rawkit Documentation

Release 0.6.0

Cameron Paul, Sam Whited

Nov 26, 2018

Contents

Note: *rawkit* is currently unmaintained. Code is provided as-is but no guarantee of support is provided, and there is no guarantee that patches will be merged or reviewed.

`rawkit` is a `ctypes`-based set of LibRaw bindings for Python inspired by Wand. It is licensed under the MIT License.

```
from rawkit.raw import Raw
from rawkit.options import WhiteBalance

with Raw(filename='some/raw/image.CR2') as raw:
  raw.options.white_balance = WhiteBalance(camera=False, auto=True)
  raw.save(filename='some/destination/image.ppm')
```

CHAPTER 1

Requirements

- Python
  - CPython 2.7+
  - CPython 3.5+
  - PyPy 2.5+
  - PyPy3 2.4+
- LibRaw
  - LibRaw 0.16.x
  - LibRaw 0.17.x
  - LibRaw 0.18.x
  - LibRaw 0.19.x

CHAPTER 2

Installing rawkit

First, you'll need to install LibRaw:

- *libraw* or libraw16 on Arch
- *LibRaw* on Fedora 21+ and EPEL 6
- *libraw-bin* on Ubuntu trusty+
- *libraw-bin* on Debian Jessie+

Now you can fetch rawkit from PyPi:

```
$ pip install rawkit
```

CHAPTER 3

Tutorials

3.1 Getting Started

If you read the beginning of this documentation, you've seen one example of using rawkit already. Let's see an even simpler form of it:

```
from rawkit.raw import Raw

with Raw(filename='some/raw/image.CR2') as raw:
  raw.save(filename='some/destination/image.ppm')
```

This constructs a *rawkit.raw.Raw* object which loads the file `image.CR2` as a context manager and then saves the output file `image.ppm`. One of the design goals of rawkit is "have sane defaults", which means that this is pretty much all you need to do to get a decent looking photo. Of course, you probably want to customize how your photo is developed. For this you can use *rawkit.options*.

The `Raw` object you created has a *rawkit.options.Options* object already with the aforementioned sane defaults, so instead of constructing a new object let's just modify the existing one to tweak the white balance and a few other options:

```
from rawkit.raw import Raw
from rawkit.options import WhiteBalance, colorspaces, gamma_curves

with Raw(filename='some/raw/image.CR2') as raw:
  raw.options.white_balance = WhiteBalance(camera=False, auto=True)
  raw.options.colorspace = colorspaces.adobe_rgb
  raw.options.gamma = gamma_curves.adobe_rgb
  raw.save(filename='some/destination/image.ppm')
```

By default rawkit uses the white balance written to the raw file by your camera (if available) and falls back to automatically guessing at the white balance if no camera specified white balance is available. However, here we've constructed a new *rawkit.options.WhiteBalance* object which does not attempt to use the camera white balance (note that `WhiteBalance` objects are immutable, so you'll always need to construct a new one if you're changing the

white balance). We've also changed the colorspace to Adobe RGB instead of the default sRGB, and changed the gamma curve to use the corrective power function for the Adobe RGB colorspace.

Lots of other options can be set. A full list can be found in the API documentation for the *rawkit.options* module.

Of course, we probably don't want to process just one raw file. A common photography workflow is to do some basic level of processing to lots of files at once (eg. an entire days worth of shooting) and then go back and tweak individual photos as necessary. To do this, we can construct our own options object and reuse it:

```
import sys

from rawkit.raw import Raw
from rawkit.options import WhiteBalance, colorspaces, gamma_curves
from rawkit.options import Options

opts = Options({
  'white_balance': WhiteBalance(camera=False, auto=True),
  'colorspace': colorspaces.adobe_rgb,
})

opts.gamma = gamma_curves.adobe_rgb


for rawfile in sys.argv[1:]
  with Raw(filename=rawfile) as raw:
    raw.options = opts
    raw.save(filename='{}.ppm'.format(rawfile))
```

As you can see, two methods for setting options on an `Options` object are presented here: via a dict passed to the constructor, or by manually setting the properties. Because the dict method tolerates arbitrary fields, you must be very careful not to make a typo. Eg. setting:

```
opts = Options({
  'colourspace': colorspaces.adobe_rgb,
  'white_blaance': WhiteBalance(greybox=[1034, 1058, 1096, 1085])
})
```

will run without error, but there will be no difference to your output photos. However, trying to set options via:

```
opts = Options()
opts.colourspace = colorspaces.adobe_rgb
opts.white_blaance = WhiteBalance(greybox=[1034, 1058, 1096, 1085])
```

Will result in an `AttributeError`. This is the recommended method for manually setting options because it will fail early and loudly!

Now that we've seen the basics (loading and saving raw files and setting options), let's turn our simple example into something useful: A program which will take in the name of one or more raw files and attempt to save them as standard TIFF files. First, we'll snag the arguments and add a bit of error checking (we'll also get rid of the options and just use the defaults for now):

```
import sys

from libraw.errors import FileUnsupported
from rawkit.errors import InvalidFileType
from rawkit.raw import Raw

```

(continues on next page)

(continued from previous page)

```
if __name__ == "__main__":

  for rawfile in sys.argv[1:]:
    try:
      with Raw(filename=rawfile) as raw:
        outfile = '{}.tiff'.format(rawfile)
        raw.save(filename=outfile)
        print(
          'Wrote file: "{}".'.format(
            outfile
          )
        )
    except (InvalidFileType, FileUnsupported):
        print(
          'WARNING: File "{}" could not be processed.'.format(
            rawfile
          ),
          file=sys.stderr
        )
```

Of course, while this works, it's still a bit slow. Let's add a thread pool to the mix and process multiple raw files at once (not that this has anything to do with actually using rawkit, but we might as well do things right):

```
import concurrent.futures
import os
import sys

from libraw.errors import FileUnsupported
from rawkit.errors import InvalidFileType
from rawkit.raw import Raw

def develop_photo(rawfile):
    with Raw(filename=rawfile) as raw:
        outfile = '{}.tiff'.format(rawfile)
        raw.save(filename=outfile)
        return outfile

if __name__ == "__main__":

    with concurrent.futures.ThreadPoolExecutor(max_workers=(
        (os.cpu_count() or 2) * 2)) as executor:
        develop_futures = {executor.submit(develop_photo, raw): raw for raw
            in sys.argv[1:]}
        for future in concurrent.futures.as_completed(develop_futures):
            raw = develop_futures[future]
            try:
                data = future.result()
            except (InvalidFileType, FileUnsupported):
                print(
                  'WARNING: File "{}" could not be processed'.format(raw),
                  file=sys.stderr
                )
            else:
                print('Wrote file: "{}"'.format(data))
```

That's it, you've made a useful application which uses rawkit to develop raw photos! For a slightly more interesting example, take a look at the source to photoREPL, an experimental interface for editing photos from the command line.

3.2 Using Rawkit with NumPy

Rawkit can be used to easily access raw data as NumPy arrays.

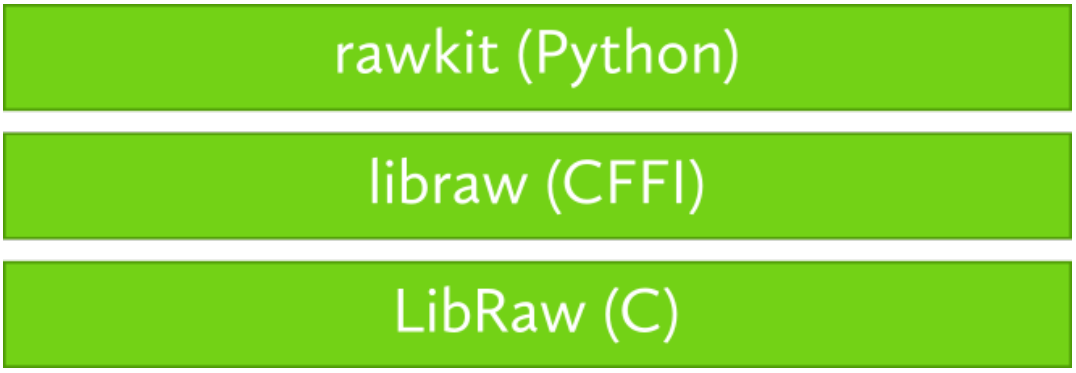
```
from rawkit.raw import Raw

with Raw(filename='some/raw/image.CR2') as raw:
    pixels = raw.as_array()
    color_filter_array = raw.color_filter_array

    # Randomly chosen pixel
    x = 307
    y = 123

    intensity = pixels[y][x]
    color = color_filter_array[y % 2][x % 2]
    message = 'The pixel at {x},{y} has intensity {i} and color {c}'

    print(message.format(
        x=x,
        y=y,
        i=intensity,
        c=color,
    ))
```

CHAPTER 4

Architecture and Design

4.1 Architecture

When we talk about "rawkit" we're actually talking about an entire stack of libraries which work together to give you a simple way to work with raw photo data in Python. However, under the hood, rawkit comprises three separate libraries which operate in a teired structure:



The bottom layer is the LibRaw C library, which is used to actually extract data from raw photo files, and to do basic processing. LibRaw is not actually bundled with rawkit, and must already be installed on the end users computer. The next layer, also called *libraw*, is a low-level Python library which uses ctypes to link to the LibRaw C code. This library, while written in Python, generally just looks and acts like the lower-level C code, albeit with slightly more Pythonic error handling and a few helper functions to make it easier to use from within Python. However, you generally shouldn't use libraw. Instead, you should use the highest level methods available, *rawkit*. The actual rawkit namespace provides a module which builds on libraw to provide a fully Pythonic interface to the underlying library (eg. `rawkit.Raw` objects, context managers, an API for setting options, etc.). If at all possible, you should use the rawkit module in your applications, but the libraw module is still exposed in case you need to dig down and perform some functionality that is not exposed by rawkit.

