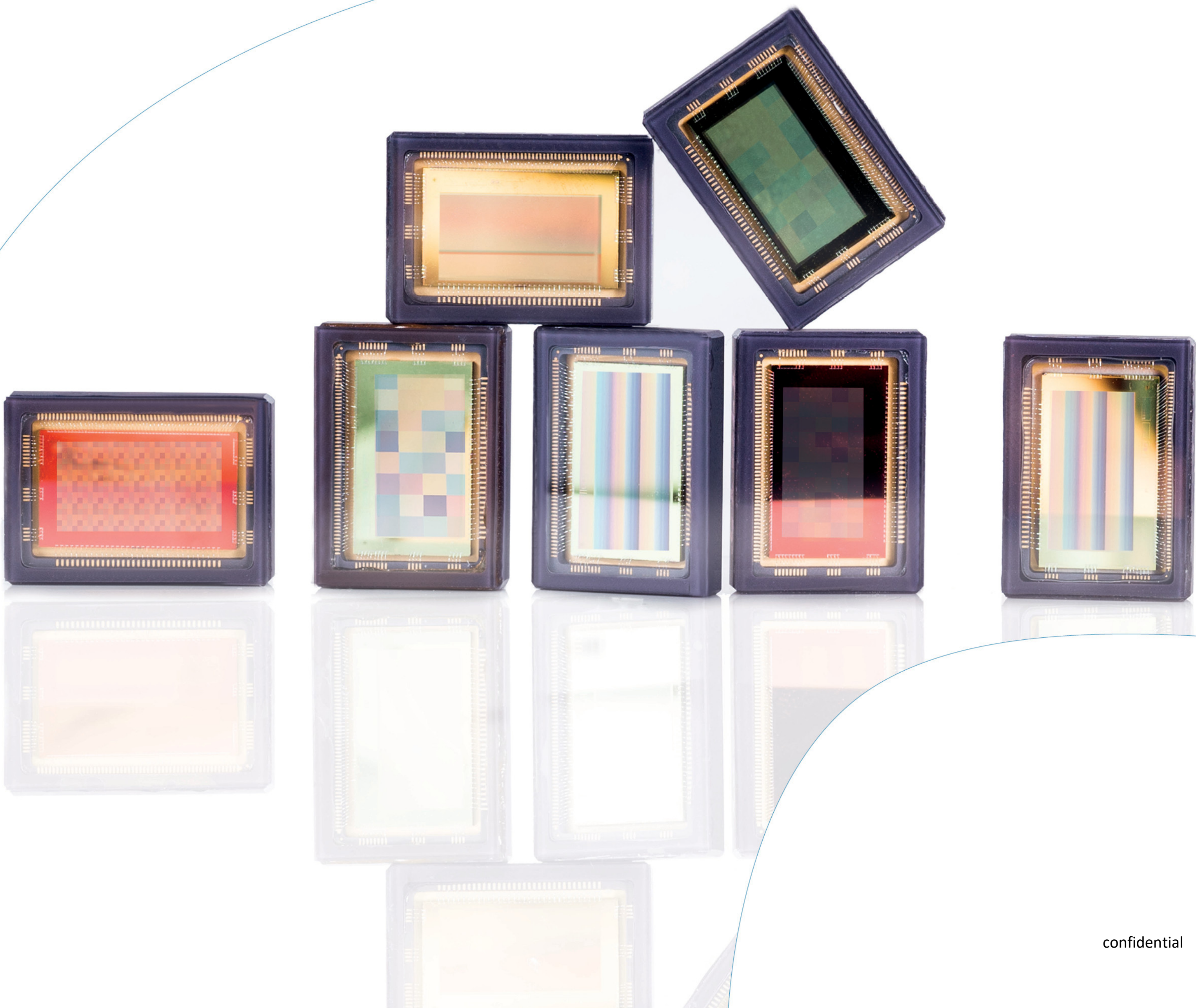
imec

embracing a better life

Calibration Files

Reference Manual



confidential

Revision history

V1.0	May. 3, 2016	First version of the document.
V1.1	Aug. 29, 2017	Updated description of Tag correction_matrices.correction_matrix.
V1.2	Oct. 25, 2017	Updated fields of Tag sensor_info and Tag calibration_info.
V1.3	Mar. 21, 2019	Updated calibration_info and optical_component.
V1.4	Jul. 16, 2019	Removed outdated information and added missing info on peak.
V1.5	Oct. 15, 2019	Updated support contact.
V1.6	May 18, 2020	Update to calibration_info and optical_component.
V1.7	Feb. 10, 2021	Update to sensor_info and correction_matrix.
V1.8	Mar. 25, 2021	Review of reference manual, fixing type of spectral_range_start_nm and spectral_range_end_nm, completing counts and required flags in tables.

