tends to represent detail, with red tending to also have smoother tones than green. Therefore tonal rendering is controlled by how we blend of the three input channels.

Here a monochrome portrait is produced by simply selecting the grey channel as output. A smooth skin tone is achieved by reducing the blue channels input and also emphasising the red channels input relative to green. An RGB mix of 0.9, 0.3, -0.3 was used together with an 0.1 EV exposure increase to lighten the image.

In this example an RGB mix of 0.4, 0.75, -0.15 uses more green than red, bringing back some features. We still reduce the blue channel in the mix to de-emphasize unwanted skin texture.

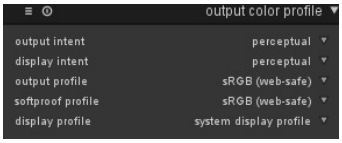
Table of mixing values for some b/w films

Classic black and white films have different characteristic color responses. Select gray as output mixing channel, and try out the values suggested below for your favourite film type.

Film Type	Red	Green	Blue
AGFA 200X	0.18	0.41	0.41
Agfapan 25	0.25	0.39	0.36
Agfapan 100	0.21	0.40	0.39
Agfapan 400	0.20	0.41	0.39
Ilford Delta 100	0.21	0.42	0.37
Ilford Delta 400	0.22	0.42	0.36
Ilford Delta 3200	0.31	0.36	0.33
Ilford FP4	0.28	0.41	0.31
Ilford HP5	0.23	0.37	0.40
Ilford Pan F	0.33	0.36	0.31
Ilford SFX	0.36	0.31	0.33
Ilford XP2 Super	0.21	0.42	0.37
Kodak T-Max 100	0.24	0.37	0.39
Kodak T-Max 400	0.27	0.36	0.37
Kodak Tri-X 400	0.25	0.35	0.40
Normal Contrast	0.43	0.33	0.30
High Contrast	0.40	0.34	0.60
Generic B/W	0.24	0.68	0.08

3.4.8.3. Output color profile

Overview



This module manages the output profiles for display and export as well as the rendering intent to be used when mapping between the different color spaces.

darktable comes with pre-defined profiles sRGB, AdobeRGB, XYZ and linear RGB but you can provide additional profiles by placing these in $DARKTABLE/share/darktable/color/out and $HOME/.config/darktable/color/out. $DARKTABLE is used here to represent your chosen darktable installation directory and $HOME your home directory.

Usage

A configuration parameter “always use littlecms2 during export” in darktable's *core options* (see Section 6.2, “Core options”) defines how darktable renders colors for display and export. If the configuration parameter is disabled darktable uses a simplified and very fast internal rendering algorithm. If the option is checked the external library *littlecms2* [http://www.littlecms.com/] with higher accuracy and significantly higher processing overhead is used instead.

output intent

Sets the rendering intent for output/export. You can easily override this setting whenever you do exports from lighttable mode.

perceptual	Suited to pictures as it maintains the relative position of colors. This is usually the best choice.
relative colorimetric	Out-of-gamut colors are converted to colors having the same lightness, but different saturation. Other colors remain unmodified.
saturation	Saturation is kept but lightness is slightly changed.
absolute colorimetric	Keep the white point.

Only rendering with littelcms2 gives you a choice of rendering intent. The option is hidden, if darktable's internal rendering routines are used.

display intent

Sets the rendering intent for your display. See above for available options.

output profile

Sets the color profile for output/export.

softproof profile

Sets the color profile for softproofing.

softproof allows you to preview your image rendered using a printer profile so as to see how colors will end up on the final print and is toggled on and off by pressing “s”.

Likewise *gamut check* is toggled on and off by pressing “g”; this function highlights in cyan all pixels out of gamut with respect to the selected softproof profile.

softproof and *gamut check* are mutually exclusive modes that can be activated at any place in darkroom mode and when this module is in focus the status is displayed below the image.

Tip: at other times it may not be obvious if the *softproof* mode is still active or not. If in doubt press “g”, which will switch to *gamut check* and be clearly distinguishable by cyan marked pixels. Press “g” again and you are back to normal display mode.

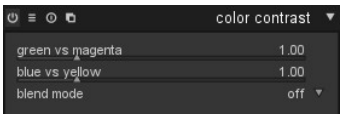
display profile

Sets the color profile for the display. The additional option “system display profile” is taken directly from your system's color manager or from your X Server. When working with a calibrated display this is the preferred option to choose.

3.4.8.4. Color contrast

Overview

The color contrast module provides simplified control for changing the contrast or separation of colors between either green/magenta or blue/yellow axis.



Usage

Higher values increase color contrast, lower values decrease it. The effect of this module's sliders are similar to applying a steepened or flattened a- or b-curve in module *tone curve* (see Section 3.4.7.3, “Tone curve”).

green vs. magenta

Changes color contrast for green versus magenta.

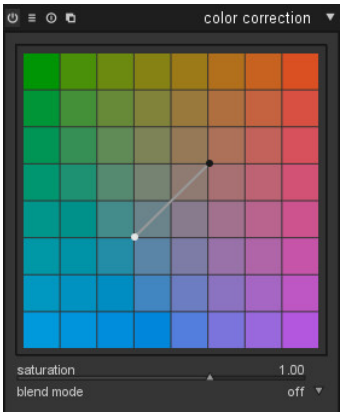
blue vs. yellow

Changes color contrast for blue versus yellow.

3.4.8.5. Color correction

Overview

This module can be used to modify the global saturation, give a tint to the image or to split tone it.



Usage

color board

For split toning drag the white dot to the desired highlight tint and then select a tint for shadows with the dark spot. For a simple global tint set both spots to the same color.

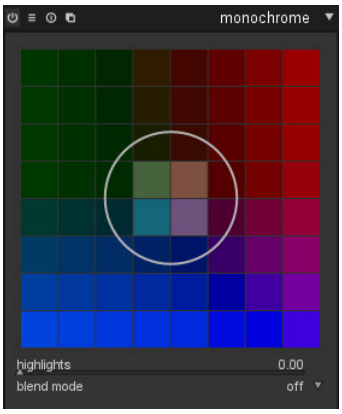
saturation

Use the saturation slider to correct the global saturation.

3.4.8.6. Monochrome

Overview

This module is a quick way to convert an image into black and white and provides a variable color filter for that conversion.



Usage

The default central location of the filter has a neutral effect but dragging it to an alternate position applies a filter analogously to taking a b&w photograph through a conventional color filter.

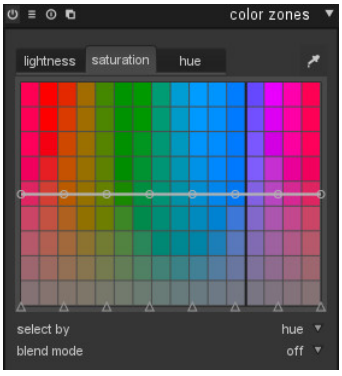
As well as position you can change the filter size by scrolling with your mouse wheel. This makes the filter's range of hues more or less selective.

Tip: First reduce the filter size to concentrate its effect and move it across the hue pallet to find the best filter value for your desired image rendition. Then expand the filter to include more hues and thus more natural tonality.

3.4.8.7. Color zones

Overview

This module selectively modifies the colors in your image. It is highly versatile and allows every transformation possible in the LCh colorspace.



Usage

The horizontal axis represent the different values you can modify. The vertical axis shows the changes you can achieve.

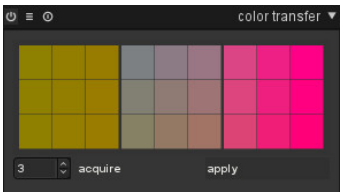
For both horizontal and vertical axises you can work on lightness, saturation or hue. A color picker is activated by pressing and will show the picked values in the diagram.

You can click on any of the eight nodes on the curve and drag to adjust it vertically. A circle indicates how strong adjacent nodes will be affected. Use the scroll wheel of your mouse to change the circle diameter. You can also use the eight controlpoints (triangles which define the vertical value of the nodes) at bottom to adjust the curve.

3.4.8.8. Color transfer

Overview

This module transfers color characteristics from one image to another. In its current version this module has several issues and is not recommended for use - a rework is under way.



Usage

number of clusters

Set the number of color clusters to use.

acquire

Press this button to acquire colors from the source image.

apply

Press this button to impose the colors to the destination image.

3.4.8.9. Vibrance

Overview

Vibrance is a widely used term in image processing but the mechanism and end result differ from program to program. Vibrance in darktable saturates and brings down the lightness of the most saturated pixels to make the colors more vivid.

Usage

Vibrance only has one parameter which controls the amount applied.

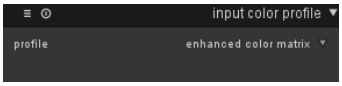
vibrance

The amount of vibrance to apply to the image.

3.4.8.10. Input color profile

Overview

This module can be used to override darktable's automatic allocation of input color profile if there is an alternative that more closely matches your original image's color space.



Usage

profile

Choose the profile or color matrix to apply, darktable offers many widespread matrices along with an enhanced matrix for some camera models. The enhanced matrices were processed by the darktable team in order to provide a look closer to the manufacturer's.

You can also supply your own input ICC profiles and put them into $DARKTABLE/share/darktable/color/in or $HOME/.config/darktable/color/in. $DARKTABLE is used here to represent darktable's installation directory and $HOME your home directory. One common source of ICC profiles is the software that is shipped with your camera; it often contains profiles specific to your camera model. You may need to activate module *unbreak input profile* (see Section 3.4.8.11, “Unbreak input profile”) to use your extra profiles.

If your input image is a low dynamic range file like JPEG, or a raw in DNG format, it might already contain an embedded ICC profile which darktable will use as a default. You can always overrule darktable and select a different profile or select “embedded icc profile” to restore the default.

3.4.8.11. Unbreak input profile

Overview

This module adds a correction curve to image data, which is required if you have selected certain input profiles in module *input color profile*.

Usage

If you decide in module *input color profile* to use an ICC profile from the camera manufacturer, a correction curve very frequently needs to be pre-applied to image data - or else the final output looks much too dark. This extra processing is not required if you use darktable's standard or enhanded color matrices. The correction curve is defined with a linear part extending from the shadows to some upper limit and a gamma curve covering mid-tones and highlights. For further reading please also have a look at darktable's neighbouring project UFRaw [http://ufraw.sourceforge.net].

linear

Set the upper limit for the region counted as shadows and where no gamma correction is performed. Typically values between 0.0 and 0.1 are required by the profile.

gamma

Set the gamma value to compensate your input profile. Often the required value is 0.45 (the reciprocal of 2.2 gamma used by some manufacturer's profile).

3.4.9. Correction group

The correction group contains the modules that will correct typical problems in an photo such as hotpixels, spot removal, noise, lens correction among others. This group also includes the basic sharpening tools.

3.4.9.1. Sharpen

Overview

This is an standard UnSharp Mask (USM) tool for sharpening the details of an image.

Usage

This module works by enhancing the contrast around edges and thereby enhances the impression of sharpness of an image. In darktable this module is only applied to the L-channel in Lab color space.

radius

USM applies a gaussian blur to your image as part of its algorithm. This controls the blur radius which in turn defines the spatial extent of edge enhancement. Too high values will lead to ugly over-sharpening.

amount

This controls the strength of the sharpening.

threshold

Contrast differences below this threshold are excluded from sharpening. Use this to avoid amplification of noise.

3.4.9.2. Equalizer

Overview

This versatile module can be used to achieve a variety of effects, such as: bloom, denoising, and local contrast enhancement. It works in the wavelet domain and parameters can be tuned for each frequency band separately.

Usage

Each frequence band can be tweaked independently. In particular, you can adjust contrast boost and denoise threshold splines for both lightness and chromaticity, as well as the sharpness of the wavelet basis on each frequency scale.

Each spline can be dragged with a proportional edit approach; use the mouse wheel to adjust the radius in which your changes will have an effect. The transparent area indicates where you would drag the spline with the current mouse position and radius. The small little triangles on the x-axis can be moved to alter the x-position of the spline nodes.

Drag the upper line (bright circles, here for the lightness channel) to affect local contrast. Pulling it up, as shown here, will result in a contrast boost for that frequency band. Higher frequencies, i.e. smaller details, are to the right of the grid. Pulling it down works, too.

The bottom spline (black circles) is used to perform denoising. It adjusts the wavelet shrinkage threshold for each frequency band. Pull it up to see the effect. In this example, the noise which has been amplified by local contrast enhancement is removed.

This screen shows the effect of the sharpen parameter. It is here pulled down to zero for all bands. This is effectively a regular à trous wavelet, without edge detection, and results in the characteristic halos around sharp edges in the image.

This image is the other extreme. The wavelet basis now oversharpens, which results in ugly gradient reversals near the ridge of the rock.

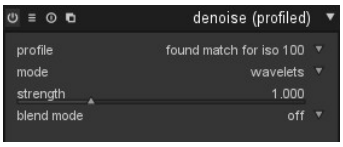
Note that the sharpness parameter only affects the wavelet basis, not the image directly. You will have to change some denoise/contrast boost parameters to see an effect following adjustments to the sharpness parameter.