More details about each tier can be found below.

4.1.1 LibRaw

The foundation of the entire rawkit stack is the LibRaw C library. LibRaw is maintained by LibRaw, LLC. and does the actual grunt work of loading raw files, extracting data, and developing photos via its dcraw emulation layer. It is the only real dependency of rawkit and must be installed on the end-users computer before this library will actually work.

4.1.2 libraw

The *libraw* module is a set of Python bindings which use ctypes to talk to the LibRaw library on the users system. The libraw module provides very low level bindings that mostly juts mimic the C structs present in LibRaw. It also defines function and method arguments and return types, allows you to use Python functions as callbacks to LibRaw events, maps LibRaw error codes to actual Python exceptions, and handles the actual linking with `libraw.so` (or the equivalent library on your system). In general, you should never have to call libraw directly. Instead, you should use the higher level API's provided by *rawkit*.

4.1.3 rawkit

The *rawkit* module is the highest level part of the rawkit architecture. This module handles raw files in a Pythonic way by abstracting them to a `rawkit.Raw` object which acts as a context manager, and allowing you to set options for how that raw file should be processed. It also contains a set of utility functions (see *rawkit.util*) for dealing with common operations that may not be directly related to raw files (eg. discovering support for raw files, or getting a list of cameras supported by the linked version of LibRaw).

CHAPTER 5

API Reference

The *rawkit* package provides two modules: *rawkit* and *libraw*. The *rawkit* module provides a high-level Pythonic interface for developing raw photos, while the *libraw* module provides a CTypes based interface for interacting with the low-level LibRaw C APIs. Most of the time, developers will want to use the *rawkit* module instead of using *libraw* directly.

5.1 Contents

5.1.1 libraw package

Introduction

The *libraw* package contains low-level CTYPES based APIs for interfacing with LibRaw by LibRaw, LLC.

While this library can be used on its own to access the full functionality of LibRaw and develop raw photos, we recommend using the higher-level *rawkit* module, which provides a more pythonic interface to LibRaw.

Submodules

libraw.bindings — Low-level LibRaw bindings

The *libraw.bindings* module handles linking against the LibRaw binary.

class libraw.bindings.**LibRaw**

Bases: ctypes.CDLL

A ctypes.CDLL that links against *libraw.so* (or the equivalent on your platform).

Raises ImportError – If LibRaw cannot be found on your system, or linking fails.

version

A string representation of the version of LibRaw which we have linked against. eg.

```
"0.16.1-Release"
```

Returns The version

Return type str

version_number
A numeric representation of the version of LibRaw which we have linked against in `(Major, Minor, Patch)` form. eg.

```
(0, 16, 1)
```

Returns The version number

Return type 3 tuple

libraw.callbacks — LibRaw callback definitions

Warning: You will need to keep a reference to your callback functions for as long as you want to call them from C code, otherwise they may be garbage collected and lead to a segmentation fault.

libraw.callbacks.**data_callback**
alias of ctypes.CFunctionType

libraw.callbacks.**exif_parser_callback**
alias of ctypes.CFunctionType

libraw.callbacks.**memory_callback**
alias of ctypes.CFunctionType

libraw.callbacks.**progress_callback**
alias of ctypes.CFunctionType

libraw.errors — Pythonic error handling for LibRaw

exception libraw.errors.**BadCrop**
Bases: *libraw.errors.LibRawError*

The cropping coordinates specified are invalid (eg. the top left corner of the cropping rectangle is outside the image).

exception libraw.errors.**CanceledByCallback**
Bases: *libraw.errors.LibRawError*

Image processing was canceled because the progress callback requested it.

exception libraw.errors.**DataError**
Bases: *libraw.errors.LibRawError*

Data unpacking failed.

exception libraw.errors.**FileUnsupported**
Bases: *libraw.errors.LibRawError*

The file is not a raw file or is from an unsupported camera.

exception libraw.errors.**InputClosed**
Bases: *libraw.errors.LibRawError*

There is no input stream, or the input stream has been closed.

exception libraw.errors.**InsufficientMemory**
Bases: *libraw.errors.LibRawError*

Memory allocation failed.

exception libraw.errors.**LibRawError**
Bases: exceptions.Exception

A base exception class from which all other exceptions that originate in LibRaw inherit.

exception libraw.errors.**NoThumbnail**
Bases: *libraw.errors.LibRawError*

The raw file does not contain a thumbnail.

exception libraw.errors.**OutOfOrderCall**
Bases: *libraw.errors.LibRawError*

A LibRaw function depends on another function being called first and was invoked out of order.

exception libraw.errors.**RequestForNonexistentImage**
Bases: *libraw.errors.LibRawError*

The image file directory in the raw file which you are trying to access does not contain an image.

exception libraw.errors.**UnspecifiedError**
Bases: *libraw.errors.LibRawError*

Something bad happened, but we don't know what.

exception libraw.errors.**UnsupportedThumbnail**
Bases: *libraw.errors.LibRawError*

The thumbnail format is not supported.

class libraw.errors.**c_error**
Bases: ctypes.c_int

An error type for LibRaw (since LibRaw errors are ints and you can't distinguish between functions that return an error and functions that return an int that doesn't code for an error).

libraw.errors.**check_call**(*exit_code*, *func*, *arguments*)
Throws a Python error which corresponds to the given LibRaw exit code.

Parameters **exit_code** (*int*) – An exit code returned by a LibRaw function.

Raises

- *UnspecifiedError* – We're not sure what happened.
- *FileUnsupported* – The file is not a raw file that we recognize.
- *RequestForNonexistentImage* – The given IFD does not contain an image.
- *OutOfOrderCall* – Something was called out of order (eg. before data was unpacked)
- *NoThumbnail* – The image does not have a thumbnail.
- *UnsupportedThumbnail* – The embedded thumbnail format is unsupported.
- *InputClosed* – The input stream has been closed.

- *InsufficientMemory* – We're out of memory.
- *DataError* – The unpacking step failed.
- IOError – Reading was interrupted (or the file is corrupt).
- *CanceledByCallback* – A callback canceled the operation.
- *BadCrop* – The crop range was invalid.

libraw.errors.**raise_if_error**(*error_code*)

raise_if_error() raises a meaningful exception that corresponds to the given LibRaw integer return value.

Parameters **error_code** (*int*) – An exit code returned by a LibRaw function.

Raises

- *UnspecifiedError* – We're not sure what happened.
- *FileUnsupported* – The file is not a raw file that we recognize.
- *RequestForNonexistentImage* – The given IFD does not contain an image.
- *OutOfOrderCall* – Something was called out of order (eg. before data was unpacked)
- *NoThumbnail* – The image does not have a thumbnail.
- *UnsupportedThumbnail* – The embedded thumbnail format is unsupported.
- *InputClosed* – The input stream has been closed.
- *InsufficientMemory* – We're out of memory.
- *DataError* – The unpacking step failed.
- IOError – Reading was interrupted (or the file is corrupt).
- *CanceledByCallback* – A callback canceled the operation.
- *BadCrop* – The crop range was invalid.

libraw.structs — LibRaw struct definitions

class libraw.structs_16.**libraw_colordata_t**

Bases: _ctypes.Structure

Describes all color data of the image.

black
Structure/Union member

black_stat
Structure/Union member

cam_mul
Structure/Union member

cam_xyz
Structure/Union member

canon_ev
Structure/Union member

cblack
Structure/Union member

cmatrix
Structure/Union member

curve
Structure/Union member

data_maximum
Structure/Union member

flash_used
Structure/Union member

maximum
Structure/Union member

model2
Structure/Union member

phase_one_data
Structure/Union member

pre_mul
Structure/Union member

profile
Structure/Union member

profile_length
Structure/Union member

rgb_cam
Structure/Union member

white
Structure/Union member

class libraw.structs_16.**libraw_data_t**
Bases: _ctypes.Structure

A container which comprises the data structures that make up libraw's representation of a raw file.

color
Structure/Union member

idata
Structure/Union member

image
Structure/Union member

other
Structure/Union member

params
Structure/Union member

parent_class
Structure/Union member

process_warnings
Structure/Union member

progress_flags
Structure/Union member

rawdata
Structure/Union member

sizes
Structure/Union member

thumbnail
Structure/Union member

class libraw.structs_16.**libraw_decoder_info_t**
Bases: _ctypes.Structure

Describes a raw format decoder name and format.