Kapeldreef 75
3001 Leuven
Belgium
http://www.imec-int.com

Support contact

hsisupport@imec.be
http://hsisupport.imec.be

Contents

1 Introduction

This document describes the content and structure of the sensor calibration files. The structure is the same for all types of sensors.

2 Versioning

Each tag in the calibration file has a version attribute. This facilitates backward compatibility of the calibration files. The version number of a tag is increased when the content of the tag is changed. I.e., addition of removal of children. The version is not increased when the version of one of its children is increased.

3 sensor_calibration

Root of the calibration data.

Version 1

Content for version 1 of sensor_calibration.

Attributes

name	description	required	type
version	The version number for this tag.	yes	integer
sensor_id	The four-digit unique identifier of the sensor	yes	string
timestamp	Date and time on which the sensor has been calibrated. The format of the timestamp is YYYYMMDDThhmmss.	yes	string

Tags

name	description	required	count	version	type
sensor_info	Information about the sensor.	yes	1	1	XML
filter_info	Information about the HSI filters on the sensor	yes	1	1	XML
system_info	System level information that impact the response of the HSI filters on the sensor	yes	1	0	XML

Version 2

Content for version 2 of sensor_calibration.

Attributes

name	description	required	type
version	The version number for this tag.	yes	integer
sensor_id	The four-digit unique identifier of the sensor	yes	string
timestamp	Date and time on which the sensor has been calibrated. The format of the timestamp is YYYYMMDDThhmmss.	yes	string

Tags

name	description	required	count	version	type
sensor_info	Information about the sensor.	yes	1	2	XML
filter_info	Information about the HSI filters on the sensor	yes	1	1	XML
system_info	System level information that impact the response of the HSI filters on the sensor	yes	1	0	XML

4 sensor_info

Contains information about the sensor.

Version 0

Content for version 0 of the sensor_info structure.

Attributes

name	description	required	type
version	The version number for this tag.	yes	integer

Tags

name	description	required	count	type
width	Width of the sensor, in pixels	yes	1	integer
height	Height of the sensor, in pixels	yes	1	integer
adc_gain	Register value of the ADC gain	yes	1	integer
input_bpp	Input number of bits per pixel	yes	1	integer
effective_bpp	Effective number of bits per pixel	yes	1	integer
pixel_pitch	Pixel size, in micrometers	yes	1	float

Version 1

Content for version 1 of the sensor_info structure.

Attributes

name	description	required	type
version	The version number for this tag.	yes	integer

Tags

name	description	required	count	type
width	Width of the sensor, in pixels	yes	1	integer
height	Height of the sensor, in pixels	yes	1	integer
pixel_pitch	Pixel size, in um	yes	1	float
full_well_capacity_e	Measured full well capacity, in electrons	yes	1	integer

Version 2

Content for version 2 of the sensor_info structure.

Attributes

name	description	required	type
version	The version number for this tag.	yes	integer
sensor_type	Allowed values: CMV2K, CDL640	yes	string

Tags

name	description	required	count	type

width_px	Width of the sensor, in pixels	yes	I	integer
height_px	Height of the sensor, in pixels	yes	I	integer
pixel_pitch_um	Pixel size, in um	yes	I	float
bit_depth	The recommended bit depth	yes	I	integer
overall_gain	The corresponding overall gain	yes	I	float
analog_gain	The recommended analog gain	yes (CMV2K)	I	float
digital_gain	The digital gain used during gain calibration	yes (CMV2K)	I	float
full_well_capacity_e	The corresponding full well capacity, in electrons	yes (CMV2K)	I	integer
gain_mode	The gain mode	yes (CDL640)	I	string

Notes:

- The *gain_mode* can have three different values, representing a specific InGaAsMode for the SCD sensor, as shown in the table below:

gain mode	**SCD InGaAsMode**
low	NORMAL_IMAGING_ITR_3000K NORMAL_IMAGING_IWR_3000K
medium	NORMAL_IMAGING_ITR_600KE
high	LNIM_CDS_ITR_12KE

5 filter_info

Contains all information about the HSI filters on the sensor.