This module additionally has a “mix” slider below the spline GUI. Adjusting the slider will upscale or downscale the splines on the y-axis. The slider was added as a convenience tool to help you modify the strength of the effect. It is not a module parameter in itself; when you leave darkroom mode all changes will be consolidated into the spline curves.

Have a look at the presets where there are a broad variety of examples that will provide a good starting point to gain an intuitive understanding of the controls.

3.4.9.3. Denoise - profiled

Overview



This module offers an easy to use and - at the same time - highly efficient denoise operation. Under the hood it applies (your choice of) a non-local means or edge-aware wavelet denoise algorithm with parameters specifically profiled for certain camera models and ISO settings.

Usage

The darktable team, with the help of many users, has measured noise profiles for various cameras. Differentiated by ISO settings we evaluated how the noise statistics develop with brightness for the three color channels. Our set of profiles already covers over 70 popular camera models from different manufacturers.

profile

Based on EXIF data of your RAW file, darktable will automatically determine the camera model and ISO setting. If found in its database, the corresponding noise profile will be used. If your image has an intermediate ISO value, the statistical properties will be interpolated between the two closest datasets in the database, and this interpolated setting will show up as the first line in the combo box. You also have the option to manually overwrite this selection to suit your personal preferences better. The top-most entry in the combo box brings you back to the profile darktable deems most suited.

mode

This module can eliminate noise with two different core algorithms. “non-local means” is a bit better suited to tackle luma (lightness) noise; “ wavelet” has its strength in eliminating chroma (color) noise. If needed you can apply two instances of this module (see Section 3.4.5, “Multiple instances”). The “non-local means” instance should be combined with blend mode “lightness”; the “wavelet” instance with blend mode “color”. For more information on blend modes have a look at Section 3.4.2, “Blending operators”.

patch size

This slider is only available if mode “non-local means” is selected. It controls the size of the patches being matched when deciding which pixels to average (see also Section 3.4.9.4, “Denoise - Non local means”). Setting this to higher values can give more sharpness. Processing time will stay about the same.

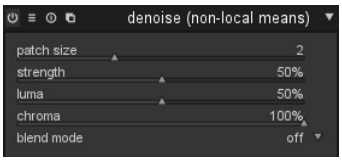
strength

This parameter is here to fine-tune the strength of the denoise effect. The default value has been chosen to maximize the peak signal to noise ratio. It's mostly a matter of taste if you prefer a rather low noise level at the costs of a higher loss of detail, or if you accept more remaining noise in order to have finer structures better preserved within your image.

3.4.9.4. Denoise - Non local means

Overview

This is a denoise algorithm, which will work on chroma and/or luma.



Usage

This module reduces noise in your image but preserves structures. This is accomplished by averaging a pixel with other pixels in the image. The weight of averaging depends on the similarity of the surrounding pixel's neighborhood with the neighborhood of the one pixel to be denoised. A patch with a certain size is used to measure that similarity. As denoising is a resource hungry process, it slows down pixelpipe processing significantly; consider activating this module late in your workflow.

patch size

The radius of the patch for similarity evaluation.

strength

The strength of the denoise. Higher values lead to a stronger effect.

luma

Amount of denoise to apply to luma. Select carefully in order not to lose too much structure.

chroma

Amount of denoise to apply to chroma. You can be much more aggressive with this parameter compared to luma.

3.4.9.5. Denoise - bilateral

Overview

This module is used to denoise high ISO pictures. It is flagged as a slow module due to its high resource consumption, both in terms of CPU cycles and in terms of memory usage. Quite counter-intuitively, the greater the values for sliders, the lesser resources.

Usage

This module reduces noise in your image but preserves sharp edges. This is accomplished by averaging pixels with their neighbors, taking into account not only the geometric distance but also the distance on the range scale, i.e. differences in color intensities. As denoising is a resource hungry process, it slows down pixelpipe processing significantly; consider to activate this module late in your workflow.

radius

Set the spatial extent of the gaussian blur.

red

Blur intensity for red channel.

green

Blur intensity for green channel.

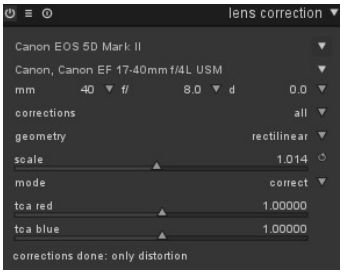
blue

Blur intensity for blue channel.

3.4.9.6. Lens correction

Overview

This module is able to correct certain lens flaws, namely distortions, transversal chromatic aberrations (TCA) and vignetting. It relies on the external library lensfun, which comes with correction profiles for many (but not all) common cameras end lenses.



Usage

In order to perform lens corrections the module uses EXIF data of your image to identify the specific camera/lens combination and collects the needed correction parameters from a profile in lensfun's database.

camera

The camera make and model as determined by EXIF data. You can overrride this manually and select your camera from a hierarchical menu.

lens

The lens make and model as determined by EXIF data. You can overrride this manually and select your lens from a hierarchical menu. This is mainly needed for pure mechanical lenses.

photometric parameters

Corrections additionally depend on certain photometric parameters that are read from EXIF-data: focal length (needed for distortion, TCA, vignetting), aperture (needed for TCA, vignetting) and focal distance (needed for vignetting). Many cameras do not record focal distance in their EXIF data; most likely you need to set this manually.

You can manually override all automatically selected parameters. Either take one of the predefined values from the pull-down menu; or - with the pull-down menu still open - just type in your own value.

If no controls for the three photometric parameters are displayed, this means that no correction profile could be found for the automatically identified camera/lens combination. You may try to find the right profile yourself by searching for it in the menu. If there is no matching profile, visit lensfun's home page [http://lensfun.berlios.de] and learn how to generate your own set of correction parameters. Don't forget to share your profile with the lensfun team!

corrections

This combobox gives you a choice about which corrections (out of distortion, TCA and vignetting) darktable shall apply. Change this from its default “all”, if your camera has already done some internal corrections (e.g. of vignetting), or if you plan to do certain corrections with a separate program.

geometry

In addition to the correction of lens flaws, this module can change the projection type of your image. Set this combobox to the aimed projection type, like “rectilinear”, “fish-eye”, “panoramic”, “equirectangular”.

scale

This slider allows you to adjust the scaling factor of your image. Pressing the auto scale button (right to the slider) will let darktable find the best fit to avoid black corners.

mode

The default behavior of this module is to correct lens flaws. Switch this togglebutton to “distort” in order to simulate the behavior of a specific lens (inverted effect).

TCA red

This slider allows to override the correction parameter for TCA. You can also use this slider to manually set the parameter in case the lens profile does not support TCA correction. Look out for colored seams at features with high contrast edges and adjust this parameter and the following one to minimize those seams.

TCA blue

This slider allows to override the correction parameter for TCA. You can also use this slider to manually set the parameter in case the lens profile does not support TCA correction.

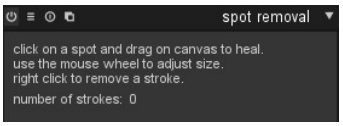
corrections done

You will sometimes observe that for a given camera/lens combination only part of the possible corrections (distortion, TCA, vignetting) are supported by lensfun's profiles. This message box will tell you what corrections have actually been applied.

3.4.9.7. Spot removal

Overview

Spot removal allows you to correct a zone in your image by using another zone as model.



Usage

Click on the area to be healed then drag to a clean area. The circle with line inside is the clean reference area.

Use mouse scroll-wheel over the stroke to change its size.

Right-click on a stroke to delete it.

Close the module to complete the changes.

Examples

Let's use this portrait as example; we want to remove some dirt and unwanted catchlight from camera pop-up strobe.

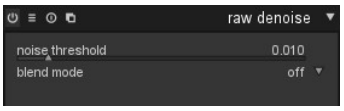
I have marked all the spots that I want to remove from the image. I first start by pressing the spot and drag to an area that should be used as reference to remove the spot. The handle (circle) which has a line that ends in the center of the circle indicates reference source.

And here is the result image of the spotremoval.

3.4.9.8. Raw denoise

Overview

Raw denoise allows you to perfom denoising on pre-demosaic data. It is ported from *dcraw* [http://www.cybercom.net/~dcoffin/dcraw/].



Usage

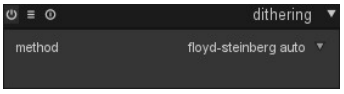
noise threshold

Set the threshold for noise detection. Higher values lead to stronger noise removal and higher loss of image detail.

3.4.9.9. Dithering

Overview

This module eliminates some of the typical banding artifacts which can occur, when darktable's internal 32-bit floating point data are transfered into a discrete 8-bit or 16-bit integer output format for display or file export.



Banding is a problem which can arise, when an image is downsampled into a lower bit-depth. Downsampling happens regularly, when darktable displays or exports the results of a pixelpipe. In order to avoid banding, you may activate this modules.

If you export images with a reduced width/height and want best dithering results, please make sure that you de-activate option “do high quality resampling during export” in *core options* (see Section 6.2, “Core options”), else the final scaling step will counteract dithering.

Some users might want to avoid the additional processing overhead of this module while doing interactive work - knowing that the final results in file output is good enough :) In that case there is a configuration parameter “dithering for darkroom mode” in *core options* (see Section 6.2, “Core options”). Dithering of center view in darkroom mode is skipped if this config parameter is de-activated. This is a speed-up option, which only affects monitor display, not file exports.

Viewing from some distance an image dithered into a very low bit depth (like “floyd-steinberg 1-bit b&w”) will give the impression of a homogeneous grayscale image. We try to mimic this impression in darktable when you look at zoomed-out images in the center view, in the navigation window and for thumbnails. This is accomplished by dithering those images into a higher number of grayscale levels. Note that as a consequence the histogram - which is derived from the navigation window - will show this increased number of levels and is no longer a full match of the output image.

Usage

method

This combobox sets the dithering method. Floyd-Steinberg error diffusion - with some typical output bit depths - and random noise dithering are both supported. Floyd-Steinberg systematically distributes quantization errors over neighboring pixels, whereas random dithering just adds some level of randomness to break sharp tonal value bands. The default setting is “floyd-steinberg auto”, which automatically adapts to the desired output format.

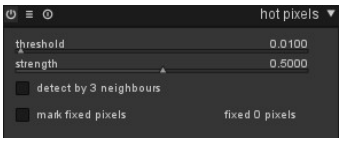
damping

This slider is only displayed if you choose method “random”. It controls the level of added random noise expressed as a damping factor in dB. A value of -80dB is a good fit for 8-bit output formats.

3.4.9.10. Hotpixels

Overview

This module is able to automatically detect and eliminate hotpixels. Hotpixels are pixels which failed to record light level correctly. Detected hotpixels are replaced by an average value of their neighbors.



Usage

You control the detection sensitivity with the threshold parameter and the level of elimination with the strength parameter.

threshold

The threshold of the detection, i.e. how strong a pixel's value needs to deviate from its neighbors to be regarded as a hotpixel.

strength

The strength of blending hotpixels with their surrounding.

detect by 3 neighbours

This will extend the detection of hotpixels, it will even regard a pixel as hot if a minimum of only three (instead of four) neighbor pixels deviate by more than the threshold level.

mark fixed pixels

This options will mark the pixels that have been corrected. It also displays the count of detected and fixed pixels.

3.4.9.11. Chromatic aberrations

Overview

This module allows you to correct chromatic aberrations.

Usage

The module has no parameters. On activation it will automatically try to optimize away visible CA's.

The underlying model assumes as input an uncropped photographic image. The module is likely to fail when you zoom into the image, as in that case it will only receive part of your photograph as input. As a consequence, chromatic aberrations do not get corrected properly. This limitation only applies to interactive work, not to file export.

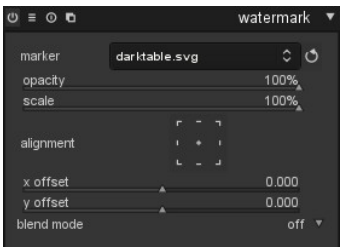
3.4.10. Effect group

In the effect group you will find modules with a more artistic touch.

3.4.10.1. Watermark

Overview

The watermark module provides a way to render a vector-based overlay onto your image. Watermarks are standard SVG documents and can be designed using *Inkscape* [http://www.inkscape.org].



The SVG processor of darktable also substitutes strings within the SVG document, which gives the opportunity to include image dependent information in the watermark such as aperture, exposure time and other metadata.

User-designed watermarks are placed into the directory $HOME/.config/darktable/watermarks. Once in place, use the reload button at the right of the watermark file name to update the list of available watermarks to use.

Here follows a list of available variable strings that is supported for substitution within the svg document.