decoder_flags
Structure/Union member

decoder_name
Structure/Union member

class libraw.structs_16.**libraw_image_sizes_t**
Bases: _ctypes.Structure

Describes the size of the image.

flip
Structure/Union member

height
Structure/Union member

iheight
Structure/Union member

iwidth
Structure/Union member

left_margin
Structure/Union member

mask
Structure/Union member

pixel_aspect
Structure/Union member

raw_height
Structure/Union member

raw_pitch
Structure/Union member

raw_width
Structure/Union member

top_margin
Structure/Union member

width
Structure/Union member

class libraw.structs_16.**libraw_imgother_t**
Bases: _ctypes.Structure

Information read from the raw file that is unnecessary for raw processing.

aperture
Structure/Union member

artist
Structure/Union member

desc
Structure/Union member

focal_len
Structure/Union member

gpsdata
Structure/Union member

iso_speed
Structure/Union member

shot_order
Structure/Union member

shutter
Structure/Union member

timestamp
Structure/Union member

class libraw.structs_16.**libraw_internal_output_params_t**
Bases: _ctypes.Structure

fuji_width
Structure/Union member

mix_green
Structure/Union member

raw_color
Structure/Union member

shrink
Structure/Union member

zero_is_bad
Structure/Union member

class libraw.structs_16.**libraw_iparams_t**
Bases: _ctypes.Structure

The primary parameters of the image.

cdesc
Structure/Union member

colors
Structure/Union member

dng_version
Structure/Union member

filters
Structure/Union member

is_foveon
Structure/Union member

make
Structure/Union member

model
Structure/Union member

raw_count
Structure/Union member

xtrans
Structure/Union member

class libraw.structs_16.**libraw_output_params_t**
Bases: _ctypes.Structure

Output parameters for processing the image with dcraw.

aber
Structure/Union member

adjust_maximum_thr
Structure/Union member

auto_bright_thr
Structure/Union member

bad_pixels
Structure/Union member

bright
Structure/Union member

ca_correc
Structure/Union member

cablue
Structure/Union member

camera_profile
Structure/Union member

cared
Structure/Union member

cclean
Structure/Union member

cfa_clean
Structure/Union member

cfa_green
Structure/Union member

cfaline
Structure/Union member

cropbox
Structure/Union member

dark_frame
Structure/Union member

dcb_enhance_fl
Structure/Union member

dcb_iterations
Structure/Union member

eeci_refine
Structure/Union member

es_med_passes
Structure/Union member

exp_correc
Structure/Union member

exp_preser
Structure/Union member

exp_shift
Structure/Union member

fbdd_noiserd
Structure/Union member

force_foveon_x3f
Structure/Union member

four_color_rgb
Structure/Union member

gamm
Structure/Union member

green_matching
Structure/Union member

green_thresh
Structure/Union member

greybox
Structure/Union member

half_size
Structure/Union member

highlight
Structure/Union member

lclean
Structure/Union member

linenoise
Structure/Union member

med_passes
Structure/Union member

no_auto_bright
Structure/Union member

no_auto_scale
Structure/Union member

no_interpolation
Structure/Union member

output_bps
Structure/Union member

output_color
Structure/Union member

output_profile
Structure/Union member

output_tiff
Structure/Union member

shot_select
Structure/Union member

sony_arw2_hack
Structure/Union member

straw_ycc
Structure/Union member

threshold
Structure/Union member

use_auto_wb
Structure/Union member

use_camera_matrix
Structure/Union member

use_camera_wb
Structure/Union member

use_fuji_rotate
Structure/Union member

use_rawspeed
Structure/Union member

user_black
Structure/Union member

user_cblack
Structure/Union member

user_flip
Structure/Union member

user_mul
Structure/Union member

user_qual
Structure/Union member

user_sat
Structure/Union member

wf_deband_treshold
Structure/Union member

wf_debanding
Structure/Union member

class libraw.structs_16.**libraw_processed_image_t**
Bases: _ctypes.Structure

A container for processed image data.

bits
Structure/Union member

colors
Structure/Union member

data
Structure/Union member

data_size
Structure/Union member

height
Structure/Union member

type
Structure/Union member

width
Structure/Union member

class libraw.structs_16.**libraw_rawdata_t**
Bases: _ctypes.Structure

Raw image data (after it has been unpacked) and a backup copy of color info used during post processing.

color
Structure/Union member

color3_image
Structure/Union member

color4_image
Structure/Union member

ioparams
Structure/Union member

iparams
Structure/Union member

ph1_black
Structure/Union member

raw_alloc
Structure/Union member

raw_image
Structure/Union member

sizes
Structure/Union member

class libraw.structs_16.**libraw_thumbnail_t**
Bases: _ctypes.Structure

Describes the thumbnail image embedded in the raw file.

tcolors
Structure/Union member

tformat
Structure/Union member

theight
Structure/Union member

thumb
Structure/Union member

tlength
Structure/Union member

twidth
Structure/Union member

class libraw.structs_16.**ph1_t**
Bases: _ctypes.Structure

Contains color data read by Phase One cameras.

black_off
Structure/Union member

format
Structure/Union member

key_off
Structure/Union member

split_col
Structure/Union member

t_black
Structure/Union member

tag_210
Structure/Union member

tag_21a
Structure/Union member

libraw.structs — LibRaw struct definitions

class libraw.structs_17.**libraw_canon_makernotes_t**
Bases: _ctypes.Structure

AverageBlackLevel
Structure/Union member

CanonColorDataSubVer
Structure/Union member

CanonColorDataVer
Structure/Union member

SpecularWhiteLevel
Structure/Union member

class libraw.structs_17.**libraw_colordata_t**
Bases: _ctypes.Structure

Describes all color data of the image.

FujiExpoMidPointShift
Structure/Union member

OlympusSensorCalibration
Structure/Union member

baseline_exposure
Structure/Union member

black
Structure/Union member

black_stat
Structure/Union member

cam_mul
Structure/Union member

cam_xyz
Structure/Union member

canon_ev
Structure/Union member

canon_makernotes
Structure/Union member

cblack
Structure/Union member

cmatrix
Structure/Union member

curve
Structure/Union member

data_maximum
Structure/Union member

digitalBack_color
Structure/Union member

dng_color
Structure/Union member

flash_used
Structure/Union member

maximum
Structure/Union member

model2
Structure/Union member

phase_one_data
Structure/Union member

pre_mul
Structure/Union member

profile
Structure/Union member

profile_length
Structure/Union member

rgb_cam
Structure/Union member

white
Structure/Union member

class libraw.structs_17.**libraw_data_t**
Bases: _ctypes.Structure

A container which comprises the data structures that make up libraw's representation of a raw file.

color
Structure/Union member

idata
Structure/Union member

image
Structure/Union member

lens
Structure/Union member

other
Structure/Union member

params
Structure/Union member

parent_class
Structure/Union member

process_warnings
Structure/Union member

progress_flags
Structure/Union member

rawdata
Structure/Union member

sizes
Structure/Union member

thumbnail
Structure/Union member

class libraw.structs_17.**libraw_decoder_info_t**
Bases: _ctypes.Structure

Describes a raw format decoder name and format.

decoder_flags
Structure/Union member

decoder_name
Structure/Union member

class libraw.structs_17.**libraw_dng_color_t**
Bases: _ctypes.Structure

calibration
Structure/Union member

colormatrix
Structure/Union member

illuminant
Structure/Union member

class libraw.structs_17.**libraw_dnglens_t**
Bases: _ctypes.Structure

MaxAp4MaxFocal
Structure/Union member

MaxAp4MinFocal
Structure/Union member

MaxFocal
Structure/Union member

MinFocal
Structure/Union member

class libraw.structs_17.**libraw_gps_info_t**
Bases: _ctypes.Structure

GPS data for the image.

altitude
Structure/Union member

altref
Structure/Union member

gpsparsed
Structure/Union member

gpsstatus
Structure/Union member

gpstimestamp
Structure/Union member

latitude
Structure/Union member

latref
Structure/Union member

longitude
Structure/Union member

longref
Structure/Union member

class libraw.structs_17.**libraw_image_sizes_t**
Bases: _ctypes.Structure

Describes the size of the image.

flip
Structure/Union member

height
Structure/Union member

iheight
Structure/Union member

iwidth
Structure/Union member

left_margin
Structure/Union member

mask
Structure/Union member

pixel_aspect
Structure/Union member

raw_height
Structure/Union member

raw_pitch
Structure/Union member

raw_width
Structure/Union member

top_margin
Structure/Union member

width
Structure/Union member

class libraw.structs_17.**libraw_imgother_t**
Bases: _ctypes.Structure

Information read from the raw file that is unnecessary for raw processing.

aperture
Structure/Union member

artist
Structure/Union member

desc
Structure/Union member

focal_len
Structure/Union member

gpsdata
Structure/Union member

iso_speed
Structure/Union member

parsed_gps
Structure/Union member

shot_order
Structure/Union member

shutter
Structure/Union member

timestamp
Structure/Union member

class libraw.structs_17.**libraw_internal_output_params_t**
Bases: _ctypes.Structure

fuji_width
Structure/Union member

mix_green
Structure/Union member

raw_color
Structure/Union member

shrink
Structure/Union member

zero_is_bad
Structure/Union member

class libraw.structs_17.**libraw_iparams_t**
Bases: _ctypes.Structure

The primary parameters of the image.