Version 1

Content for version 1 of filter_info.

Attributes

name	description	required	type
version	The version number for this tag.	yes	integer

Tags

name	description	required	count	type
calibration_info	Information about the calibration measurements	yes	1	XML
filter_zones	Enumeration of all filter zones on the sensor	yes	1	XML

6 calibration_info

The sensor is calibrated in a monochromator setup. During calibration, the sensor is illuminated with collimated light of specific wavelengths. The tag contains information about the calibration measurements.

Version 3

Content for version 3 of the calibration_info structure.

Attributes

name	description	required	type
version	The version number for this tag.	yes	integer

Tags

name	description	required	count	type
sample_points_nm	Generic vector to depict the exact wavelengths at which the filters are characterized, in nanometres	yes	1	vector<float>
bit_depth	the ADC resolution used during calibration	yes	1	integer
analog_gain	analog (PGA) gain used during sensor calibration	yes	1	float
digital_gain	digital (ADC) gain used during sensor calibration	yes	1	float
measured_overall_gain	overall gain measured during sensor calibration in LSB per electron	yes	1	float

Notes:

1. The conversion gain is computed as

   2^*bit_depth* / *full_well_capacity_e*

2. The overall gain is computed as

   conversion_gain x *analog_gain* x *digital_gain*

3. The measured overall gain might deviate from the computed overall gain since the actual analog gain applied by the PGA on the sensor differs from what is reported in the CMOSIS sensor documentation. The actual analog gain is computed as

   measured_overall_gain / (conversion_gain x *digital_gain*)

Version 4

Content for version 4 of the calibration_info structure. Introduced with the introduction of sensor_calibration version 2 and sensor_info version 2.

Attributes

name	**description**	**required**	**type**
version	The version number for this tag.	yes	integer

Tags

name	**description**	**required**	**count**	**type**
sample_points_nm	Generic vector to depict the exact wavelengths at which the filters are characterized, in nanometers	yes	1	CSV, float

Notes:

1. The attribute *nr_elements* of tag *sample_points_nm* describes the number of elements in the csv list.

7 filter_zone

The tag *filter_zones* is a container of filter zone information for each individual filter zone. The detailed information about one single filter zone of HSI filters on the sensor is contained in an element with tag *filter_zone*.

Version 3

Content for version 3 of the filter_zone structure.

Attributes

name	description	required	type
version	The version number for this tag.	yes	integer
layout	Type of HSI sensor (MOSAIC, TILED, WEDGE)	yes	string
index	0-based index of the filter zone on the sensor	yes	integer

Tags

name	description	required	count	type
filter_area	The area (ROI) on the sensor on which the filters are deposited	yes	1	XML
pattern_width	The number of different filters along the width in the pattern	yes	1	integer
pattern_height	The number of different filters along the height in the pattern	yes	1	integer
filter_width	The size in pixels of each different filter along the width of the pattern	yes	1	integer
filter_height	The size in pixels of each different filter along the height of the pattern	yes	1	integer
spectral_range_start_nm	The lower bound of the spectral range for which the sensor was designed	yes	1	float
spectral_range_end_nm	The upper bound of the spectral range for which the sensor was designed	yes	1	float
bands	Enumeration of all the bands on the sensor and their spectral properties	yes	1	XML

Notes:

1. The width and height of one filter pattern in pixels equals *pattern_width* x *filter_width* and *pattern_height* x *filter_height* respectively.
2. The filter pattern is repeated on the filter area to fill the whole area.

8 filter_area

Defines the position of the area on which filters are deposited on the sensor.

Version 0

Content for version 0 of the filter_area structure.

Attributes

name	description	required	type
version	The version number for this tag.	yes	integer

Tags

name	description	required	count	type
offset_x	Column offset of the filter area from the first column of the sensor	yes	1	integer
offset_y	Row offset of the filter area from the first row of the sensor	yes	1	integer
width	Width of the filter area	yes	1	integer
height	Height of the filter area	yes	1	integer

9 band

The tag *bands* is a container of band information for each individual band. The details about the bands is contained in multiple elements with tag *band*.