$(DARKTABLE.NAME)	The application name
$(DARKTABLE.VERSION)	The version of darktable
$(IMAGE.ID)	The unique image id within current library
$(IMAGE.FILENAME)	The image filename
$(IMAGE.EXIF)	The image exif string
$(EXIF.DATE)	The image date
$(EXIF.DATE.SECOND)	Seconds from the image EXIF data
$(EXIF.DATE.MINUTE)	Minutes from the image EXIF data
$(EXIF.DATE.HOUR)	Hours from the image EXIF data (24h)
$(EXIF.DATE.HOUR_AMPM)	Hours from the image EXIF data (12h, AM/PM)
$(EXIF.DATE.DAY)	Day from the image EXIF data
$(EXIF.DATE.MONTH)	
$(EXIF.DATE.SHORT_MONTH)	
$(EXIF.DATE.LONG_MONTH)	
$(EXIF.DATE.SHORT_YEAR)	Abbreviated year from the image EXIF data (2013 is "13")
$(EXIF.DATE.LONG_YEAR)	Full year from the image EXIF data
$(DATE)	Current system date
$(DATE.SECOND)	Current system time seconds
$(DATE.MINUTE)	Current system time minutes
$(DATE.HOUR)	Current system time hours (24h)
$(DATE.HOUR_AMPM)	Current system time hours (12, AP/PM)
$(DATE.DAY)	Current system time day
$(DATE.MONTH)	Current system time month
$(DATE.SHORT_MONTH)	
$(DATE.LONG_MONTH)	
$(DATE.SHORT_YEAR)	Current system time year (abbreviated)
$(DATE.LONG_YEAR)	Current system time year
$(EXIF.MAKER)	The maker of camera model
$(EXIF.MODEL)	The camera model
$(EXIF.LENS)	The specific lens used
$(Xmp.dc.creator)	The creator string
$(Xmp.dc.publisher)	The publisher string
$(Xmp.dc.title)	The title of the image
$(Xmp.dc.description)	The description of the image
$(Xmp.dc.rights)	The rights assigned to the image

Usage

marker

Choose the watermark of interest. You can use the reload button next to the combobox to update the list with all newly added watermarks.

opacity

Set the opacity of the render of watermark.

scale

Scale the watermark pixel-independently.

alignment

Use these controls to align the watermark to any edge or center of the image.

x offset

Pixel-independent offset realtive to the choice of alignment on x-axis.

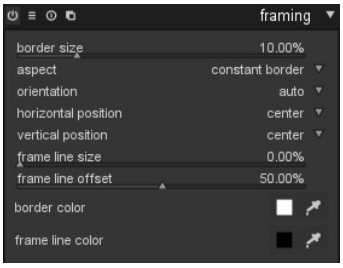
y offset

Pixel-independent offset relative to the choice of alignment on y-axis.

3.4.10.2. Framing

Overview

This module is an artistic feature to generate a frame around your image. The frame consists of a border with a user defined color and a frame line inside that border, which has another user defined color. There are various options for you to control the geometry of your frame.



Usage

border size

This slider controls the size of the frame in percent of the underlying full image.

aspect

With this combobox you can choose between different aspect ratios for the final output of this module, i.e. underlying image plus frame.

orientation

If you select a non-square aspect ratio this combobox defines the orientation - portrait or landscape. Set to “auto” if you want darktable to select the most reasonable orientation based on the underlying image.

horizontal position

Select from a set of pre-defined ratios where you want your underlying image be positioned on the horizontal axis. You can also right click and enter your own ratio as “x/y”.

vertical position

Select from a set of pre-defined ratios where you want your underlying image be positioned on the vertical axis. You can also right click and enter your own ratio as “x/y”.

frame line size

The percentage of the frame line size relative to the border size at its smallest part.

frame line offset

Where the frame line is positioned relative to the underlying image. Select a value of 0 for a frame line touching the image, 100% for a frame line touching the outer border limits.

border color

Clicking on the colored field will open a color selector dialog which allows one to define a color for the border in HSL or RGB color space. You can also activate a color picker by pressing and take a color probe from your image.

frame line color

Clicking on the colored field will open a color selector dialog which allows one to define a color for the frame in HSL or RGB color space. You can also activate a color picker by pressing and take a color probe from your image.

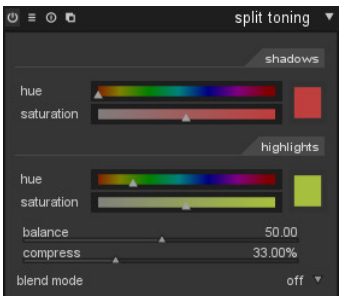
Examples

Example image with a user defined frame.

3.4.10.3. Splittoning

Overview

darktable's splittoning method creates a two color linear toning effect where the shadows and highlights are represented by two different colors. In the example image below you can see an original black and white image and one where a splittoning effect is applied with blue in shadows and a yellowish color in highlights.



Compared to traditional splittoning our module has more parameters to influence its behavior. We have parameter “balance”, which offsets the 50% gray level in your image - at your choice - more to the shadows or more to the highlights. Additionally, with parameter “compression” you can compress toning in the shadows and highlights and leave a gap in the mid-tones, which remain untouched by the effect.

The splittoning module does not convert images to black and white and has limited benefits on color images. So, if you want to do traditional splittoning, use the *monochrome* module (see Section 3.4.8.6, “Monochrome”) to make the image black and white before playing around with splittoning effect.

Usage

shadows and highlights color

These controls are used to set the color of the splittoning effect, you select the desired color and saturation for both shadows and highlights, you can also click the color preview box to bring up a common color picker dialog.

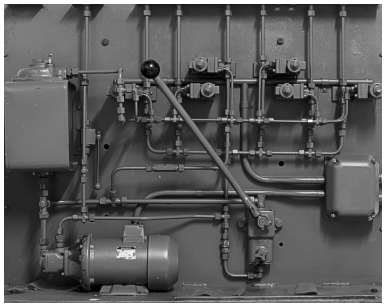
balance

This parameter represents the ratio of toning between shadows and highlights. For a value of 50% half of the lightness range in image is used for shadows toning and the other half for highlights toning.

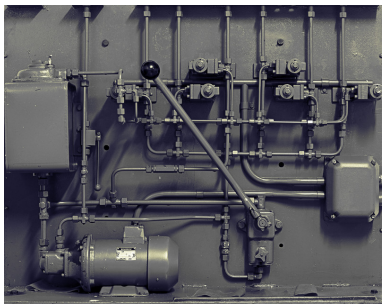
compression

Compression is a percentage of total lightness range that is not affected by color toning. Default value is set to 33%; this is not the default behaviour of an original splittoning which would be 0% compression. The choice of 33% as a default is to invite you expriment-ing with these parameters and how it extends the original splittoning method.

Examples



Original black and white image.

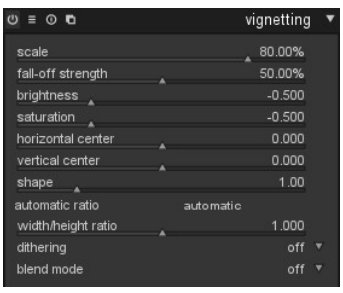


Splittoning with blue shadows and yellow highlights.

3.4.10.4. Vignetting

Overview

This module is an artistic feature which creates vignetting (modification of the brightness/saturation at the borders).



Usage

The vignetting module has an extensive set of parameters to precisely tune its effect. It also will display graphical controls within the image if the module is in focus. Give it a try to get a feeling of how it works. On-screen controls and parameter sliders will stay in sync.

scale

Set the radius of the vignetting area.

fall-off strength

Sets the progressiveness of the fall-off. Higher values will cause a steeper transition.

brightness

Sets the intensity of brightening (positive values) or darkening (negative values).

saturation

Controls how strong colors become when desaturated or saturated in the darkened or brightened vignetting area.

horizontal center

Shifts the center of the vignetting area horizontally.

vertical center

Shifts the center of the vignetting area vertically.

shape

Influences the shape of the vignetting area. The default value of 1 causes a circular or elliptical area. Smaller values will shift the shape into a more square one; higher values turn it into a cross-like shape.

automatic ratio

Click this button to automatically adjust the width/height ratio of the vignetting area to the aspect ratio of the underlying image. The vignetting area will typically become elliptical.

width/height ratio

Manually adjust the width/height ratio of the vignetting area.

dithering

With this combobox you can activate random noise dithering to overcome banding artifacts caused by vignette gradients. Select “8-bit output” to prevent banding on monitor display and for JPEGs. When set to “16-bit output”, only a little dithering will be applied, just strong enough to compensate for banding on the fine grained 16-bit level. This feature is mostly obsoleted by our new module *dithering* (see Section 3.4.9.9, “Dithering”).

Examples

An image with vignetting and with graphical vignetting controls displayed.

3.4.10.5. Soften

Overview

This module is an artistic feature that creates a softened image, commonly known as the Orton effect.

Usage

Michael Orton achieved his results on slide film by using two exposures of the same scene: one well exposed and one overexposed; he then used a darkroom technique to blend those into a final image where the overexposed image was blurred.

This module is almost a copy of Orton's analogue process into the digital domain. You can control brightnesss and blur with the provided parameters; we also add a control for saturation of the overexposured image for more play.

size

Set the size of blur of the overexposed image in the process, the bigger the softer.

saturation

Set the saturation of the overexposed image.

brightness

Expressed in [EV], the brightness slider selects the increase in brightness.

mix

Controls the mix of the overexposed image and the overall effect.

Examples

This is the original image, use it as reference for the changes below...

In this image I used the default values, and added 0.33EV to brightness for a little more light in the soft layer.

This version is the same as above but with 25% saturation.

3.4.10.6. Grain

Overview

This module is an artistic feature which simulates the grain of a film.

Usage

The grain is processed on the L channel from CIELAB.

coarseness

Set the grain size, which has been scaled to simulate an ISO number.

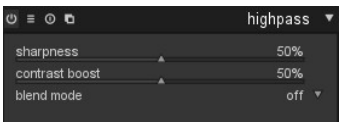
strength

Set the strength of the effect.

3.4.10.7. Highpass

Overview

Highpass acts as a high pass filter. The primary usage for this filter is in combination with a blending operator. Try out blend mode “soft light” to get high-pass sharpening. Use the opacity slider to adjust the strength of the effect or even *conditional blending* (see Section 3.4.3, “Conditional blending”) to limit the effect to only parts of your image.



Usage

sharpen

Set the sharpness. The higher, the more details.

contrast boost

Set the contrast boost.

3.4.10.8. Lowpass

Overview

A lowpass filter (eg. gaussian blur) with additional control of the outcome both of contrast and saturation. The primary usage for lowpass filter is in combination with a *blending operator* (see Section 3.4.2, “Blending operators”). Try out the preset named “local contrast mask” with an “overlay” blending operation.

Usage

This module offers enormous artistic potential, albeit, with results that are sometimes difficult to predict.

radius

Set the radius of the blur.

soften with

This combobox defines the blur algorithm; you can choose between “gaussian” blur (default) and “bilateral” filter. The latter leads to an edge preserving blur. “gaussian” will blur all image channels: L, a and b. “bilateral” will only blur L channel.

contrast

Changes the contrast. Negative values result in an inverted negative image. Higher absolute values increase contrast; lower absolute values reduce contrast. A value of zero leads to a neutral plane.

saturation

Changes the color saturation. Negative values result in complementary colors by inverting the a/b-channels. Higher absolute values increase color saturation; lower absolute values reduce color saturation. A value of zero leads to a desaturated black&white image.

Examples

The original image, already heavily processed. The boat is almost a silhouette.

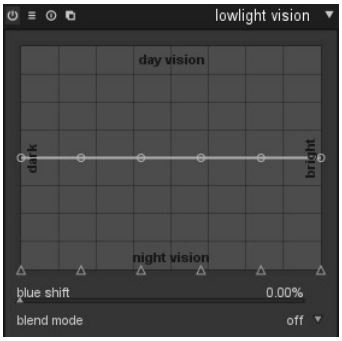
Bilateral blur with high radius. Desaturated, inverted and with high contrast.

Resulting image from lowpass filter ...

... and final image after this was applied with blend mode "vividlight".

3.4.10.9. Low light

Overview



The *low light* module allows one to simulate human lowlight vision, thus providing ability to make low-light pictures look closer to reality. It can also be used to perform a day to night conversion.

The idea is to calculate a scotopic vision [http://en.wikipedia.org/wiki/Scotopic_vision] image which is perceived by rods rather than than cones in the eye under low light. Scotopic lightness then is mixed with photopic value (regular color image pixel) using some blending function. Also this module is able to simulate the Purkinje effect [http://en.wikipedia.org/wiki/Purkinje_effect] by adding some blueness to the dark parts of the image.

Usage

This module comes with several presets. Give them a try to get a better feeling how it works.

curve

The horizontal axis is about pixel lightness from dark (left) to bright (right). The vertical axis represents the kind of vision from night vision (bottom) to day vision (top)

blue

Set the blue hint in shadows (Purkinje effect).

Examples

Image 1. This is the original image.

Image 1. With low light module on.

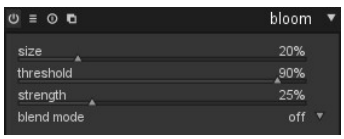
Image 2. This is the original image.

Image 2. With low light module on.