cdesc
Structure/Union member

colors
Structure/Union member

dng_version
Structure/Union member

filters
Structure/Union member

is_foveon
Structure/Union member

make
Structure/Union member

model
Structure/Union member

raw_count
Structure/Union member

software
Structure/Union member

xmpdata
Structure/Union member

xmplen
Structure/Union member

xtrans
Structure/Union member

xtrans_abs
Structure/Union member

class libraw.structs_17.**libraw_lensinfo_t**
Bases: _ctypes.Structure

EXIF_MaxAp
Structure/Union member

FocalLengthIn35mmFormat
Structure/Union member

Lens
Structure/Union member

LensMake
Structure/Union member

MaxAp4MaxFocal
Structure/Union member

MaxAp4MinFocal
Structure/Union member

MaxFocal
Structure/Union member

MinFocal
Structure/Union member

dng
Structure/Union member

makernotes
Structure/Union member

nikon
Structure/Union member

class libraw.structs_17.**libraw_makernotes_lens_t**
Bases: _ctypes.Structure

Adapter
Structure/Union member

AdapterID
Structure/Union member

Attachment
Structure/Union member

AttachmentID
Structure/Union member

CamID
Structure/Union member

CameraFormat
Structure/Union member

CameraMount
Structure/Union member

CanonFocalUnits
Structure/Union member

CurAp
Structure/Union member

CurFocal
Structure/Union member

FocalLengthIn35mmFormat
Structure/Union member

FocalType
Structure/Union member

Lens
Structure/Union member

LensFStops
Structure/Union member

LensFeatures_pre
Structure/Union member

LensFeatures_suf
Structure/Union member

LensFormat
Structure/Union member

LensID
Structure/Union member

LensMount
Structure/Union member

MaxAp
Structure/Union member

MaxAp4CurFocal
Structure/Union member

MaxAp4MaxFocal
Structure/Union member

MaxAp4MinFocal
Structure/Union member

MaxFocal
Structure/Union member

MinAp
Structure/Union member

MinAp4CurFocal
Structure/Union member

MinAp4MaxFocal
Structure/Union member

MinAp4MinFocal
Structure/Union member

MinFocal
Structure/Union member

Teleconverter
Structure/Union member

TeleconverterID
Structure/Union member

body
Structure/Union member

class libraw.structs_17.**libraw_nikonlens_t**
Bases: _ctypes.Structure

NikonEffectiveMaxAp
Structure/Union member

NikonLensFStops
Structure/Union member

NikonLensIDNumber
Structure/Union member

NikonLensType
Structure/Union member

NikonMCUVersion
Structure/Union member

class libraw.structs_17.**libraw_output_params_t**
Bases: _ctypes.Structure

Output parameters for processing the image with dcraw.

aber
Structure/Union member

adjust_maximum_thr
Structure/Union member

auto_bright_thr
Structure/Union member

bad_pixels
Structure/Union member

bright
Structure/Union member

ca_correc
Structure/Union member

cablue
Structure/Union member

camera_profile
Structure/Union member

cared
Structure/Union member

cclean
Structure/Union member

cfa_clean
Structure/Union member

cfa_green
Structure/Union member

cfaline
Structure/Union member

coolscan_nef_gamma
Structure/Union member

cropbox
Structure/Union member

dark_frame
Structure/Union member

dcb_enhance_fl
Structure/Union member

dcb_iterations
Structure/Union member

eeci_refine
Structure/Union member

es_med_passes
Structure/Union member

exp_correc
Structure/Union member

exp_preser
Structure/Union member

exp_shift
Structure/Union member

fbdd_noiserd
Structure/Union member

force_foveon_x3f
Structure/Union member

four_color_rgb
Structure/Union member

gamm
Structure/Union member

green_matching
Structure/Union member

green_thresh
Structure/Union member

greybox
Structure/Union member

half_size
Structure/Union member

highlight
Structure/Union member

lclean
Structure/Union member

linenoise
Structure/Union member

med_passes
Structure/Union member

no_auto_bright
Structure/Union member

no_auto_scale
Structure/Union member

no_interpolation
Structure/Union member

output_bps
Structure/Union member

output_color
Structure/Union member

output_profile
Structure/Union member

output_tiff
Structure/Union member

shot_select
Structure/Union member

sony_arw2_options
Structure/Union member

sony_arw2_posterization_thr
Structure/Union member

straw_ycc
Structure/Union member

threshold
Structure/Union member

use_auto_wb
Structure/Union member

use_camera_matrix
Structure/Union member

use_camera_wb
Structure/Union member

use_fuji_rotate
Structure/Union member

use_rawspeed
Structure/Union member

user_black
Structure/Union member

user_cblack
Structure/Union member

user_flip
Structure/Union member

user_mul
Structure/Union member

user_qual
Structure/Union member

user_sat
Structure/Union member

wf_deband_treshold
Structure/Union member

wf_debanding
Structure/Union member

x3f_flags
Structure/Union member

class libraw.structs_17.**libraw_processed_image_t**
Bases: _ctypes.Structure

A container for processed image data.

bits
Structure/Union member

colors
Structure/Union member

data
Structure/Union member

data_size
Structure/Union member

height
Structure/Union member

type
Structure/Union member

width
Structure/Union member

class libraw.structs_17.**libraw_rawdata_t**
Bases: _ctypes.Structure

Raw image data (after it has been unpacked) and a backup copy of color info used during post processing.

color
Structure/Union member

color3_image
Structure/Union member

color4_image
Structure/Union member

ioparams
Structure/Union member

iparams
Structure/Union member

ph1_cblack
Structure/Union member

ph1_rblack
Structure/Union member

raw_alloc
Structure/Union member

raw_image
Structure/Union member

sizes
Structure/Union member

class libraw.structs_17.**libraw_thumbnail_t**
Bases: _ctypes.Structure

Describes the thumbnail image embedded in the raw file.

tcolors
Structure/Union member

tformat
Structure/Union member

theight
Structure/Union member

thumb
Structure/Union member

tlength
Structure/Union member

twidth
Structure/Union member

class libraw.structs_17.**ph1_t**
Bases: _ctypes.Structure

Contains color data read by Phase One cameras.

black_col
Structure/Union member

black_row
Structure/Union member

format
Structure/Union member

key_off
Structure/Union member

split_col
Structure/Union member

split_row
Structure/Union member

t_black
Structure/Union member

tag_210
Structure/Union member

tag_21a
Structure/Union member

libraw.structs — LibRaw struct definitions

class libraw.structs_18.**libraw_P1_color_t**
Bases: _ctypes.Structure

romm_cam
Structure/Union member

class libraw.structs_18.**libraw_canon_makernotes_t**
Bases: _ctypes.Structure

AESetting
Structure/Union member

AFAreaHeights
Structure/Union member

AFAreaMode
Structure/Union member

AFAreaWidths
Structure/Union member

AFAreaXPositions
Structure/Union member

AFAreaYPositions
Structure/Union member

AFImageHeight
Structure/Union member

AFImageWidth
Structure/Union member

AFPoint
Structure/Union member

AFPointsInFocus
Structure/Union member

AFPointsInFocus1D
Structure/Union member

AFPointsInFocus30D
Structure/Union member

AFPointsInFocus5D
Structure/Union member

AFPointsSelected
Structure/Union member

AverageBlackLevel
Structure/Union member

BlackMaskBottomBorder
Structure/Union member

BlackMaskLeftBorder
Structure/Union member

BlackMaskRightBorder
Structure/Union member

BlackMaskTopBorder
Structure/Union member

CanonColorDataSubVer
Structure/Union member

CanonColorDataVer
Structure/Union member

ChannelBlackLevel
Structure/Union member

ContinuousDrive
Structure/Union member

ExposureMode
Structure/Union member

FlashActivity
Structure/Union member

FlashBits
Structure/Union member

FlashExposureLock
Structure/Union member

FlashGuideNumber
Structure/Union member

FlashMeteringMode
Structure/Union member

FlashMode
Structure/Union member

FlashOutput
Structure/Union member

FocusContinuous
Structure/Union member

FocusMode
Structure/Union member

HighlightTonePriority
Structure/Union member

ImageStabilization
Structure/Union member

ManualFlashOutput
Structure/Union member

MeteringMode
Structure/Union member

NumAFPoints
Structure/Union member

PrimaryAFPoint
Structure/Union member

SensorBottomBorder
Structure/Union member

SensorHeight
Structure/Union member

SensorLeftBorder
Structure/Union member

SensorRightBorder
Structure/Union member

SensorTopBorder
Structure/Union member

SensorWidth
Structure/Union member

SpecularWhiteLevel
Structure/Union member

SpotMeteringMode
Structure/Union member

ValidAFPoints
Structure/Union member

class libraw.structs_18.**libraw_colordata_t**
Bases: _ctypes.Structure

Describes all color data of the image.