Version 3

Content for version 3 of the band structure.

Attributes

name	description	required	type
version	The version number for this tag.	yes	integer
index	0-based index for the position of the band in the pattern, numbered from left to right, top to bottom	yes	integer
selected	Flags if the signal of this band is within specifications (true) or not (false). Usage of non-selected bands will result in wrong spectra	yes	boolean

Tags

name	description	required	count	type
peaks	Enumeration of the peaks and their properties in the band's filter responses	yes	1	XML
response	Vector with the band's response or quantum efficiency per wavelength measured during sensor calibration	yes	1	CSV, float

Notes:

2. The values of the *response* range between 0 (0%) and 1 (100%) and equals the conversion rate of photons to electrons per wavelength. The response vector gives the contribution of each wavelength to the band's signal. The elements of the vector are stored as a csv list. The tag's attribute *nr_elements* describes the amount of elements in the csv list and is always equal to the number of measurement points in *calibration_info*.
3. The number of bands always equals *filter_info.pattern_width* x *filter_info.pattern_height*.
4. There is one band for each position in the pattern.
5. The matrix in which each row is a band's response is also called the sensor's response matrix. Multiplying this matrix with an irradiance spectrum results in the simulated sensor response.

10 peak

The tag *peaks* is a container of peak information for each peak in a band's response composition. The details about the peaks is contained in multiple element with tag *peak*.

Version 2

Content for version 2 of the peak structure.

Attributes

name	description	required	type
version	The version number for this tag.	yes	integer
order	Denotes the harmonic order of the peak	yes	integer
shape	The filter shape that optimally fits the band responses. Possible value: Fabry-Perot, Guassian, Lorentzian	yes	string

Tags

name	description	required	count	type
wavelength_nm	The peak wavelength of the ideal Fabry-Perot filter fitted to the measured band response, in nanometres	yes	1	float
fwhm_nm	The full width of the peak at half the maximum, i.e., full width half maximum, in nanometres	yes	1	float
QE	The quantum efficiency of the peak measured at the peak's central wavelength. The values range between 0 (0%) and 1 (100%)	yes	1	float
contribution	The contribution of the peak's signal to the band's total response. Measured as the area under the fitted peak in [centre ± 1.5xFWHM] relative to the total response	yes	1	float
fit_error	Goodness of fit to an ideal Fabry-Perot filter shape with given peak wavelength, FWHM and peak QE	yes	1	float

11 system_info

Contains all system level information that impact the response of the HSI filters on the sensor.

Version 0

Content for version 0 of the system_info structure.

Attributes

name	description	required	type
version	The version number for this tag.	yes	integer

Tags

name	description	required	count	type
optical_components	Enumeration of the optical components in the system	yes	1	XML
spectral_correction_info	Information for spectral correction of the raw data	yes	1	XML

12 optical_component

The tag *optical_components* is a container of optical component information for multiple individual optical components. The details about each optical component are contained in an element with tag *optical_component*.

The tag *optical_component* contains the full description of the optical component in the measurement system.

Version 1

Content for version 1 of the optical_component structure.

Attributes

name	description	required	type
version	The version number for this tag.	yes	integer

Tags

name	description	req.	#	type
type	Type of the optical component. Possible types are: • *shortpass_filter* • *longpass_filter* • *bandpass_filter* • *notch_filter* • *linear_polarizer* • *circular_polarizer* • *source*	yes	1	string
manufacturer	The manufacturer of the optical component	yes	1	string
part_id	The unique part identifier as given by the manufacturer	yes	1	string
tag	Custom tag uniquely identifying the component	yes	1	string
description	Textual description of the component	yes	1	string
measurement	Source of the response measurement	yes	1	string
transmission_range_start_nm	Start of the component's transmission range in nanometres, defined as the first 50% transmission point of the targeted spectral range or equal to the first value in *sample_points_nm*	yes	1	float
transmission_range_end_nm	End of the component's transmission range in nanometres, defined as the last 50% transmission point of the targeted spectral range or equal to the last value in *sample_points_nm*	yes	1	float

created	The date on which the optical component was created and measured in the format YYYY-MM-DD	yes	1	string
sample_points_nm	Vector to depict the exact wavelengths at which the filters are characterized, in nanometres. The elements of the vector are stored as a csv list	yes	1	CSV, float
response	The measured response (transmission efficiency / emission energy) for each data point	yes	1	CSV, float

Notes:

- The values of *response* range between 0 (0%) and 1 (100%). The elements of the vector are stored as a csv list. The tag's attribute *nr_elements* describes the amount of elements in the csv list and is always equal to the number of sample points in *calibration_info*.