3.4.10.10. Bloom

Overview

This module boost highlights and creates a soft blooms them over the image, hence the name of the effect. There are numerous ways to use this module depending on the image's actual scenery lighting.



Usage

Starting from the default settings change the strength value for a pleasent look, then change the size to control the spread of light.

size

Represents the spatial extent of the bloom effect.

threshold

Set the threshold for the increase in brightness.

strength

Set the strength of overlightning for the effect.

Examples

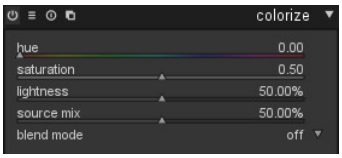
This is the original image, use it as reference for the changes below...

Here we have choosen to use a size of 10%, which is a rather small radius for the soft light spread. We boosted up the strength to 50% for a more exaggerated effect.

3.4.10.11. Colorize

Overview

This module is an artistic feature that adds a solid layer of color to your image.



Usage

Several parameters control the effect of this module. Much more versatility can be reached if you apply blending or even conditional blending (see Section 3.4.2, “Blending operators” and Section 3.4.3, “Conditional blending”).

hue

Selects the hue of the color layer.

saturation

Selects the color saturation of shadow tones.

lightness

Selects the lightness of the color layer.

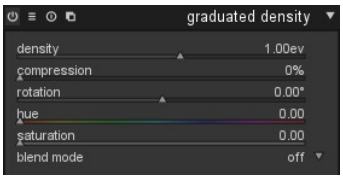
source mix

This slider controls how the lightness of the input image is mixed in. If you set this to zero a uniformly colored plane will result.

3.4.10.12. Graduated Density

Overview

This module aims at simulating a graduated density filter, in order to correct exposure and color in a progressive manner.



Usage

The module uses a gradient to modify the exposure and the color cast of the image in a non-homogeneously manner.

density

Set the density of the filter in [ev]. A low value underexposes slightly whereas a high value creates a strong filter.

It is expressed as [ev] that is equivalent to *f-stops*. Lens filters are often referred as ND2, ND4, ND8 and so on. Each time you add an [ev] you double the ND. So ND2 is 1 ev, ND4 is 2 ev, and so on. You can also express it in optical density or transmittance. The table below sums up the different approach for the most common filters:

ND	[ev] or f-stop	absorbance	transmittance
ND2	-1	0.3	50%
ND4	-2	0.6	25%
ND8	-3	0.9	12.5%
ND400	-9	2.7	0.195%

compression

Set progressiveness of the gradient. A low value creates a smooth transition, whereas a high value makes the transition abrupt.

hue

Set the hue to add a color cast to the gradient.

saturation

Set the saturation to add a color cast to the gradient.

position

You can set the position of the gradient directly on the image by moving the white line. For fine tuning, you can also use the rotation slider. Negative values turn clockwise.

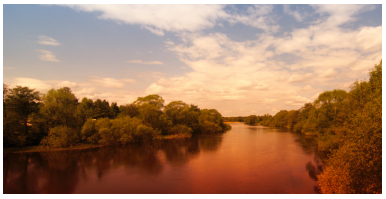
TIP: If you know you intend to use the graduated density filter before actually making a shot with your camera you might want to underexpose by one or two thirds of an f-stop to make sure detail remains in the highlights. When all detail has truly been blown out the graduated density filter cannot produce a pleasing results, this is a limitation that is inherent to digital postprocessing. For instructions on how to intentionally underexpose, please consult your camera's manual, look for “exposure compensation”.

Examples

Here is an example that shows various options of darktable's graduated density filter:

This is the original image with a pretty overexposed sky, use it as reference for the changes below...

And now we have added a neutral ND8 filter which does a pretty good job on the image..

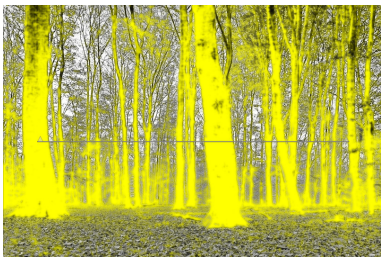


And at last, I added an orange colored filter rotating it -180 degrees, applying it on water/trees for a more artistic use of the filter.

darktable's graduated density filter is a powerful tool. Nevertheless, hardware filters have some advantages over a pure software solution. With a physical GND filter you can in fact reduce the dynamic range of your scene to make it better fit the limits of your camera sensor.

In this example a hardware GND filter (Hitech ND0.6, soft edge) helped me to prevent over-exposure in the sky and tree tops, while at the same time getting a well exposed image of the ground. A rather disturbing element is the decay of lightness in the tree trunks from bottom to top.

darktable's graduated density filter together with the conditional blending feature (see Section 3.4.3, “Conditional blending”) comes in handy. We can add a lightness gradient that is just inverted in relation to the hardware filter. As we only want to compensate the unnatural decay of lightness in the tree trunks, we combine the module with a suited blend mask.



The resulting image.

3.5. Examples

3.5.1. Converting to black and white

3.5.1.1. Overview

Black and white conversion can be achieved in several ways with darktable. Indeed, darktable comes with a lot of modules, especially for color manipulation. In this manual, I will show you 4 ways to perform a black and white conversion.

3.5.1.2. The obvious way: monochrome module

To perform the conversion, just activate the *monochrome* module (Section 3.4.8.6, “Monochrome”). You can then simulate a color filter, by dragging the circle above the colours you want to filter. Filter size can be modified thanks to wheel scrolling.

3.5.1.3. The simple way: color correction module

To perform such conversion we use the *color correction* module (Section 3.4.8.5, “Color correction”).

1. Activate the color correction module

2. Use the bottom slider to set saturation to zero

3.5.1.4. The artistic way: color zones module

To perform the conversion we use the *color zone* module (Section 3.4.8.7, “Color zones”).

1. Activate the colour zones module

2. By default, the first radio-buttons row is set to “colorness” whereas the second is set to “hue” wich means that color are selected according to their hue (horizontal scale) and you can change for each hue its “colorness” (vertical scale). You simply need to set all points to the minimum of the vertical scale to de-saturate every hue.

3. But now if you want, you can keep some hues a little bit saturated, so your image will be all black and white but some hue. A classical use for portrait is to keep red hue saturated in order to make the lips standing out.

   You can also use presets that perform black and white conversion, keeping some hues saturated.

3.5.1.5. The sophisticated way: channel mixer module

To perform the conversion we use the *channel mixer* module (Section 3.4.8.2, “Channel mixer”).

1. Activate the channel mixer module

2. Select the gray output channel

3. Set the proportion of each color, the sum having to equal 1 if you want to keep your global lightness.

3.5.2. Cross-processing

3.5.2.1. Overview

Cross-processing is a analog processing technique where slide film (normally developed thanks to an E6 solution) is processed in chemicals used for processing print film (C41). The resulting images get skewed colors usually a cyan hue and increased contrast and saturation.

The standard way for doing digital cross-processing is to use a channel curve tool but darktable lacks this tool for the moment and another way to accomodiate the effect is used.

3.5.2.2. Procedure

This procedure uses tone curve, channel mixer and splittoning modules.

1. Image preparation

   Prepare the image for the cross process steps by adjusting the base settings such as exposure, whitebalance etc. for a correctly looking image.

2. Boost contrast

   Select the medium contrast curve preset for *tone curve* module (Section 3.4.7.3, “Tone curve”) to boost the overall contrast in the image. You might later return here to tune the curve for better result.

3. Color cast

   This step changes the color cast as the base for the effect using the *channel mixer* module (Section 3.4.8.2, “Channel mixer”). You might later again return to this and finetune the colorcast of the final result.

   a. Enable the channel mixer module

   b. Select blue channel and set blue color value to 0.8

   c. Select red channel and change blue color value to 0.1

   d. Select green channel and change blue color value to 0.1

4. Splittoning

   We use *splittoning* (Section 3.4.10.3, “Splittoning”) to add some more coloring to the result for cyan/blue shadows and yellow highlight.

   a. Enable the splittoning module

   b. Select a cyan/blue tone for shadows and set saturation around 50%

   c. Select a yellow/orange tone for highlights and set saturation around 70%

d. Set compression to 10%

e. Use the balance slider to tune the splittoning effect. This differs on every image due to it's exposure, motive etc.

3.5.3. Cyan toned image

3.5.3.1. Overview

Cyan is a nice color touchup for black and white images, this example guides you through how to make this with darktable and how to control the tone. You can choose any tone of your like for this tutorial...

3.5.3.2. Procedure

This procedure uses tone curve, channel mixer and splittoning modules.

1. Image preparation

   Prepare the image for the cyan toned steps by adjusting the base settings such as exposure,black level, contrast etc. for a correctly looking image.

2. Black and white

   Enable the *monochrome* module (Section 3.4.8.6, “Monochrome”) to make the image black and white.

3. Add color tone

   This step selects the base tone of the image using *channel mixer* (Section 3.4.8.2, “Channel mixer”), we are going for cyan tone but you can choose any tone that you like here.

   a. Enable the channel mixer module

   b. Select red channel destination and set red color value to 0.7

   c. Select green channel destination and red color value to 1.15

   d. Select blue channel destination and red color value to 1.15

   As you notice we mix blue and green color to get a cyan tone, we substract 0.3 from red channel and add them to blue and green.

4. Splittoning

   The result of previous step does also add a color cast on highlight that we actually want to have white for a prettier result. We also want to add some blue color cast to the shadows in order to emphasis them. Module *splittoning* (Section 3.4.10.3, “Splittoning”) can accomplish this.

   a. Enable the splittoning module

b. Select a blue/cyan tone for shadows and set saturation around 50%

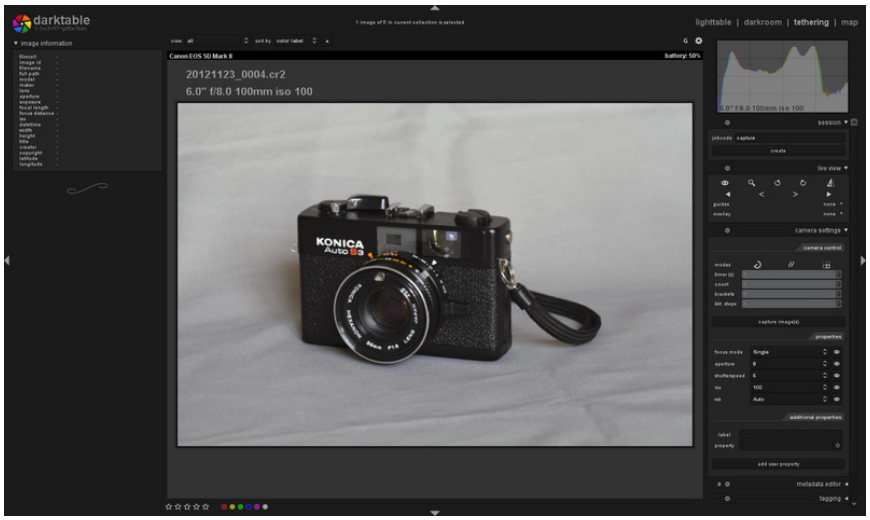
c. Set highlights saturation to zero, to remove saturation on highlights.

d. Set compression to zero

e. Use the balance slider to tune the effect, our example uses a balance of 70/30

Chapter 4. Tethering

The tethering view allows you to capture images directly into darktable from your connected camera.



4.1. Overview

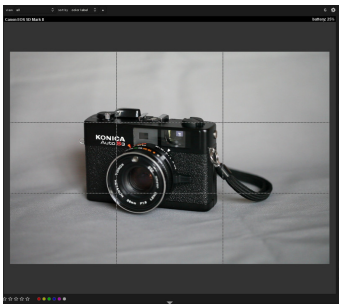
To use the tethering feature you need to connect your camera to your PC using a USB cable. Your computer might ask to mount or view the connected camera. Do not mount or view the camera. If that happens automatically, you will need to “unmount/eject” the camera. This is required to unlock the camera so darktable can lock it for usage.

After the USB cable is connected, look at the import panel in lighttable mode (see Section 2.2.1, “Import”). If your camera is not visible in this panel, click the “scan devices” button and it will appear with two functions: “import from camera” and “tethered shoot”. Click “tethered shoot” to enter the tethering mode.

darktable uses gphoto2 to interface with your camera. If you have problems finding the connected camera as described above, check the troubleshoot section in this chapter in order to verify your camera has tethering support.

4.1.1. Tethering

In the center view images are shown while you capture them. You can get an exposure by either using darktable's userinterface or manually triggering a capture on your camera. If you are using LiveView it will be shown in darktable's center view.



When entering tethering view, a filmroll will be created using the same structure as defined when you import from the camera. Job code will be predefined as “capture”.

If you want to group your captures into different filmrolls, you should use the session panel in right side. When entering a new name and pressing enter, a new filmroll will be created and captured images will go into this new filmroll.

darktable provides some nifty tools to setup a capture in the user interface. You can setup timelapse captures and brackets for HDR creations. The configuration is so dynamic that you can create sequential capture of brackets - go figure... For more information read the documentation about the capture panel and the examples in this chapter.

