
Cam. for CRS

Release 1.0.0

Pavel Krsek

Dec 04, 2024

CONTENTS

1	Basler camera class	1
---	---------------------	---

BASLER CAMERA CLASS

The BaslerCamera is the main class which represents interface to the Basler IP cameras. For each camera one object of the class is created and connected with the physical camera based on the IP address or the camera name.

class basler_camera.**BaslerCamera**

Class represents the basledr camera interface

close()

The method close communication with the camera.

config_attrs: list[str]

List of attributes to be save to config file. The attributes are exported/ imported as dictionary by methods `get_as_dict` and `set_from_dict`.

connect_by_ip(ip_addr="")

The method connects the camera by its IP address. The methods lists cameras nearby and selects the camera with appropriate IP address. In case of error the exceptions are raised.

Parameters

ip_addr (*str*) – The desired IP address of the camera.

connect_by_name(name="")

The method connects the camera by its user defined name. The methods lists cameras nearby and selects the camera with appropriate name. In case of error the exceptions are raised.

Parameters

name (*str*) – The desired name of the camera.

exposure_time: float

Camera exposure time [ms] (default value 0.0 mean auto exposure on)

frame_rate: float

Image capture framerate (frames per sec.) (default value 0.0 mean automatic / maximal)

gain: float

Camera gain (default is 0.0, switch off)

gamma: float

Camera gama correction (default is 1.0)

get_as_dict()

The method returns the class parameters as a dictionary. This method is used for writing data into the configuration file. Default implementation should be reimplemented.

Returns

Parameters names and values in the form of a dictionary.

Return type

dict[str, Any]

get_image(*time_out=0*)

The method obtain one image from the camera. The method waits for the image until the image is obtained or timeout is reached.

Parameters

time_out (*int*) – Timeout for the image obtaining [ms]. When the value 0 is used the timeout defined by corresponding attribute of camera object is applied.

Returns

The obtained image is returned. When the image has size 0, means that the image was not obtained within the timeout or other error ocured.

Return type

NDArray[Shape[“row, col, 3”], Any]

grab_image(*time_out=0*)

The method is enccapsulation of the method get_image. This method chcek if the grabing of images is started. If not, the grabbing is started and than the image is obtained. State of grabbing is set back to the state before calling the method.

Parameters

time_out (*int*) – Timeout for the image obtaining [ms]. When the value 0 is used the timeout defined by corresponding attribute of camera object is applied.

Returns

The obtained image is returned. When the image has size 0, means that the image was not obtained within the timeout or other error ocured.

Return type

NDArray[Shape[“row, col, 3”], Any]

grab_timeout: int

Time out for obtaining the image from camera [ms]

ip_address: str

Camera IP address as string

open()

The method opens comunication with connected camera and prepare covnverter for converting the image from camera into the format accepted by OpenCV (cv2).

set_from_dict(*data*)

The method sets the class parameters from the dictionary. This method is used for reading data from the configuration file. Default implementation should be reimplemented.

Parameters

data (*dict[str, Any]*) – Parameters names and values in the form of a dictionary.

Return type

None

set_parameters()

The method sets the camera parameters (in the camera) by values storeed in the corresponding attributes of this class. The attribustes must be set to desired values before setting the parameters in the camera.