13 spectral_correction_info

Contains information for spectral correction of the raw data. The spectral correction is done by applying a correction matrix to the measured spectra. This correction matrix is computed by analysing the response matrix over a specific spectral range and recomposing virtual filters from the response decomposition data.

The spectral correction matrices are enumerated in the child element *correction_matrices*.

Version 0

Content for version 0 of the *spectral_correction_info* structure.

Attributes

name	description	required	type
version	The version number for this tag.	yes	integer

Tags

name	description	required	count	type
correction_matrices	Enumeration of correction_matrix	yes	1	XML

14 correction_matrix

The tag *correction_matrices* is a container of multiple correction matrices. The details about the correction matrices are contained in multiple elements with tag *correction_matrix*.

Version 4

Content for version 4 of the *correction_matrix* structure.

Attributes

name	description	required	type
version	The version number for this tag.	yes	integer
timestamp	Date and time on which the correction matrix was created. The format of the timestamp is YYYYMMDDThhmmss.xxxxxx	yes	string

Tags

name	description	required	count	type
name	Name of the correction matrix. Any string value is possible.	yes	1	string
algorithm	the algorithm used to create the matrix. Possible algorithms are: • *m0*: correction matrix computed from the estimated system model • *m1*: correction matrix refined with by spectral reference data	yes	1	string
algorithm_version	Version of the algorithm used to generate the correction matrix depicted by two dot-separated numbers. E.g., "1.3". The first number is odd when the correction matrix is computed with ihspt, and even when computed with libs_hsimatlab.	yes	1	string
type	Allowed values: 'rgb', 'hyperspectral' (old name), 'reflectance' (new name), 'radiometric' (old name), 'irradiance' (new name)	yes	1	string
optical_components	Enumeration of all optical components specific for this correction matrix	yes	1	XML
virtual_bands	Enumerates the virtual bands that are computed with the spectral correction	yes	1	XML

Notes:

- The generic, i.e. system level *optical_components* come on top of these listed under *system_info*. This allows defining both a set of fixed optical components (e.g., band pass filters) and a set of application specific optical components (e.g., the light spectrum or a refinement of the bandpass filters)

Version 5

Content for version 5 of the correction_matrix structure.

Attributes

name	description	required	type
version	The version number for this tag.	yes	integer
timestamp	Date and time on which the correction matrix was created. The format of the timestamp is YYYYMMDDThhmmss.xxxxxx	yes	string

Tags

name	description	required	count	type
name	Name of the correction matrix. Any string value is possible.	yes	1	string
algorithm	the algorithm used to create the matrix. Possible algorithms are: • *m0*: correction matrix computed from the estimated system model • *m1*: correction matrix refined with by spectral reference data	yes	1	string
algorithm_version	Version of the algorithm used to generate the correction matrix depicted by two dot-separated numbers. E.g., "1.3". The first number is odd when the correction matrix is computed with ihspt, and even when computed with libs_hsimatlab.	yes	1	string
type	Allowed values: 'rgb', 'reflectance', 'irradiance'	yes	1	string
minimum_band_energy	Minimum energy of the activated/selected measured band responses, i.e. of those bands used for spectral correction	yes	1	float
optical_components	Enumeration of all optical components specific for this correction matrix	yes	1	XML
virtual_bands	Enumerates the virtual bands that are computed with the spectral correction	yes	1	XML

Notes:

- The generic, i.e. system level *optical_components* come on top of these listed under *system_info*. This allows defining both a set of general optical components (e.g., band pass filters) and a set of application specific optical components (e.g., the light spectrum or a refinement of the bandpass filters)

- The *minimum_band_energy* is computed after applying all optical components to the measured band responses. The energy of a band response is computed as the sum of the final response values over all wavelengths (assuming a 1 nm wavelength sampling).