4.2. Tethering panels

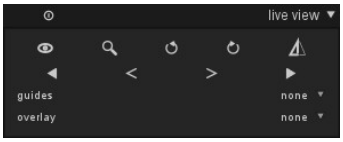
This section contains documentation for panels that are specific to the tethering view.

4.2.1. Session

A session is a sequence of exposures taken in tethering mode and going into a single filmroll. A new session equals a new filmroll. A filmroll is created with the same storage structure that is used when you import images from camera into darktable.

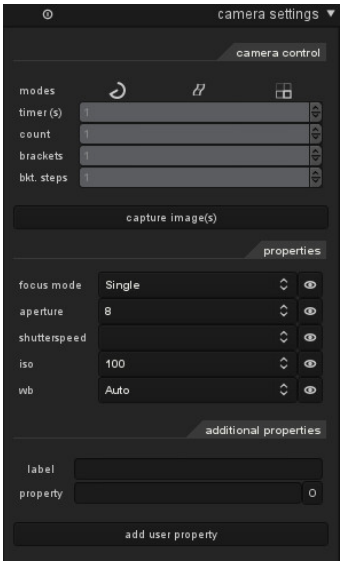
It's a bit akward, but configuring this storage structure is done in the camera import dialog for now.

4.2.2. Live view



This panel gives you control of your camera's live view mode. Functionality such as focus setting, rotation, adding guides and overlays are supported.

4.2.3. Camera settings



The camera settings section allows you to set up a capture job. This can include sequence, bracket and delayed captures. You are also able to control other camera settings such as focus mode, aperture, shutter speed, ISO and white balance.

4.3. Examples

This section contains examples of typical usages of tethering.

4.3.1. Studio setup with screening

This is a pretty common use case. You have your studio and subject set up, the camera is connected to your computer and tethering view is active in darktable. You work at the camera and take images. Whenever you want, you can screen the image directly on your computer monitor instead of using the camera LCD for validation.

This workflow is efficient and effective, since you can immediately review your captures instead of waiting until after the shoot when everyone is gone. If you're shooting a model this is a pretty nice way to pre-view the captures with the client instead of fumbling around with your camera.

Working in the tethered mode can save you time and aggravation. Set a session name, shoot your images and they will save in the correct filmroll for the session for easy on-site review.

4.3.2. Capturing a timelapse

A timelapse is a video clip composed of images taken in a time sequence. A typical example is to take a timelapse of cityscapes where you capture clouds and traffic etc.

To setup a timelapse capture, create a new session as descibed previously. Now decide if you want to shoot in manual or auto mode. Only use auto in situations were the ambient light will change significantly during the time of the shoot, eg. shooting a timelapse over 24 hours might give you easier control of light in that kind of captured sequence.

The camera settings panel is where you define delay and sequence. Sequence will give you the oppertunity to choose how many images you want to capture and delay will set the time in seconds between captures.

To start the capture click the capture button in the same panel and watch the filmstrip fill up with images. The latest captured image is always displayed in center view.

4.4. Troubleshoot

4.4.1. Verify that your camera is supported

This troubleshoting guide will give you steps to verify your camera can be used with tethering. This is done using the gphoto2 commandline tools. This is what darktable uses to interface with your camera.

1. Verify that camera is detected

   The following command will verify a camera that is connected to the computer and detected by gphoto2. Find your camera port name to use it in the following tests below. Usally port “usb:” will be enough and therefore used in these examples.

   ```
   env LANG=C gphoto2 --auto-detect
   ```

2. Verify camera driver abillities

   Execute the following command and verify that the *capture choices* ability supports “Image” and *configuration support* is “yes”. darktable will check these two abilities and decide if “tethered shoot” button should be shown or not.

   ```
   env LANG=C gphoto2 --port usb: --abilities
   ```

3. Verify camera remote capture

   This step will verify that your camera can be remotely controlled; that it can capture an image, download it to your computer and display it within darktable.

   ```
   env LANG=C gphoto2 --port usb: --capture-image-and-download
   ```

4. Verify camera tethered capture

   And this last step tests if your camera supports events which darktable heavily relies on. Running this command will make the gphoto2 process wait for an image capture event which you must manually trigger on your camera. If successful, the image will be downloaded to your computer.

   ```
   env LANG=C gphoto2 --port usb: --capture-tethered
   ```

4.4.2. So, now what?

If any of the steps above failed, there are problems with your specific camera and driver. Please report the issues to gphoto2 mailing list for further help. You can find the mailing list at www.gphoto.org [http://www.gphoto.org/mailinglists/]. Add the following flags to the failed command above for better support and attach the log output to your mail:

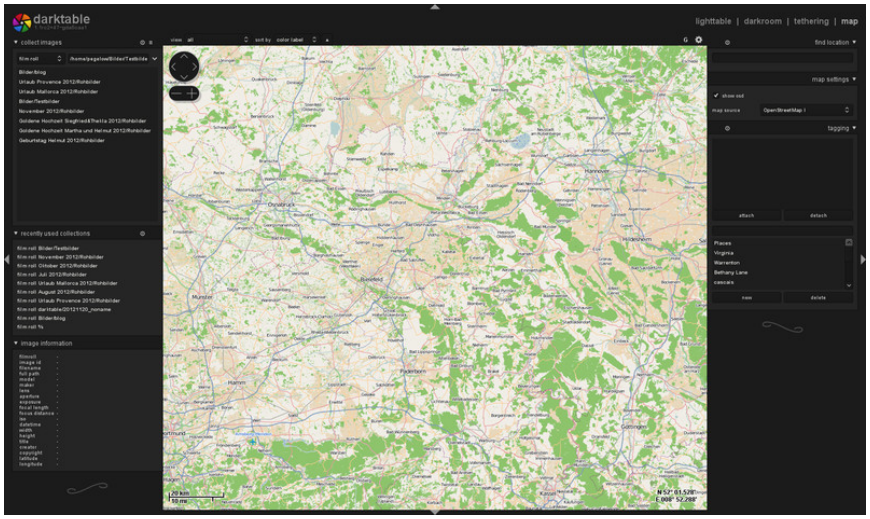
```
--debug --debug-file gphoto2_debug.log
```

If you successfully went through all the tests above, your camera will most likely be supported by darktable. Even if successful, if you stumble upon a problem in darktable, please file a bug at redmine [http://www.darktable.org/redmine]. Please attach the log outputs from the steps above and the log file output generated after starting darktable with the following command.

```
darktable -d camctl 2>1 >camctl.log
```

Chapter 5. Map

The Map view is were you geotag your images.



5.1. Overview

Map view will show you a world map with the currently open image, or filmroll of images, pinned to their geotagged location. This requires that the image was geotagged by a camera with that feature. Some newer cameras, including smartphones, are already equipped with GPS receivers. Other cameras may need additional GPS hardware to do this.

Even if your camera doesn't support this feature, there is an alternative method. darktable can match the EXIF time and date data in your image(s) to a separate GPX data tracking file created by a GPS tracker recording your movements. These can be handheld devices or a GPS tracker app on your smartphone. This is all done in the lighttable view (see Section 2.2.10, “Geotagging”).

5.1.1. Center map view

In the center of the map view you will see a map.

Map data are taken from open map sources on the internet. New map data are only available if you are connected to the internet. darktable keeps a disk cache of previously loaded map data.

Your mouse will allow navigation in the map. Left-click will drag the map; using the scroll-wheel will zoom in or out.

There are on-screen controls and displays that assist you to find your way. A navigation area is located on top left of the map. Use it as an alternative to mouse-dragging and scrolling. The scale of your map is displayed on bottom left. On bottom right you see the geographical coordinates for the center of the map.

Images that already have geo location attributes in their metadata are displayed as small icons on the map.

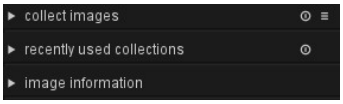
In order to assign geo coordinates to an image, activate the film-strip on the lower panel (press *Ctrl-f*). You can simply assign a geo location to an image by dragging the image icon from the film-strip and position it onto the map. darktable will record the new geo location (longitude and latitude) as part of the image metadata. Exported images will include this data.

Left and right to the central map there are panels for additional control.

5.2. Map panels

This section contains documentation for panels that are specific to the map view.

5.2.1. Left panels



The panels on the left side we already know from lighttable mode (Section 2.2, “Lighttable panels”).

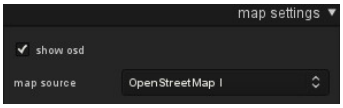
Choose your image selection rules with the *collect images* panel. Recently used collections can be chosen with their respective names in a separate panel. You can also get an overview of the information of the image under your mouse cursor in a panel labeled *image information*.

5.2.2. Find location

The *find location* module is used to search for a place on map. You need to be connected to the internet to use this feature.

To use, type in a place or address, press enter and a list of results will be shown. Click on one of the resulting items and the map will zoom to that location. Now drag images from the film strip at the bottom of the screen to their location on the map. The GPS location will be embedded in the image.

5.2.3. Map settings



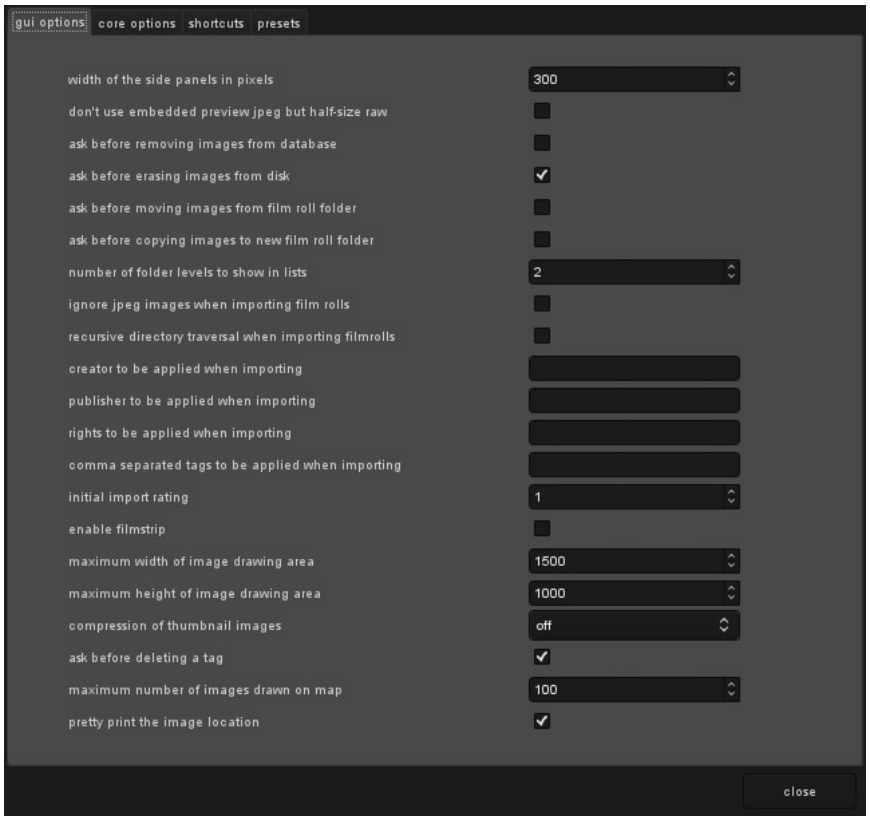
In the *map settings* panel you can select your preferred map data from various providers. Some will provide different layers, such as satelite view etc., which you can toggle.

5.2.4. Tagging

The *tagging* panel allows you to attach or detach different tags to an image as well as creating or deleting a tag. It is divided into two parts. The upper part contains tags that are currently attached to the image. The lower part contains all available tags. You must select or mouse over an image for the data to be displayed. See Section 2.2.12, “Tagging” for more details.

Chapter 6. Preferences and settings

darktable comes with a number of settings that can be configured by users. You reach the configuration menu by clicking ⚙ at the top of the screen.