LocalizedCameraModel
Structure/Union member

P1_color
Structure/Union member

UniqueCameraModel
Structure/Union member

WBCT_Coeffs
Structure/Union member

WB_Coeffs
Structure/Union member

baseline_exposure
Structure/Union member

black
Structure/Union member

black_stat
Structure/Union member

cam_mul
Structure/Union member

cam_xyz
Structure/Union member

canon_ev
Structure/Union member

cblack
Structure/Union member

ccm
Structure/Union member

cmatrix
Structure/Union member

curve
Structure/Union member

data_maximum
Structure/Union member

dng_color
Structure/Union member

dng_levels
Structure/Union member

flash_used
Structure/Union member

fmaximum
Structure/Union member

fnorm
Structure/Union member

linear_max
Structure/Union member

maximum
Structure/Union member

model2
Structure/Union member

phase_one_data
Structure/Union member

pre_mul
Structure/Union member

profile
Structure/Union member

profile_length
Structure/Union member

rgb_cam
Structure/Union member

white
Structure/Union member

class libraw.structs_18.**libraw_custom_camera_t**
Bases: _ctypes.Structure

bm
Structure/Union member

cf
Structure/Union member

flags
Structure/Union member

fsize
Structure/Union member

lf
Structure/Union member

lm
Structure/Union member

max
Structure/Union member

offset
Structure/Union member

rh
Structure/Union member

rm
Structure/Union member

rw
Structure/Union member

t_make
Structure/Union member

t_model
Structure/Union member

tm
Structure/Union member

class libraw.structs_18.**libraw_data_t**
Bases: _ctypes.Structure

A container which comprises the data structures that make up libraw's representation of a raw file.

color
Structure/Union member

idata
Structure/Union member

image
Structure/Union member

lens
Structure/Union member

makernotes
Structure/Union member

other
Structure/Union member

params
Structure/Union member

parent_class
Structure/Union member

process_warnings
Structure/Union member

progress_flags
Structure/Union member

rawdata
Structure/Union member

shootinginfo
Structure/Union member

sizes
Structure/Union member

thumbnail
Structure/Union member

class libraw.structs_18.**libraw_decoder_info_t**
Bases: _ctypes.Structure

Describes a raw format decoder name and format.

decoder_flags
Structure/Union member

decoder_name
Structure/Union member

class libraw.structs_18.**libraw_dng_color_t**
Bases: _ctypes.Structure

calibration
Structure/Union member

colormatrix
Structure/Union member

forwardmatrix
Structure/Union member

illuminant
Structure/Union member

class libraw.structs_18.**libraw_dng_levels_t**
Bases: _ctypes.Structure

analogbalance
Structure/Union member

dng_black
Structure/Union member

dng_blacklevel
Structure/Union member

dng_cblack
Structure/Union member

dng_whitelevel
Structure/Union member

class libraw.structs_18.**libraw_dnglens_t**
Bases: _ctypes.Structure

MaxAp4MaxFocal
Structure/Union member

MaxAp4MinFocal
Structure/Union member

MaxFocal
Structure/Union member

MinFocal
Structure/Union member

class libraw.structs_18.**libraw_fuji_info_t**
Bases: _ctypes.Structure

AFMode
Structure/Union member

ExrMode
Structure/Union member

FlashMode
Structure/Union member

FocusMode
Structure/Union member

FocusPixel
Structure/Union member

FrameHeight
Structure/Union member

FrameRate
Structure/Union member

FrameWidth
Structure/Union member

FujiAutoDynamicRange
Structure/Union member

FujiDevelopmentDynamicRange
Structure/Union member

FujiDynamicRange
Structure/Union member

FujiDynamicRangeSetting
Structure/Union member

FujiExpoMidPointShift
Structure/Union member

FujiFilmMode
Structure/Union member

ImageStabilization
Structure/Union member

Macro
Structure/Union member

Rating
Structure/Union member

ShutterType
Structure/Union member

WB_Preset
Structure/Union member

class libraw.structs_18.**libraw_gps_info_t**
Bases: _ctypes.Structure

GPS data for the image.

altitude
Structure/Union member

altref
Structure/Union member

gpsparsed
Structure/Union member

gpsstatus
Structure/Union member

gpstimestamp
Structure/Union member

latitude
Structure/Union member

latref
Structure/Union member

longitude
Structure/Union member

longref
Structure/Union member

class libraw.structs_18.**libraw_image_sizes_t**
Bases: _ctypes.Structure

Describes the size of the image.

flip
Structure/Union member

height
Structure/Union member

iheight
Structure/Union member

iwidth
Structure/Union member

left_margin
Structure/Union member

mask
Structure/Union member

pixel_aspect
Structure/Union member

raw_height
Structure/Union member

raw_pitch
Structure/Union member

raw_width
Structure/Union member

top_margin
Structure/Union member

width
Structure/Union member

class libraw.structs_18.**libraw_imgother_t**
Bases: _ctypes.Structure

Information read from the raw file that is unnecessary for raw processing.

FlashEC
Structure/Union member

aperture
Structure/Union member

artist
Structure/Union member

desc
Structure/Union member

focal_len
Structure/Union member

gpsdata
Structure/Union member

iso_speed
Structure/Union member

parsed_gps
Structure/Union member

shot_order
Structure/Union member

shutter
Structure/Union member

timestamp
Structure/Union member

class libraw.structs_18.**libraw_internal_output_params_t**
Bases: _ctypes.Structure

fuji_width
Structure/Union member

mix_green
Structure/Union member

raw_color
Structure/Union member

shrink
Structure/Union member

zero_is_bad
Structure/Union member

class libraw.structs_18.**libraw_iparams_t**
Bases: _ctypes.Structure

The primary parameters of the image.