15 virtual_band

The tag *virtual_bands* is a container of band information for each individual virtual band. The details about the virtual bands are contained in multiple elements with tag *virtual_band*.

Version 1

Content for version 1 of the *virtual_band* structure.

Attributes

name	description	required	type
version	The version number for this tag.	yes	integer

Tags

name	description	required	count	type
wavelength_nm	The peak wavelength of the ideal Fabry-Perot filter fitted to the measured band response, in nanometers	yes	1	float
fwhm_nm	The full width of the peak at half the maximum, i.e., full width half maximum, in nanometers	yes	1	float
coefficients	The spectral correction coefficients	yes	1	CSV, float

Notes:

1. Multiplying a measured spectrum with the *coefficients* results in a spectrally corrected value for the virtual band's wavelength. The elements of the vector are stored as a csv list. The tag's attribute *nr_elements* describes the amount of elements in the csv list and is always equal to the total number of bands on the sensor.
2. The matrix in which each row is a virtual band's spectral correction coefficients is also called the sensor's correction matrix. Multiplying this matrix with an irradiance spectrum results in corrected irradiance spectrum.
3. In practice there are at most an equal number of virtual bands as there are actual bands on the sensors.
4. The raw spectrum must be sorted on pattern position index and NOT on peak wavelength when multiplying with the correction matrix.

16 Usage

This section provides more details on how to use the content of the sensor calibration file to interpret and process the data.

Data labelling

The pattern position index of each band links the band's calibration data to the pixels on the sensor. All pixels with the same position in the filter pattern belong to the same band. Their response is given by the band's *response*. The band's *peak* tags summarize the most sensitive regions (i.e., peaks) in the band's response. When composing a hyperspectral image without spectral correction, the band's data can be labelled with the pattern position index or with the wavelength value from the peak tag. When multiple peaks are present, the value is taken from the peak with the largest contribution. It is recommended to label uncorrected data with the peak wavelength only for visualization purposes and keep track of the band index for processing purposes (see 03).

System response

A hyperspectral measurement system typically consists of

1. an HSI sensor;
2. one or more optical filters;
3. a light source.

Each of these components will impact the response of the system. The sensor calibration file intends to describe the impact of all these components to provide an accurate system model. The actual response for a band is extracted from the calibration file by pointwise multiplication of:

1. The band response as measured at sensor level after production with ortho-collimated light in a monochromator setup. This information is found in the *response* field for each *band* on the sensor described in *filter_info*.
2. The response of the optical components common to each scenario or setup in which the system is used. These optical components are typically either built-in to the camera (i.e., cannot change) or mandatory to be used with the sensor to obtain correct data. This information is found in the *response* field of the *optical_components* described under *system_info*.
3. The response of the optical components specific for the setup in which the system is used. These are typically removable optical components and spectra of light sources. This information is found in the *response* field of the *optical_components* of a specific *correction_matrix* described under *system_info*.

Spectral correction

As described in section 02 the sensor's signal is a combination of the sensor level response and the system level components. To measure a correct spectral signature, it is required to calibrate the signal. This is done through spectral correction, which consists of multiplying the signal with a correction matrix. This matrix is specifically computed for the system in which the HSI sensor is used. The corrected signal behaves as if it was captured by a sensor with ideal filters that have a single peak. These are referred to as virtual filters or virtual bands.

The correction matrix is found in the *virtual_bands* tag under *spectral_correction_info. correction_matrices*. The virtual filters are enumerated in the field *virtual_bands*. Each virtual filter has a wavelength, a FWHM and a set of correction coefficients. These correction coefficients are the parameters of a linear function combining the sensor's readings into a corrected value. The number of coefficients always corresponds to the number of bands on the sensor and are ordered by pattern position index. The coefficients are stored in the tag *coefficients*. Each virtual filter's set of correction coefficients corresponds to a row in the correction matrix. Ordering the rows of the correction matrix by virtual wavelength will result in a spectrum in which the samples are ordered by wavelength.

Note: the bands in the uncorrected data must be sorted by band index when multiplying it with the correction matrix.

Note: the default correction matrix must be applied to your signal after removing dark counts and normalizing by a reference measurement.