6.1. GUI options

These options control the look and feel of darktable.

width of the side panels in pixels

This controls the size of side panels in pixels. Side panels are found left and right to the center view (default 300).

don't use embedded preview jpeg but half-size raw

Check this option to not use the embedded jpeg from the raw file but process the raw data. This is slower but gives you color managed thumbnails (default off).

ask before removing images from database

Always ask the user before any image is removed from the database (default on).

ask before erasing images from disk

Always ask the user before any image file is deleted (default on).

ask before moving images from film roll folder

Always ask the user before any image file is moved (default on).

ask before copying images to new film roll folder

Always ask the user before any image file is copied (default on).

number of folder levels to show in lists

The number of folder levels to show in film roll names, starting from the right (default 1).

ignore jpeg images when importing film rolls

When having RAW+JPEG images together in one directory it makes no sense to import both. With this flag one can ignore all jpegs found (default off).

recursive directory traversal when importing filmrolls

Not only import images from the directory selected but recursively go through all subdirectories as well (default off).

creator to be applied when importing

If provided, automatically add this string as a creator tag when importing images (default none).

publisher to be applied when importing

If provided, automatically add this string as a publisher tag when importing images (default none).

rights to be applied when importing

If provided, automatically add this string as a copyrights tag when importing images (default none).

initial import rating

Initial star rating (from 0 to 5) for all images when importing a filmroll (default 1).

enable filmstrip

Enable the filmstrip in darkroom, tethering and geomapping modes (default on).

maximum width of image drawing area

Maximum width of the image drawing area in darkroom mode - adjust to your screen. Needs a restart and will invalidate current thumbnail caches (default 1300).

maximum height of image drawing area

Maximum height of the image drawing area in darkroom mode - adjust to your screen. Needs a restart and will invalidate current thumbnail caches (default 1000).

compression of thumbnail images

Controls the compression of thumbnail images in memory and on disk. Options: "off", "low quality (fast)", "high quality (slow)"; (default off).

ask before deleting a tag

Always ask user before deleting a tag from an image (default on).

maximum number of images drawn on map

The maximum number of geotagged images drawn on the map. Increasing this number can slow down the drawing of the map. Needs a restart if changed (default 100).

pretty print the image location

Show a more readable representation of the geo location in the image information module (default on).

6.2. Core options

These options control some of the internals of darktable.

memory in bytes to use for mipmap cache

On order to speed-up display of filmrolls, darktable stores thumbnails in a cache on disk and loads it into memory at startup. This value controls the cache size in bytes. It needs a restart if changed (default 536870912).

number of background threads

This controls how many parallel threads are used to create thumbnails during import. On 32bit systems it is strongly recommended to set this to 1. Needs a restart if changed (default 2).

export multiple images in parallel

Set this variable to the desired number of threads if you want multithreaded export to process multiple images at a time. Be warned: every thread will need about 1GB of memory. Setting this to 1 switches on per-image parallelization, which darktable does very efficiently and which normally obsoletes parallel exports. 32bit systems need to set this to 1 (default 1).

host memory limit (in MB) for tiling

In order to manage large images on systems with limited memory darktable does tile-wise processing. This variable controls the maximum amount of memory (in MB) a module may use during image processing. Lower values will force memory hungry modules to process an image with increasing number of tiles. Setting this to 0 will omit any limits. Values below 500 will be treated as 500. On a 32bit system you should set this to 500. Needs a restart if changed (default 1500).

minimum amount of memory (in MB) for a single buffer in tiling

If set to a positive, non-zero value, this variable defines the minimum amount of memory (in MB) that darktable should take for a single tile. On a 32bit system you should set this to 8. 64bit systems can live with higher values. Needs a restart if changed (default 16).

write sidecar file for each image

These redundant XMP files can later be re-imported into a different database, preserving your changes to the image. It's strongly recommended to have this option activated so you don't lose data in case of a database corruption. Backing up your RAW file plus the accompanying XMP file will allow you to fully restore your work (default on).

activate opencl support

darktable can use your GPU to speed up processing significantly. Interface OpenCL requires suitable hardware and matching OpenCL drivers on your system. If one of those is not found the option is greyed out. Can be switched on and off at any time and takes immediate effect (default on).

always use littlecms2 during export

If this option is activated, darktable will use system library littlecms2 instead of its own routines. This is about 28x slower than the default but might give more accurate results in some cases (default off).

do high quality resampling during export

The image will first be processed in full resolution, and downscaled at the very end. This can result in better quality sometimes, but will always be slower (default off).

dithering for darkroom mode

In darkroom mode the image in the center view will be dithered if the *dithering* module is enabled. This adds to the latency of darktable. You may prefer to apply dithering only when exporting images. Switch this parameter to FALSE if you want to have a slightly faster processing speed and can accept banding artifacts on display.

demosaicing for zoomed out darkroom mode

Interpolation when not viewing 1:1 in darkroom mode: "always bilinear (fast)" is fastest, but not as sharp. "at most ppg (reasonable)" is using ppg + interpolation modes specified below, "full (possibly slow)" will use exactly the settings for full-size export (default "at most ppg (reasonable)").

pixel interpolator

Pixel interpolator used in rotation, lens correction, up- and downscaling; options are "bilinear", "bicubic", "lanczos2", "lanczos3" (default "lanczos3").

password storage backend to use

The storage backend for password storage. Options: "none", "kwallet", "gnome keyring" (default none).

6.3. Shortcuts

darktable has a large set of keyboard shortcuts that, with the work of Robert Bieber, is now user configurable through the preference pane.

When you open the shortcuts menu you are presented with a hierarchical list of all actions that can receive a keyboard shortcut. Go to the action you want to change and double click. You are then prompted to press the new key combination to be mapped to the selected action. In order to remove an existing keyboard shortcut, click on the action and press backspace.

You can export your mappings to a file or import mappings from a file. Press “default” to reset all keyaccelerators to their default state.

Below is a table with the default keybindings for actions available in darktable.

<darktable>/global/lighttable view	l
<darktable>/global/darkroom view	d
<darktable>/global/capture view	t
<darktable>/global/map view	m
<darktable>/global/increase brightness	F10
<darktable>/global/decrease brightness	F9
<darktable>/global/increase contrast	F8
<darktable>/global/decrease contrast	F7
<darktable>/global/leave fullscreen	Escape
<darktable>/global/quit	<Primary>q
<darktable>/global/switch view	period
<darktable>/global/toggle fullscreen	F11
<darktable>/global/toggle header	<Primary>h
<darktable>/global/toggle side borders	Tab
<darktable>/views/lighttable/color red	F1
<darktable>/views/lighttable/color yellow	F2
<darktable>/views/lighttable/color green	F3
<darktable>/views/lighttable/color blue	F4
<darktable>/views/lighttable/color purple	F5
<darktable>/views/lighttable/navigate down	<Shift>g
<darktable>/views/lighttable/navigate page down	Page_Down
<darktable>/views/lighttable/navigate page up	Page_Up
<darktable>/views/lighttable/navigate up	g
<darktable>/views/lighttable/preview	z
<darktable>/views/lighttable/rate 1	1
<darktable>/views/lighttable/rate 2	2
<darktable>/views/lighttable/rate 3	3
<darktable>/views/lighttable/rate 4	4
<darktable>/views/lighttable/rate 5	5
<darktable>/views/lighttable/rate desert	0

<darktable>/views/lighttable/rate reject	r
<darktable>/views/lighttable/realign images to grid	l
<darktable>/views/lighttable/scroll center	apostrophe
<darktable>/views/lighttable/scroll down	Down
<darktable>/views/lighttable/scroll left	Left
<darktable>/views/lighttable/scroll right	Right
<darktable>/views/lighttable/scroll up	Up
<darktable>/views/darkroom/export	<Primary>e
<darktable>/views/darkroom/image forward	space
<darktable>/views/darkroom/image back	BackSpace
<darktable>/views/darkroom/overexposed	o
<darktable>/views/darkroom/toggle film strip	<Primary>f
<darktable>/views/darkroom/zoom close-up	<Alt>1
<darktable>/views/darkroom/zoom fill	<Alt>2
<darktable>/views/darkroom/zoom fit	<Alt>3
<darktable>/views/map/redo	<Primary>r
<darktable>/views/map/undo	<Primary>z
<darktable>/views/capture/toggle film strip	<Primary>f
<darktable>/modules/copy_history/copy all	<Primary>c
<darktable>/modules/copy_history/copy	<Primary><Shift>c
<darktable>/modules/copy_history/paste all	<Primary>v
<darktable>/modules/copy_history/paste	<Primary><Shift>v
<darktable>/modules/export/export	<Primary>e
<darktable>/modules/filmstrip/color red	F1
<darktable>/modules/filmstrip/color yellow	F2
<darktable>/modules/filmstrip/color green	F3
<darktable>/modules/filmstrip/color blue	F4
<darktable>/modules/filmstrip/color purple	F5
<darktable>/modules/filmstrip/duplicate image	<Primary>d
<darktable>/modules/filmstrip/copy history parts	<Primary><Shift>c
<darktable>/modules/filmstrip/copy history	<Primary>c
<darktable>/modules/filmstrip/paste history parts	<Primary><Shift>v
<darktable>/modules/filmstrip/paste history	<Primary>v
<darktable>/modules/filmstrip/rate 1	1
<darktable>/modules/filmstrip/rate 2	2
<darktable>/modules/filmstrip/rate 3	3
<darktable>/modules/filmstrip/rate 4	4
<darktable>/modules/filmstrip/rate 5	5
<darktable>/modules/filmstrip/rate desert	0
<darktable>/modules/filmstrip/rate reject	r
<darktable>/modules/image/group	<Primary>g

<darktable>/modules/image/ungroup	<Primary><Shift>g
<darktable>/modules/image/remove from collection	Delete
<darktable>/modules/lighttable_mode/zoom max	<Alt>1
<darktable>/modules/lighttable_mode/zoom in	<Alt>2
<darktable>/modules/lighttable_mode/zoom out	<Alt>3
<darktable>/modules/lighttable_mode/zoom min	<Alt>4
<darktable>/modules/live_view/toggle live view	v
<darktable>/modules/select/select all	<Primary>a
<darktable>/modules/select/invert selection	<Primary>i
<darktable>/modules/select/select none	<Primary><Shift>a
<darktable>/modules/tagging/tag	<Primary>t
<darktable>/image operations/clipping/commit	Return
<darktable>/image operations/colorout/toggle gamutcheck	g
<darktable>/image operations/colorout/toggle softproofing	s
<darktable>/image operations/flip/rotate 90 degrees ccw	bracketleft
<darktable>/image operations/flip/rotate 90 degrees cw	bracketright

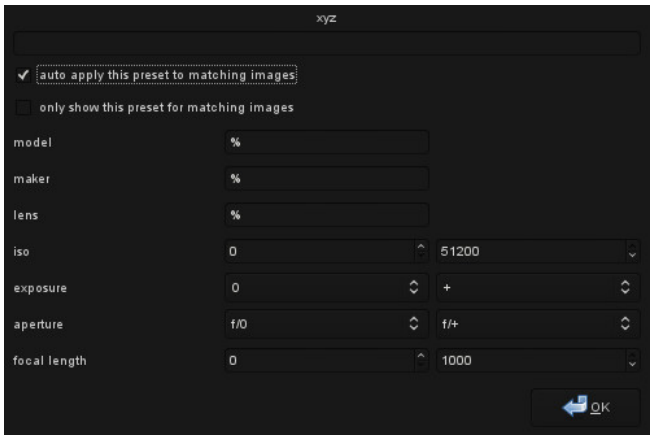
6.4. Presets

This menu gives you an overview of the presets that are defined for darktable's modules. In this dialog you can select whether a certain user defined preset shall be auto-applied to matching images.

darktable already comes with a set of pre-defined presets for several modules. In addition you can define your own presets from within each module in darkroom mode (see Section 3.4.4, “Module presets”).

Pre-defined presets are shown with a lock symbol. Their auto-apply properties can not be changed.

Double clicking on a user-defined preset will open a menu.



auto apply this preset to matching images

activate this checkbutton to automatically apply the preset on newly imported images; a set of fields is displayed where you can define patterns to be matched against EXIF data.

only show this preset for matching images

activate this checkbutton to hide the preset in darkroom mode if it does not match the defined patterns.

model

a pattern to be matched against the EXIF field that describes your camera model; use "%" as wildcard.

maker

a pattern to be matched against the EXIF field that describes the maker of your camera; use "%" as wildcard.

lens

a pattern to be matched against the EXIF field that describes your lens; use "%" as wildcard.

iso

only apply the preset if the ISO value of your image lies within the given range.

exposure

only apply the preset if the exposure time of your image lies within the given range; set "+" as the upper value to match arbitrarily long exposures.

aperture

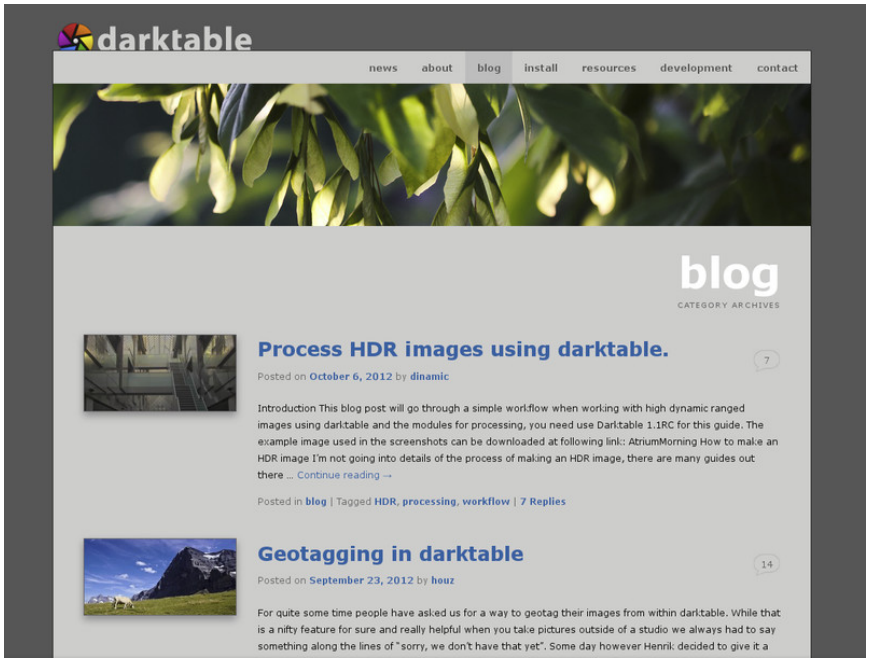
only apply the preset if the aperture of your image lies within the given range; set "f/0" as the lower value to match arbitrarily open apertures; set "f/+" as the upper value to match arbitrarily closed apertures.

focal length

only apply the preset if the focal length of your image lies within the given range (from 0 to 1000).