cdesc
Structure/Union member

colors
Structure/Union member

dng_version
Structure/Union member

filters
Structure/Union member

guard
Structure/Union member

is_foveon
Structure/Union member

make
Structure/Union member

model
Structure/Union member

raw_count
Structure/Union member

software
Structure/Union member

xmpdata
Structure/Union member

xmplen
Structure/Union member

xtrans
Structure/Union member

xtrans_abs
Structure/Union member

class libraw.structs_18.**libraw_lensinfo_t**
Bases: _ctypes.Structure

EXIF_MaxAp
Structure/Union member

FocalLengthIn35mmFormat
Structure/Union member

InternalLensSerial
Structure/Union member

Lens
Structure/Union member

LensMake
Structure/Union member

LensSerial
Structure/Union member

MaxAp4MaxFocal
Structure/Union member

MaxAp4MinFocal
Structure/Union member

MaxFocal
Structure/Union member

MinFocal
Structure/Union member

dng
Structure/Union member

makernotes
Structure/Union member

nikon
Structure/Union member

class libraw.structs_18.**libraw_makernotes_lens_t**
Bases: _ctypes.Structure

Adapter
Structure/Union member

AdapterID
Structure/Union member

Attachment
Structure/Union member

AttachmentID
Structure/Union member

CamID
Structure/Union member

CameraFormat
Structure/Union member

CameraMount
Structure/Union member

CanonFocalUnits
Structure/Union member

CurAp
Structure/Union member

CurFocal
Structure/Union member

FocalLengthIn35mmFormat
Structure/Union member

FocalType
Structure/Union member

FocusRangeIndex
Structure/Union member

Lens
Structure/Union member

LensFStops
Structure/Union member

LensFeatures_pre
Structure/Union member

LensFeatures_suf
Structure/Union member

LensFormat
Structure/Union member

LensID
Structure/Union member

LensMount
Structure/Union member

MaxAp
Structure/Union member

MaxAp4CurFocal
Structure/Union member

MaxAp4MaxFocal
Structure/Union member

MaxAp4MinFocal
Structure/Union member

MaxFocal
Structure/Union member

MinAp
Structure/Union member

MinAp4CurFocal
Structure/Union member

MinAp4MaxFocal
Structure/Union member

MinAp4MinFocal
Structure/Union member

MinFocal
Structure/Union member

MinFocusDistance
Structure/Union member

Teleconverter
Structure/Union member

TeleconverterID
Structure/Union member

body
Structure/Union member

class libraw.structs_18.**libraw_makernotes_t**
Bases: _ctypes.Structure

canon
Structure/Union member

fuji
Structure/Union member

olympus
Structure/Union member

sony
Structure/Union member

class libraw.structs_18.**libraw_nikon_makernotes_t**
Bases: _ctypes.Structure

AFAreaHeight
Structure/Union member

AFAreaMode
Structure/Union member

AFAreaWidth
Structure/Union member

AFAreaXPposition
Structure/Union member

AFAreaYPosition
Structure/Union member

AFImageHeight
Structure/Union member

AFImageWidth
Structure/Union member

AFPoint
Structure/Union member

AFPointsInFocus
Structure/Union member

AFPointsUsed
Structure/Union member

ActiveDLighting
Structure/Union member

ContrastDetectAF
Structure/Union member

ContrastDetectAFInFocus
Structure/Union member

ExposureBracketValue
Structure/Union member

ExternalFlashExposureComp
Structure/Union member

ExternalFlashFlags
Structure/Union member

FlashColorFilter
Structure/Union member

FlashControlCommanderMode
Structure/Union member

FlashExposureBracketValue
Structure/Union member

FlashExposureCompensation
Structure/Union member

FlashExposureCompensation2
Structure/Union member

FlashExposureCompensation3
Structure/Union member

FlashExposureCompensation4
Structure/Union member

FlashFirmware
Structure/Union member

FlashFocalLength
Structure/Union member

FlashGNDistance
Structure/Union member

FlashGroupControlMode
Structure/Union member

FlashGroupOutputAndCompensation
Structure/Union member

FlashMode
Structure/Union member

FlashOutputAndCompensation
Structure/Union member

FlashSetting
Structure/Union member

FlashSource
Structure/Union member

FlashType
Structure/Union member

FocusMode
Structure/Union member

ImageStabilization
Structure/Union member

PhaseDetectAF
Structure/Union member

PrimaryAFPoint
Structure/Union member

ShootingMode
Structure/Union member

VRMode
Structure/Union member

VibrationReduction
Structure/Union member

class libraw.structs_18.**libraw_nikonlens_t**
Bases: _ctypes.Structure

NikonEffectiveMaxAp
Structure/Union member

NikonLensFStops
Structure/Union member

NikonLensIDNumber
Structure/Union member

NikonLensType
Structure/Union member

NikonMCUVersion
Structure/Union member

class libraw.structs_18.**libraw_olympus_makernotes_t**
Bases: _ctypes.Structure

AFAreas
Structure/Union member

AFPoint
Structure/Union member

AFPointSelected
Structure/Union member

AFResult
Structure/Union member

AutoFocus
Structure/Union member

ColorSpace
Structure/Union member

FocusMode
Structure/Union member

ImageStabilization
Structure/Union member

OlympusCropID
Structure/Union member

OlympusFrame
Structure/Union member

OlympusSensorCalibration
Structure/Union member

class libraw.structs_18.**libraw_output_params_t**
Bases: _ctypes.Structure

Output parameters for processing the image with dcraw.

aber
Structure/Union member

adjust_maximum_thr
Structure/Union member

auto_bright_thr
Structure/Union member

bad_pixels
Structure/Union member

bright
Structure/Union member

ca_correc
Structure/Union member

cablue
Structure/Union member

camera_profile
Structure/Union member

cared
Structure/Union member

cclean
Structure/Union member

cfa_clean
Structure/Union member

cfa_green
Structure/Union member

cfaline
Structure/Union member

coolscan_nef_gamma
Structure/Union member

cropbox
Structure/Union member

custom_camera_strings
Structure/Union member

dark_frame
Structure/Union member

dcb_enhance_fl
Structure/Union member

dcb_iterations
Structure/Union member

eeci_refine
Structure/Union member

es_med_passes
Structure/Union member

exp_correc
Structure/Union member

exp_preser
Structure/Union member

exp_shift
Structure/Union member

fbdd_noiserd
Structure/Union member

four_color_rgb
Structure/Union member

gamm
Structure/Union member

green_matching
Structure/Union member

green_thresh
Structure/Union member

greybox
Structure/Union member

half_size
Structure/Union member

highlight
Structure/Union member

lclean
Structure/Union member

linenoise
Structure/Union member

med_passes
Structure/Union member

no_auto_bright
Structure/Union member

no_auto_scale
Structure/Union member

no_interpolation
Structure/Union member

output_bps
Structure/Union member

output_color
Structure/Union member

output_profile
Structure/Union member

output_tiff
Structure/Union member

p4shot_order
Structure/Union member

raw_processing_options
Structure/Union member

shot_select
Structure/Union member

sony_arw2_posterization_thr
Structure/Union member

threshold
Structure/Union member

use_auto_wb
Structure/Union member

use_camera_matrix
Structure/Union member

use_camera_wb
Structure/Union member

use_dngsdk
Structure/Union member

use_fuji_rotate
Structure/Union member

use_rawspeed
Structure/Union member

user_black
Structure/Union member

user_cblack
Structure/Union member

user_flip
Structure/Union member

user_mul
Structure/Union member

user_qual
Structure/Union member

user_sat
Structure/Union member

wf_deband_treshold
Structure/Union member

wf_debanding
Structure/Union member

class libraw.structs_18.**libraw_pentax_makernotes_t**
Bases: _ctypes.Structure

AFPointMode
Structure/Union member

AFPointSelected
Structure/Union member

AFPointsInFocus
Structure/Union member

DriveMode
Structure/Union member

FocusMode
Structure/Union member

SRResult
Structure/Union member

ShakeReduction
Structure/Union member

class libraw.structs_18.**libraw_processed_image_t**
Bases: _ctypes.Structure

A container for processed image data.

bits
Structure/Union member

colors
Structure/Union member

data
Structure/Union member

data_size
Structure/Union member

height
Structure/Union member

type
Structure/Union member

width
Structure/Union member

class libraw.structs_18.**libraw_rawdata_t**
Bases: _ctypes.Structure

Raw image data (after it has been unpacked) and a backup copy of color info used during post processing.

color
Structure/Union member

color3_image
Structure/Union member

color4_image
Structure/Union member

float3_image
Structure/Union member

float4_image
Structure/Union member

float_image
Structure/Union member

ioparams
Structure/Union member

iparams
Structure/Union member

ph1_cblack
Structure/Union member

ph1_rblack
Structure/Union member

raw_alloc
Structure/Union member

raw_image
Structure/Union member

sizes
Structure/Union member

class libraw.structs_18.**libraw_shootinginfo_t**
Bases: _ctypes.Structure

AFPoint
Structure/Union member

BodySerial
Structure/Union member

DriveMode
Structure/Union member

ExposureMode
Structure/Union member

FocusMode
Structure/Union member

ImageStabilization
Structure/Union member

InternalBodySerial
Structure/Union member

MeteringMode
Structure/Union member

class libraw.structs_18.**libraw_sony_info_t**
Bases: _ctypes.Structure

SonyCameraType
Structure/Union member

class libraw.structs_18.**libraw_thumbnail_t**
Bases: _ctypes.Structure

Describes the thumbnail image embedded in the raw file.

tcolors
Structure/Union member

tformat
Structure/Union member

theight
Structure/Union member

thumb
Structure/Union member

tlength
Structure/Union member

twidth
Structure/Union member

class libraw.structs_18.**ph1_t**
Bases: _ctypes.Structure

Contains color data read by Phase One cameras.

black_col
Structure/Union member

black_row
Structure/Union member

format
Structure/Union member

key_off
Structure/Union member

split_col
Structure/Union member

split_row
Structure/Union member

t_black
Structure/Union member

tag_210
Structure/Union member

tag_21a
Structure/Union member

class libraw.structs_18.**xtrans_params**
Bases: _ctypes.Structure

line_width
Structure/Union member

maxDiff
Structure/Union member

max_bits
Structure/Union member

min_value
Structure/Union member

q_points
Structure/Union member

q_table
Structure/Union member

raw_bits
Structure/Union member

total_values
Structure/Union member

5.1.2 rawkit package

Introduction

The *rawkit* module contains high-level APIs for manipulating raw photos using the low-level *libraw* module (which in turn uses the even lower-level LibRaw C library).

Eg. quickly processing a raw Canon CR2 file without using the camera white balance and saving it as a PPM image might look like this:

```
from rawkit.raw import Raw
from rawkit.options import WhiteBalance

with Raw(filename='some/raw/image.CR2') as raw:
  raw.options.white_balance = WhiteBalance(camera=False, auto=True)
  raw.save(filename='some/destination/image.ppm')
```

rawkit.**VERSION = '0.6.0'**
The current version of the *rawkit* package.

Submodules

rawkit.errors — Errors thrown by rawkit

These errors are thrown by various rawkit functions and methods when things go wrong. They will only be raised by rawkit; for lower level errors raised by the underlying libraw bindings, see *libraw.errors*.

exception rawkit.errors.**InvalidFileType**
Bases: exceptions.ValueError

Raised when an invalid file type or file extension is passed to a rawkit method.

exception rawkit.errors.**NoFileSpecified**
Bases: exceptions.ValueError

Raised when the method or function excpects a *filename* argument, but no file name (or a value of *None*) was specified.

rawkit.metadata — Metadata structures

class rawkit.metadata.**Metadata**(*aperture*, *timestamp*, *shutter*, *flash*, *focal_length*, *height*, *iso*, *make*, *model*, *orientation*, *width*)
Bases: tuple

Common metadata for a photo.

Orientation matches the values from the EXIF 2.3 specification:

1 - The 0th row is at the visual top of the image, and the 0th column is the visual left-hand side.

2 - The 0th row is at the visual top of the image, and the 0th column is the visual right-hand side.

3 - The 0th row is at the visual bottom of the image, and the 0th column is the visual right-hand side.

4 - The 0th row is at the visual bottom of the image, and the 0th column is the visual left-hand side.

5 - The 0th row is the visual left-hand side of the image, and the 0th column is the visual top.

6 - The 0th row is the visual right-hand side of the image, and the 0th column is the visual top.

7 - The 0th row is the visual right-hand side of the image, and the 0th column is the visual bottom.