Chapter 7. Special topics

This chapter touches several technical topics which might help you to get darktable running on specific hardware or optimize its performance. A lot of additional technical background information and many tips and tricks are also covered in an extensive blog section that you can find on our homepage [http://www.darktable.org].



7.1. darktable and memory

darktable's memory requirements are high. A simple calculation makes this clear. If you have a 20MPx image, darktable for precision reasons will store this internally as a 4 × 32-bit floating point cell for each pixel. Each full image of this size will need about 300MB of memory. As we want to process the image, we will at least need two buffers for each module – one for input and one for output. If we have a more complex module, its algorithm might additionally require several intermediate buffers of the same size. Without further optimization, anything between 600MB and 3GB would be needed only to store and process image data. On top we have darktable's code segment, the code and data of all dynamically linked system libraries, and not to forget further buffers where darktable stores intermediate images for quick access during interactive work (mip map cache). All in all, darktable would like to see a minimum of about 4GB to run happily.

7.1.1. Total system memory

From what I said before, it is evident that your computer needs a sane memory setup to properly run darktable. We suggest that you have a least 4GB of physical RAM plus 4 to 8GB of additional swap space installed. The latter is required, so that your system can swap out temporarily unneeded data to disk in order to free physical RAM.

Theoretically, you could also run darktable with lower amounts of physical RAM and balance this with enough swap space. However, you should be prepared that your system could then heavily "thrash", as it reads or writes data pages to and from the hard disk. We have positive reports that this functions well for several users, but it still might get extremely slow for others...

7.1.2. Available address space

Besides the total amount of system memory there is another limiting factor: the available address space of your hardware architecture. How much memory can be addressed by a process depends on the number of address bits your CPU offers. For a CPU with 32-bit address registers, this is 2^32 bytes, which makes a total of 4GB. This is the absolute upper limit of memory that can be used by a process and it constitutes a tight situation for darktable as we have seen above.

darktable's escape route is called tiling. Instead of processing an image in one big chunk, we split the image into smaller parts for every processing step (module). This will still require one full input and output buffer, but intermediate buffers can be made small enough to have everything fit into the hardware limits.

7.1.3. Memory fragmentation

Unfortunately this is not the full story yet. There is an effect called memory fragmentation, which can and will hit software that needs to do extensive memory management. If such a program allocates 5 times 300MB at a time and frees it again, that memory should normally be available for one big 1.5GB allocation afterwards. This however is often not the case. The system's memory allocator may no longer see this area as one contiguous 1.5GB block but as a row of 300MB areas. If there is no other free area of 1.5GB available, the allocation would fail. During a program run this mechanism will take away more and more of the larger memory blocks in favor of smaller ones. darktable 1.0 introduces a caching algorithm to address this problem. It pre-allocates blocks of memory and makes them available on request.

7.1.4. Further limitations

As if this were not challenging enough, there are further things that might limit your access to memory. On some older boards you need to activate BIOS option "memory remapping" in order to have all physically installed memory enabled. In addition if you are on a 32-bit OS you will probably need a kernel version that has "Physical Address Extension" (PAE) enabled. This is often but not always the case for Linux. Many distributions deliver different kernels, some with and some without PAE activated; you need to choose the right one. To check if the system is setup correctly, use the command "free" in a terminal and examine the output. If the output reports less RAM than you have installed, you have an issue needing correction; for example you have 4GB on your board, but your kernel is only seeing 3GB or less. You need to consult your BIOS manual and the information about your Linux variant for further help.

7.1.5. Setting up darktable on 32-bit systems

As we've seen 32-bit systems are difficult environments for darktable. Still many users are successfully running darktable on them, if the basic requirements in terms of total system memory and the topics mentioned in the paragraphs above are addressed properly.

There are several adjustment parameters to get it running. If you install fresh, darktable will detect your system and set conservative values by default. However, if you upgrade darktable from an older version (e.g. coming from 0.9.3 and going to 1.0), chances are you have unfavorable settings in your preferences. The consequences might be darktable aborting due to allocation failures or – very typically – darktable not being able to properly import a new film roll. As a frequent symptom you get skulls displayed instead of thumbs for many of your pictures.

If this is the case, take a minute to optimize the preference settings in this case. You will find them under "core options" (Section 6.2, "Core options") in darktable's preference dialog. You should also find these parameters as configuration variables in $HOME/.config/darktable/darktablerc and edit them there.

Here is a short explanation of the relevant parameters and their proposed settings.

number of background threads

This parameter defines the maximum number of threads that are allowed in parallel when importing film rolls or doing other background stuff. For obvious reasons on 32-bit systems you can only have one thread eating resources at a time. So you need set this parameter to 1; anything higher will kill you. For the same reason you also must set the number of parallel export threads to 1.

host memory limit (in MB) for tiling

This parameter tells darktable how much memory (in MB) it should assume is available to store image buffers during module operations. If an image can not be processed within these limits in one chunk, tiling will take over and process the image in several parts, one after the other. Set this to the lowest possible value of 500 as a starting point. You might experiment later whether you can increase it a bit in order to reduce the overhead of tiling.

minimum amount of memory (in MB) for a single buffer in tiling

This is a second parameter that controls tiling. It sets a lower limit for the size of intermediate image buffers in megabytes. The parameter is needed to avoid excessive tiling in some cases (for some modules). Set this parameter to a low value of 8. You might tentatively increase it to 16 later.

memory in bytes to use for mipmap cache

This controls how many thumbnails (or mip maps) can be stored in memory at a time. As a starting point set this to something like 256MB (give the number in bytes). To avoid the problem of memory fragmentation during longer runs of darktable, the new caching scheme frontloads the memory costs and allocates this cache once at the beginning. Some Linux kernels use over-committing memory allocation, which means you don't immediately pay for all of the memory in terms of RSS (resident set size, the non-swapped physical memory), but in any case you pay for the address space. As explained before, this poses a problem for 32-bit systems and will, at first sight, appear as a regression over the 0.9.3-style cache. In the long run however, this is all the memory that's ever going to be allocated for thumbnails. So if we can successfully grab this portion once, we are relieving a lot of pressure on fragmentation for long sessions.

7.1.6. darktable on 64-bit systems

There's not much to be said here. Of course also 64-bit systems require a sufficient amount of main memory, so the 4GB plus swap recommendation holds true. On the other hand, 64-bit architectures do not suffer from the specific 32-bit limitations like small address space and fragmentation madness.

Most modern Intel or AMD 64-bit CPUs will have available address space in the range of several Terabytes. The word “modern” is relative in this context: all AMD and Intel CPUs introduced since 2003 and 2004, respectively, offer a 64-bit mode. Linux 64-bit has been available for many years.

All relevant Linux distributions give you the choice to install a 32-bit or a 64-bit version with no added costs. You can even run old 32-bit binaries on a 64-bit Linux. The only thing you need to do: invest some time into the migration. In the end we strongly recommend moving to a 64-bit version of Linux. There really is no reason not to upgrade to 64-bit.

7.2. darktable and OpenCL

darktable can use GPU acceleration via OpenCL to improve performance.

7.2.1. The background

Processing high resolution images is a demanding task needing a modern computer. Both in terms of memory requirements and in terms of CPU power, getting the best out of a typical 15, 20 or 25 Megapixel image can quickly bring your computer to its limits.

darktable's requirements are no exception. Our decision to not compromise processing quality, has led to all calculations being done on 4 × 32bit floating point numbers. This is slower than "ordinary" 8 or 16bit integer algebra, but eliminates all problems of tonal breaks or loss of information.

A lot of hand optimization has been invested to make darktable as fast as possible. If you run a current version of darktable on a modern computer, you might not notice any "slowness". However, there are conditions and certain modules where you will feel (or hear from the howling of your CPU fan) how much your poor multi-core processor has to struggle.

That's where OpenCL comes in. OpenCL allows us to take advantage of the enormous power of modern graphics cards. Gamer's demand for high detailed 3D worlds in modern ego shooters has fostered GPU development. ATI, NVIDIA and Co had to put enormous processing power into their GPUs to meet these demands. The result is modern graphics cards with highly parallelized GPUs to quickly calculate surfaces and textures at high frame rates.

You are not a gamer and you don't take advantage of that power? Well, then you should at least use it in darktable! For the task of highly parallel floating point calculations modern GPUs are much faster than CPUs. That is especially true, when you want to do the same few processing steps over millions of items. Typical use case: processing of megapixel images.

7.2.2. How OpenCL works

As you can imagine, hardware architectures of GPUs can vary significantly. There are different manufacturers, and even different generations of GPUs from the same manufacturer may differ. At the same time GPU manufacturers don't normally disclose all hardware details of their products to the public. One of the known consequences is the need to use proprietary drivers under Linux, if you want to take full advantage of your graphics card.

Fortunately an industry consortium lead by The Khronos Group has developed an open, standardized interface called OpenCL. It allows the use of your GPU as a numerical processing device. OpenCL offers a C99-like programming language with a strong focus on parallel computing. An application that wants to use OpenCL will need OpenCL source code that it hands over to a hardware specific OpenCL compiler at run-time. This way the application can use OpenCL on different GPU architectures (even at the same time). All hardware "secrets" are hidden in this compiler and are normally not visible to the user (or the application). The compiled OpenCL code is loaded onto your GPU and - with certain API calls - it is ready to do calculations for you.

7.2.3. How to activate OpenCL in darktable

Using OpenCL in darktable requires that your PC is equipped with a suitable graphics card and that it has the required libraries in place. Namely modern graphics cards from NVIDIA and ATI come with full OpenCL support. The OpenCL compiler is normally shipped as part of the proprietary graphics driver; it is used as a dynamic library called "libOpenCL.so". This library must be in a folder where it is found by your system's dynamic linker.

When darktable starts, it will first try to find and load libOpenCL.so and – on success – check if the available graphics card comes with OpenCL support. A sufficient amount of graphics memory (1GB+) needs to be available to take advantage of the GPU. If that is OK, darktable tries to setup its OpenCL environment: a processing context needs to be initialized, a calculation pipeline to be started, OpenCL source code files (extension is .cl) need to be read and compiled and the included routines (called OpenCL kernels) need to be prepared for darktablele's modules. If all that is done, the preparation is finished.

Per default OpenCL support is activated in darktable if all the above steps were successful. If you want to de-activate it you can do so in "core options" (Section 6.2, "Core options") by unchecking "activate opencl support". This configuration parameter also tells you if OpenCL initialization failed: it is greyed out in that case.

You can at any time switch OpenCL support off and on; this will happen immediately. Depending on the type of modules you are using, you will notice the effect as a general speed-up during interactive work and during export. Most modules in darktable can take advantage of OpenCL but not all modules are demanding enough to make a noticeable difference. In order to feel a real difference, take modules like *shadows and highlights*, *sharpen*, *lowpass*, *highpass* or even more extreme *equalizer* and *profiled denoise*.

If you are interested in profiling figures, you can start darktable with command line parameters "-d opencl -d perf". After each run of the pixelpipe you will get a detailed allocation of processing time to each module plus an even more fine grained profile for all used OpenCL kernels.

Besides the speed-up you should not see any difference in the results between CPU and GPU processing. Except of rounding errors, the results are designed to be identical. If, for some reasons, darktable fails to properly finish a GPU calculation, it will normally notice and automatically (and transparently) fall back to CPU processing.

7.2.4. Possible problems and solutions

darktable will detect OpenCL run-time problems automatically. It will then reprocess everything on CPU; only speed is affected, the final result should not be endangered.

There can be various reasons why OpenCL could fail during initialization phase. We depend on hardware requirements and on the presence of certain drivers and libraries. In addition all these have to fit in terms of maker model and revision number. If anything does not fit, e.g. your graphics driver (loaded as a kernel module) does not match the version of your libOpenCL.so, OpenCL support is likely not available.

In that case, the best thing to do is start darktable from a console with

```
darktable -d opencl
```

This will give additional debugging output about the initialization and use of OpenCL. First see if you find a line that starts with "[opencl_init] FINALLY …" This should tell you, if OpenCL support is available for you or not. If initialization failed, look at the messages above for anything that reads like "could not be detected" or "could not be created". Check if there is a hint about where it failed.

Here are a few cases observed in the past:

darktable might tell you that no OpenCL aware graphics card is detected or that the available memory on your GPU is too low and the device is discarded. In that case you might need to buy a new card, if you really want OpenCL support.

darktable might find your libOpenCL.so but then tell you that it couldn't get a platform. NVIDIA drivers will often give error code -1001 in that case. This happens because libOpenCL.so is only a wrapper library. For the real work further libraries - specific to vendor, device and driver - need to be loaded. This failed for some reason. There is a structure of files in /etc/OpenCL on your system that libOpenCL.so consults to find these libraries. Check if you find something fishy in there and try to fix it. Often the needed libraries cannot be found by your system's dynamic loader. Giving full path names might help.

darktable might also tell you that a context could not be created. This often indicates a version mismatch between (loaded) graphics driver and libOpenCL. Check if you have leftover kernel modules or graphics libraries of an older install and take appropriate action. In doubt, make a clean reinstall of your graphics driver. Sometimes, immediately after a driver update, the loaded kernel driver does not match the newly installed libraries: reboot your system in that case.

darktable might crash in very rare cases directly during startup. This can happen if your OpenCL setup is completely broken or if driver/library contains a severe bug. If you can’t fix it, you can still use darktable with option “--disable-opencl”, which will skip the entire OpenCL initialization step.

darktable might fail to compile its OpenCL source files at run-time. In that case you will get a number of error messages looking like typical compiler errors. This could indicate an incompatibility between your OpenCL implementation and our interpretation of the standard. In that case visit us at darktable-devel@sourceforge.net and report the problem. Chances are good that we can help you. Please also report if you see significant differences between CPU and GPU processing of an image!

There also exists a few on-CPU implementations of OpenCL. These come as drivers provided by INTEL or AMD. We observed that they do not give us any speed gain versus our hand-optimized CPU code. Therefore we simply discard these devices.

7.2.5. Setting up OpenCL for AMD/ATI devices

While NVIDIA devices will most often run out of the box, there is more to do for AMD/ATI graphics cards. This starts with the fact that AMD/ATI will only report to darktable half of the total GPU memory. For a 1GB device this amounts to 512MB, a value which darktable in its standard configuration will refuse as not being sufficient for its tasks. Consequence: the device will not be used.

On the web you might find as a tip to set environment variable GPU_MAX_HEAP_SIZE to a value of 100 in this case. Indeed this will cause the AMD/ATI driver to report the full installed memory to darktable. However, there is a problem. On many (most?) cards this will cause buffers to be allocated on your computer (host) not on the video card! In this case all memory accesses will need to go through the slow PCIe bus. This will cost you a factor of 10x or more in performance and will render OpenCL useless for you, especially when exporting files.

Another environment variable which changes driver behavior is GPU_MAX_ALLOC_PERCENT. You could set this to 100 in order to allow memory allocations as high as 1GB on your AMD/ATI card. The problem is, this tends to cause darktable to crash sooner or later.

Our recommendation is to leave these settings untouched. Often your card will be recognized with 512MB memory and a maximum allocation size of 128MB. There are three configuration parameter which you set in file $HOME/.config/darktable/darktablerc to get things running. Here are the details:

opencl_memory_requirement

Set this parameter to 500 so that darktable will accept your 512MB graphics memory as being sufficient in memory.

opencl_memory_headroom

This parameter controls how much graphics memory (out of the reported one) darktable should leave untouched for driver and display use. As for AMD/ATI devices we anyhow only can get half of the available RAM it's safe to set this to zero. So all of the 512MB can be used by darktable.

opencl_avoid_atomics

Atomic operations in OpenCL are a special way of data synchronization. They are only used in a few kernels. Unfortunately, some (most?) AMD/ATI devices are extremely slow in processing atomics. It's better to process the affected modules on CPU rather than accepting an ultra-slow GPU codepath. Set this parameter to TRUE if you experience slow processing of modules like *shadows and highlights*, *monochrome*, *local contrast*, or *global tonemap* or if you even get intermittent system freezes.

7.2.6. OpenCL performance optimization

There are some configuration parameters in $HOME/.config/darktable/darktablerc that help to finetune your system's OpenCL performance. Performance in this context mostly means the latency of darktable during interactive work, i.e. how long it takes to reprocess your pixelpipe. For a comfortable workflow it is essential to keep latency low.

In order to get profiling info you start darktable from a terminal with

```
darktable -d opencl -d perf
```

After each reprocessing of pixelpipe - caused by module parameter change, zooming, panning, etc. - you will get the total time and the time spent in each of our OpenCL kernels. The most reliable value is the total time spent in pixelpipe. Please note that the timings given for each indiviual module are unreliable when running the OpenCL pixelpipe asynchronously (see opencl_async_pixelpipe below).

To allow for a fast pixelpipe processing with OpenCL it is essential that we keep the GPU busy. Any interrupts or a stalled data flow will add to the total processing time. This is especially important for the small image buffers we need to handle during interactive work. They can be processed quickly by a fast GPU. However, even short-term stalls of the pixelpipe will easily become a bottleneck.

On the other hand darktable's performance during file exports is more or less only governed by the speed of our algorithms and the horse-power of your GPU. Short-term stalls will not have a noticable effect on the total time of an export.

darktable comes with default settings that should deliver a decent GPU performance on most systems. However, if you want to fiddle around a bit by yourself and try to optimize things further, here is a description of the relevant configuration parameters.

opencl_async_pixelpipe

This boolean flag controls how often we block the OpenCL pixelpipe and get a status on success/failure of all the kernels that have been run. For optimum latency set this to TRUE,

so darktable runs the pixelpipe asynchronously and tries to use as few interrupts as possible. If you experience OpenCL errors like failing kernels, set the parameter to FALSE. darktable will then interrupt after each module so you can more easily isolate the problem. Problems have been reported with some older ATI/AMD cards, like HD57xx, which can produce garbled output if this parameter is set to TRUE. If in doubt, leave it at its default FALSE.

opencl_number_event_handles

Event handles are used so we can monitor success/failure of kernels and profiling info even if the pixelpipe is run asynchronously. The number of event handles is a limited resource of your OpenCL driver. For sure we can recycle them, but there is a limited number that we can use at the same time. Unfortunately, there is no way to find out what the resource limits are; so we need to guess. Our default value of 25 is quite conservative. You might want to try if higher values like 100 give better OpenCL performance. If your driver runs out of free handles you would experience failing OpenCL kernels with error code "-5 (CL_OUT_OF_RESOURCES)" or even crashes or system freezes; reduce the number again in that case. A value of 0 will block darktable from using any event handles. This will prevent darktable from properly monitoring the success of your OpenCL kernels but saves some driver overhead. The consequence is that any failures will likely lead to garbled output without darktable taking notice; only recommended if you know for sure that your system runs rock-solid. You can also set this parameter to -1, which means that darktable assumes no restriction in the number of event handles; this is not recommended.

opencl_synch_cache

This parameter, if set to TRUE, will force darktable to fetch image buffers from your GPU after each module and store them in its pixelpipe cache. This is a very resource consuming operation. It only makes sense if you have a rather slow GPU. In that case darktable might in fact save some time when module parameters have changed, as it can go back to some cached intermediate state and reprocess only part of the pixelpipe. In most cases this parameter should be set to FALSE (default).

opencl_micro_nap

In an ideal case you keep your GPU busy at 100% when reprocessing the pixelpipe. That's good. On the other hand your GPU is also needed to do regular GUI updates. It might happen that there is no sufficient time left for this task. Consequence would by a jerky reaction of your GUI on panning, zooming or when moving sliders. darktable can add small naps into its pixelpipe processing to have the GPU catch some breath and do GUI related stuff. Parameter opencl_micro_nap controls the duration of these naps in microseconds. You need to experiment in order to find an optimum value for your system. Values of 0, 100, 500 and 1000 are good starting points to try. Defaults to 1000.

7.2.7. Multiple OpenCL devices

While most systems will only have one OpenCL capable GPU installed, darktable is also able to make use of multiple devices in parallel. There is a configuration parameter which helps to optimize GPU priorities in that case.

It is important to understand how darktable uses OpenCL devices. Each processing sequence of an image - to convert an input to the final output using a certain history stack - is run in a so called pixelpipe. There are four different types of pixelpipe in darktable. One type is responsible to process the center image view (or full view) in darkroom mode, another pixelpipe processes the preview image (navigation window) top left in darkroom mode. Of each of these two pixelpipe there can be one at a time - with the full and the preview pixelpipe running in parallel. In addition there can be multiple parallel pixelpipes

doing file exports and there can be multiple parallel pixelpipes generating thumbnails. If an OpenCL device is available darktable dynamically allocates it to one specific pixelpipe for one run and releases it afterwards.

The computational demand depends a lot on the pixelpipe type. Preview image and thumbnails have a low resolution and can be processed quickly; center image view is more demanding, let alone the pixelpipe doing a file export. If you have a reasonably fast GPU and want to get a low latency during interactive work, it is therefore important that your GPU is allocated to do the more demanding center image (full) pixelpipe, while the smaller preview image can be processed in parallel by the CPU. Older versions of darktable would therefore not allow the preview pixelpipe to grab any OpenCL device.

Starting with darktable 1.2 there is a more flexible scheme to allocate and prioritize your OpenCL device(s). Configuration parameter “opencl_device_priority” holds a string with the following structure:

```
a,b,c.../k,l,m.../o,p,q.../x,y,z...
```

Each letter represents one specific OpenCL device. There are four fields in the parameter string separated by a slash, each representing one type of pixelpipe. “a,b,c...” defines the devices that are allowed to process the center image (full) pixelpipe. Likewise devices “k,l,m...” can process the preview pixelpipe, devices “o,p,q...” the export pixelpipes and finally devices “x,y,z...” the thumbnail pixelpipes. An empty field means that no OpenCL device may serve this type of pixelpipe.

darktable has an internal numbering system, where the first available OpenCL device will receive number “0”. All further devices are numbered consecutively. This number together with the device name is displayed when you start darktable with “darktable -d opencl”. You can specify a device either by number or by name (upper/lower case and whitespace do not matter). If you have more than one device - all with the same name - you need to use the device numbers in order to differentiate them.

A device specifier can be preceeded by an exclamation mark “!”, in which case the device is excluded from processing this pixelpipe. You can also give an asterisk “*” as a wildcard, representing all devices not mentioned explicitely before in that group.

Sequence order within a group matters. darktable will read the list from left to right and whenever it tries to allocate an OpenCL device to a pixelpipe it will scan the devices in that order, taking the first free device it finds.

darktable's default setting for opencl_device_priority is:

```
*/!0,*/*/*
```

Any detected OpenCL device is allowed to process our center view image. The first OpenCL device (0) is not allowed to process the preview pixelpipe. As a consequence, if there is only one GPU owned by your system, preview pixelpipe will always be processed on CPU, keeping your single GPU exclusively for the more demanding center image view. This is reasonable and identical to the old behavior. No restrictions apply to export and thumbnail pixelpipes.

The default is a good choice if you have only one device. If you have several devices it forms a reasonable starting point. However, as your devices might have quite different levels of processing power, it makes sense to invest a few thoughts and optimize your priority list.

Here is an example. Let's assume we have a system with two devices, a fast Radeon HD7950 and an older and slower GeForce GTS450. darktable (started with “darktable -d opencl”) will report the following devices:

```
[opencl_init] successfully initialized.
[opencl_init] here are the internal numbers and names of
                        OpenCL devices available to darktable:
[opencl_init]           0       'GeForce GTS 450'
[opencl_init]           1       'Tahiti'
[opencl_init] FINALLY: opencl is AVAILABLE on this system.
```

So the GeForce GTS 450 is detected as the first device; the Radeon HD7950 ('Tahiti') as the second one. This order will normally not change unless the hardware or driver configuration is modified. But it's better to use device names rather than numbers to be on the safe side.

As the GTS450 is slower than the HD7950, an optimized opencl_device_priority could look like:

```
!GeForce GTS450,*/!Tahiti,*/Tahiti,*/Tahiti,*
```

The GTS450 is explicitely excluded from doing the center image pixelpipe; this is reserved to “all” other devices (i.e. the HD7950/Tahiti). Completely the opposite for our preview pixelpipe. Here the Tahiti is excluded, so that only the GTS450 will be allowed to do the work.

For file export and thumbnail generation we want all hands on deck. However, darktable should first look if device Tahiti is free, because it's faster. If that's not the case, all other devices - in fact only the GTS450 - are checked.

7.2.8. OpenCL still does not run for me!

As has been said before OpenCL systems come with a huge variety of setups: different GPU manufacturers, different GPU models, varying amounts of GPU memory, different drivers, different distributions etc. Many of the potential problems will only appear with a very specific combination of those factors.

As we developers of darktable on our computers only have access to a small fraction of those variations, please understand that we might not be able to fix your specific problem. There is not much we can do, if there is no way for us to reproduce.

If nothing else helps, the best option might be to start darktable with

```
darktable --disable-opencl
```

In the end there is nothing in darktable which only runs on GPU. Don't let OpenCL discourage you; also darktable's CPU code is highly optimized for performance!