8 - The 0th row is the visual left-hand side of the image, and the 0th column is the visual bottom.

aperture
Alias for field number 0

flash
Alias for field number 3

focal_length
Alias for field number 4

height
Alias for field number 5

iso
Alias for field number 6

make
Alias for field number 7

model
Alias for field number 8

orientation
Alias for field number 9

shutter
Alias for field number 2

timestamp
Alias for field number 1

width
Alias for field number 10

rawkit.options — High level options for processing raw files

class rawkit.options.**Options**(*attrs=None*)
Bases: object

Represents a set of options which can be used when processing raw data.

Parameters **attrs** (*dict*) – A subscriptable object from which to take the initial state of the options object.

adjust_maximum_threshold
Automatically adjusts the maximum pixel value based on per channel maximum data.

Note: If this value is set above 0.99999, the default value will be used instead. If it is set below 0.00001, no adjustment will happen.

Type float

Default 0.75

Dcraw None

Libraw libraw.structs.libraw_output_params_t.adjust_maximum_thr

auto_brightness
Set the brightness automatically based on the image histogram and the *auto_brightness_threshold*.

Type boolean

Default True

Dcraw -W

Libraw libraw.structs.libraw_output_params_t.no_auto_bright

auto_brightness_threshold
The allowable percentage of clipped pixels when *auto_brightness* is used.

Type float

Default 0.001 (0.1%)

Dcraw None

Libraw `libraw.structs.libraw_output_params_t.auto_bright_thr`

auto_stretch
Stretches images taken on cameras with non-square pixels to the correct aspect ratio. For Fuji Super CCD cameras, rotates the image 45 degrees. This guarantees that the output pixels share a 1:1 correspondence with the raw pixels.

Type `boolean`

Default True

Dcraw `-j`

Libraw `libraw.structs.libraw_output_params_t.use_fuji_rotate`

bad_pixels_file
Points to a bad pixels map in dcraw format –

```
column row unix-timestamp\n
```

Type `str`

Default None

Dcraw `-P`

Libraw `libraw.structs.libraw_output_params_t.bad_pixels`

bps
Set the bits per sample used for the photo (8 or 16). Setting this to 16 is effectively the same as running dcraw with the `-4` option.

Type `int`

Default 8

Dcraw `-4`

Libraw `libraw.structs.libraw_output_params_t.output_bps`

brightness
Sets the brightness level by dividing the white level by this value. This is ignored if *auto_brightness* is `True`.

Type `float`

Default 1.0

Dcraw `-b`

Libraw `libraw.structs.libraw_output_params_t.bright`

chromatic_aberration
A Red-Blue scale factor that's used to correct for chromatic aberration by scaling the respective channels.

eg.

```
# (red_scale, blue_scale)
raw.options.chromatic_aberration = (0.999, 1.001)
```

Type `double tuple`

Default (1, 1)

Dcraw `-C`

Libraw `libraw.structs.libraw_output_params_t.aber`

colorspace
Sets the colorspace used for the output image. Supported colorspaces are defined as constants in *rawkit.options.colorspaces*.

Type `int`

Default `rawkit.options.colorspaces.srgb`

Dcraw `-o`

Libraw `libraw.structs.libraw_output_params_t.output_color`

cropbox
Crops the image.

Type `4 float tuple`

Default None

Dcraw None

Libraw `libraw.structs.libraw_output_params_t.cropbox`

dark_frame
A dark frame in 16-bit PGM format. This may either be a path to an existing file, or an instance of *rawkit.raw.DarkFrame*.

Type *rawkit.raw.DarkFrame* `str`

Default None

Dcraw `-K`

Libraw `libraw.structs.libraw_output_params_t.dark_frame`

darkness
Raise the black level of a photo.

Type `int`

Default None

Dcraw `-k`

Libraw `libraw.structs.libraw_output_params_t.user_black`

gamma
Sets the gamma-curve of the photo. The two values in the tuple correspond to:

- gamma[0] — Correction function power (inverted Gamma power, γ^{-1})
- gamma[1] — toe-slope (ϕ)

For a simple power curve, set the toe-slope to zero.

Type `2 double tuple`

Default None

Dcraw `-g`

Libraw `libraw.structs.libraw_output_params_t.gamm`

green_matching
Performs a second post-processing pass to correct for green channel imbalance.

Type boolean

Default False

Dcraw None

Libraw libraw.structs.libraw_output_params_t.green_matching

half_size
When developing the image, output it at 50% size. This makes developing preview images much faster.

Type boolean

Default False

Dcraw -h

Libraw libraw.structs.libraw_output_params_t.half_size

highlight_mode
The mode for dealing with highlights in the image. Some constants have been defined in *rawkit.options.highlight_modes* to make things easier, or you can set an integer directly.

Integers that are greater than or equal to 3 will attempt to reconstruct highlights. Lower numbers favor whites, and higher colors favor colors. rawkit.options.RECONSTRUCT (5) is a good compromise.

Type int

Default rawkit.options.highlight_modes.clip

Dcraw -H

Libraw libraw.structs.libraw_output_params_t.highlight

input_profile
Path to an ICC color profile file containing the input profile. Only used if the version of LibRaw that you're linking against was compiled with LCMS support.

Note that LibRaw defines a magic string, 'embed', which causes it to use the profile embedded in the raw image if present. This is the same as setting the *use_camera_profile* option.

Type string

Default None

Dcraw -o -p

Libraw libraw.structs.libraw_output_params_t.camera_profile

interpolation
Sets the interpolation algorithm.

Type *rawkit.options.interpolation*

Default *ahd*

Dcraw -q

Libraw libraw.structs.libraw_output_params_t.user_qual

keys()
A list of keys which have a value other than None and which have been set by the user (even if those options are set to the default value).

Returns List of option keys which have been set.

Return type tuple

median_filter_passes
Useful for cleaning up color artifacts by running a 3x3 median filter over the R-G and B-G channels.

Type int

Default 0

Dcraw -m

Libraw libraw.structs.libraw_output_params_t.med_passes

noise_threshold
Sets the threshold for noise reduction using wavelet denoising.

Type float

Default None

Dcraw -n

Libraw libraw.structs.libraw_output_params_t.threshold

output_profile
Path to an ICC color profile file containing the output profile. Only used if the version of LibRaw that you're linking against was compiled with LCMS support.

Type string

Default None

Dcraw -o -p

Libraw libraw.structs.libraw_output_params_t.output_profile

rgbg_interpolation
Determines if we should use four channel RGB interpolation.

Type boolean

Default False

Dcraw -f

Libraw libraw.structs.libraw_output_params_t.four_color_rgb

rotation
Rotates the image by the given number of degrees. Must be a multiple of 90 (0, 90, 180, 270, etc).

The default (None) is to use the rotation provided by the camera.

Type int

Default None

Dcraw -t

Libraw libraw.structs.libraw_output_params_t.user_flip

saturation
Determines the saturation level of the output image.

Type int

Default None

Dcraw -S

Libraw `libraw.structs.libraw_output_params_t.user_sat`

shot
Selects the shot to process for raw images that contain multiple images.

Type `int`

Default 0

Dcraw `-s`

Libraw `libraw.structs.libraw_output_params_t.shot_select`

use_camera_matrix
Use the color matrix from the raw's metadata. Only affects Olympus, Leaf, and Phase One cameras (and DNG files).

Note that we differ from the LibRaw defaults on this option. LibRaw defaults to true if the photo is in DNG format or the camera white balance is being used, and false otherwise. rawkit always defaults to true.

Type `boolean`

Default True

Dcraw `+M -M`

Libraw `libraw.libraw_output_params_t.use_camera_matrix`

use_camera_profile
True if we should use the embedded camera profile (if present in the raw file and we're linking against a version of LibRaw with LCMS support).

Type `boolean`

Default True

Dcraw `-o -p`

Libraw `libraw.structs.libraw_output_params_t.camera_profile`

values()
The values of all options which appear in *keys()*.

Returns List of options values.

Return type tuple

white_balance
The white balance of the image.

Type *rawkit.options.WhiteBalance*

Default WhiteBalance(auto=True, camera=True)

Dcraw `-a -w -A -r`

Libraw `libraw.structs.libraw_output_params_t.use_auto_wb` `libraw.structs.libraw_output_params_t.use_camera_wb` `libraw.structs.libraw_output_params_t.greybox` `libraw.structs.libraw_output_params_t.user_mul`

class rawkit.options.**WhiteBalance**
Bases: *rawkit.options.WhiteBalance*

Represents the white balance of a photo. If the camera white balance is used, but not present, we fallback to the other options. Other options white balance multipliers stack (eg. you can use auto white balance, and then specify a manual `rgbg` multiplier on top of that).

Parameters

- **auto** (*boolean*) – Determines if we should automatically set the WB.
- **camera** (*boolean*) – Causes us to use the camera defined WB if present.
- **greybox** (*4 int tuple*) – Set the WB based on a neutral grey region of the image.
- **rgbg** (*4 float tuple*) – Set the WB manually based on an RGBG channel multiplier.

Returns A white balance object.

Return type *WhiteBalance*

rawkit.options.**colorspaces = ColorSpaces(raw=0, srgb=1, adobe_rgb=2, wide_gammut_rgb=3, koc**

Constants for setting the colorspace.

- `raw_color` — Raw colorspace (unique to each camera)
- `srgb` — sRGB D65 (default colorspace)
- `adobe_rgb` — Adobe RGB (1998) D65
- `wide_gammut_rgb` — Wide Gamut RGB D65
- `kodak_prophoto_rgb` — Kodak ProPhoto RGB D65
- `xyz` — XYZ colorspace

rawkit.options.**gamma_curves = GammaCurves(linear=[1, 1], bt709=[0.45004500450045004, 4.5],**

Gamma curves for a few common color profiles.

- `linear` — A basic linear transfer function.
- `bt709` — The BT.709 (Rec. 709) curve used by HDTVs (uses the median power of sRGB, and a similar but shifted transfer function).
- `srgb` — The sRGB gamma curve (uses the max power to account for linear discontinuity and to attain the standard *IEC 61966-2-1* solution $K_0 \approx 0.04045 $).
- `adobe_rgb` — The correction function power for the Adobe RGB colorspace. The toe-slope is left off.

rawkit.options.**highlight_modes = HighlightMode(clip=0, ignore=1, blend=2, reconstruct=5)**

Constants for setting the highlight mode.

- `clip` — Clip all highlights to white (default).
- `ignore` — Leave highlights unclipped.
- `blend` — Blend clipped and unclipped highlights.
- `reconstruct` — A good average value for reconstruction of clipped highlights which compromises between favoring whites and favoring colors.

rawkit.options.**interpolation = InterpolationAlgo(linear=0, vng=1, ppg=2, ahd=3, dcb=4, modi**

***Constants for setting the interpolation algorithm* – 0. Linear**

1. VNG
2. PPG
3. AHD
4. DCB

5. Modified AHD
6. AFD
7. VCD
8. Mixed VCD and Modified AHD
9. LMMSE
10. AMaZE

Modified AHD (5) through LMMSE (9) are only useful if you're using a version of LibRaw with the "LibRaw Demosaic Pack GPL2" built in and AMaZE (10) is only useful if LibRaw was built with the "LibRaw Demosaic Pack GPL3". If you attepmt to use an interpolation method that's not built into your version of LibRaw, it will silently fallback to AHD.

Usage example:

```
from rawkit.raw import Raw
from rawkit.options import interpolation

with Raw(filename="RawFile.CR2") as raw:
    raw.options.interpolation = interpolation.ahd
    raw.save("RawFile.ppm")
```

class rawkit.options.**option**(*param=None*, *ctype=None*)
Bases: object

The *option* decorator is an internal decorator which allows you to define an option in a clean manner (specifying its name and how it maps to the libraw params).

param_writer(*func*)

setter(*func*)

write_param(*obj*, *params*)

rawkit.orientation.**get_orientation**(*data*)

rawkit.raw — High-level raw file API

class rawkit.raw.**DarkFrame**(*filename=None*)
Bases: *rawkit.raw.Raw*

Represents a dark frame—a raw photo taken in low light which can be subtracted from another photos raw data.

Creates a temporary file which is not cleaned up until the dark frame is closed.

cleanup()
Cleanup temp files.

close()
Free the underlying raw representation and cleanup temp files.

name
A tempfile in a unique directory.

Returns The name of a temp file.

Return type str

save(*filename=None*, *filetype='ppm'*)
Save the image data, defaults to using a temp file.

Parameters

- **filename** (*str*) – The name of an image file to save.
- **filetype** (*output_file_types*) – The type of file to output.

Raises *rawkit.errors.InvalidFileType* – If *filetype* is not of type *output_file_types*.

class rawkit.raw.**Raw**(*filename=None*)

Bases: object

Represents a raw file (of any format) and exposes development options to the user.

For example, the basic workflow (open a file, process the file, save the file) looks like this:

```
from rawkit.raw import Raw
from rawkit.options import WhiteBalance

with Raw(filename='some/raw/image.CR2') as raw:
    raw.options.white_balance = WhiteBalance(camera=False, auto=True)
    raw.save(filename='some/destination/image.ppm')
```

Parameters **filename** (*str*) – The name of a raw file to load.

Returns A raw object.

Return type *Raw*

Raises

- *rawkit.errors.NoFileSpecified* – If *filename* is None.
- *libraw.errors.FileUnsupported* – If the specified file is not a supported raw type.
- *libraw.errors.InsufficientMemory* – If we run out of memory while loading the raw file.
- IOError – If the file does not exist, or cannot be opened (eg. incorrect permissions).

as_array()

Get a NumPy array of the raw image data

Returns

A NumPy array of bayer pixel data structured as a list of rows, or array([]) if there is no bayer data. The margin with calibration pixels is always included. For example, if the color format is *RGGB*, the array would be of the format:

```
array([
    [R, G, R, G, ...],
    [G, B, G, B, ...],
    [R, G, R, G, ...],
    ...
])
```

Return type array

bayer_data(*include_margin=False*)

Get the bayer data and color_description for an image.

Returns

Tuple of bayer data and color filter array. This is a convenience method to return *rawkit.raw.Raw.raw_image* and *rawkit.raw.Raw.color_filter_array* as a single tuple.

Return type tuple

close()
Free the underlying raw representation.

color(*y*, *x*)
Get the active color of a pixel of bayer data.

Parameters

- **y** (*int*) – the y coordinate (or row) of the pixel
- **x** (*int*) – the x coordinate (or column) of the pixel

Returns Character representing the color, such as 'R' for red.

Return type str

color_description
Get the color_description of an image.

Returns 4 character string representing color format, such as 'RGGB'.

Return type str

color_filter
EXPERIMENTAL – This method only supports bayer filters for the time being. It will be incorrect when used with other types of sensors.

Get the color filter array for the camera sensor.

Returns

2D array representing the color format array pattern. For example, the typical 'RGGB' pattern of abayer sensor would be of the format:

```
[
    ['R', 'G'],
    ['G', 'B'],
]
```

Return type list

color_filter_array
EXPERIMENTAL – This method only supports bayer filters for the time being. It will be incorrect when used with other types of sensors.

Get the color filter array for the camera sensor.

Returns

Numpy array representing the color format array pattern. For example, the typical 'RGGB' pattern of abayer sensor would be of the format:

```
array([
    ['R', 'G'],
    ['G', 'B'],
])
```

Return type list

data_pointer()

metadata
Common metadata for the photo

Returns A metadata object.

Return type *rawkit.metadata.Metadata*

process()
Process the raw data based on `self.options`.

Raises

- *libraw.errors.DataError* – If invalid or corrupt data is encountered in the data struct.
- *libraw.errors.BadCrop* – If the image has been cropped poorly (eg. the edges are outside of the image bounds, or the crop box coordinates don't make sense).
- *libraw.errors.InsufficientMemory* – If we run out of memory while processing the raw file.

raw_image(*include_margin=False*)
Get the bayer data for an image if it exists.

Parameters **include_margin** (*bool*) – Include margin with calibration pixels.

Returns

2D array of bayer pixel data structured as a list of rows, or [] if there is no bayer data. For example, if the color format is *RGGB*, the array would be of the format:

```
[
    [R, G, R, G, ...],
    [G, B, G, B, ...],
    [R, G, R, G, ...],
    ...
]
```

Return type list

save(*filename*, *filetype=None*)
Save the image data as a new PPM or TIFF image.

Parameters

- **filename** (*str*) – The name of an image file to save.
- **filetype** (*output_file_types*) – The type of file to output. By default, guess based on the filename, falling back to PPM.

Raises

- *rawkit.errors.NoFileSpecified* – If *filename* is `None`.
- *rawkit.errors.InvalidFileType* – If *filetype* is not None or in *output_file_types*.

save_thumb(*filename=None*)
Save the thumbnail data.

Parameters **filename** (*str*) – The name of an image file to save.

Raises *rawkit.errors.NoFileSpecified* – If *filename* is `None`.

thumbnail_to_buffer()
Convert the thumbnail data as an RGB buffer.

Returns RGB data of the thumbnail.

Return type bytearray

to_buffer()
Convert the image to an RGB buffer.

Returns RGB data of the image.

Return type bytearray

unpack()
Unpack the raw data.

unpack_thumb()
Unpack the thumbnail data.

Raises

- *libraw.errors.NoThumbnail* – If the raw file does not contain a
- thumbnail.
- *libraw.errors.UnsupportedThumbnail* – If the thumbnail format is
- unsupported.

rawkit.raw.**output_file_types = OutputFileType(ppm='ppm', tiff='tiff')**
Constants for setting the output filetype.

- ppm — PGM data file.
- tiff — TIFF file.

rawkit.util — Utility functions

These functions perform helpful tasks which don't really fit anywhere else such as searching for Raw files on the disk, or checking what cameras are supported by LibRaw.

rawkit.util.**camera_list**()
Return a list of cameras which are supported by the currently linked version of LibRaw.

Returns A list of supported cameras.

Return type str array

rawkit.util.**discover**(*path*)
Recursively search for raw files in a given directory.

Parameters **path** (*str*) – A tree to recursively search.

CHAPTER 6

Indices and tables

- genindex
- modindex
- search

Python Module Index

l

r

Index

A

B

C

D

E

F

G

H

I

K

L

M

N

O

P

T

U

V

W

X

Z