Contents

FIMrecorder

FIMrecorder[1] is a special-purpose video-recording application for FIM[2] experiments. This software is at the level of a prototype. It works reliably but has only been tested in a small scope (Supported Devices). The features of this application have been designed to work well in conjunction with FIMTrack[3] [4].

1 Installation

See `README.md`[5].

[1] https://github.com/fncnt/fimrecorder

[2] https://www.uni-muenster.de/PRIA/en/FIM/index.html

[3] https://www.uni-muenster.de/PRIA/en/FIM/download.shtml

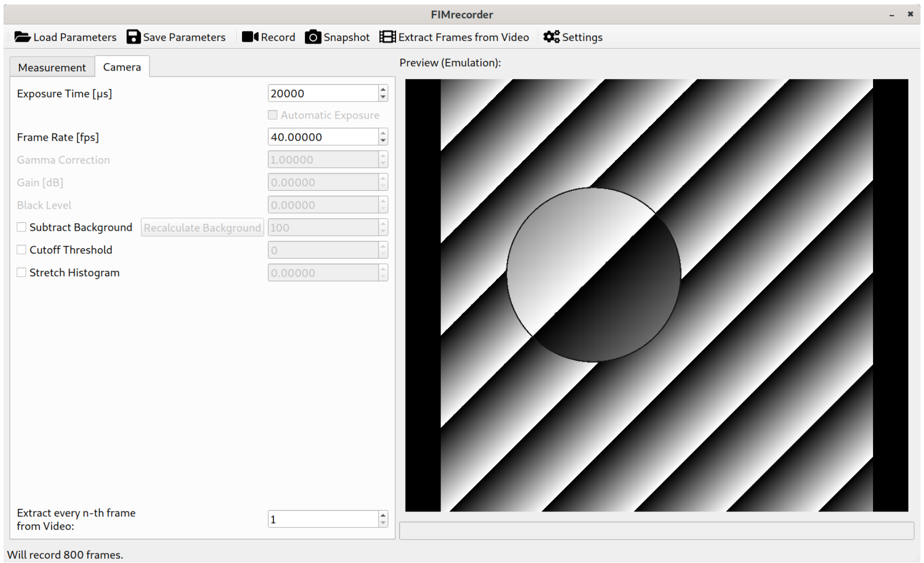
[4] https://github.com/i-git/FIMTrack

[5] https://github.com/fncnt/fimrecorder/blob/master/README.md#prerequisites

2 Supported Devices

Currently, only Basler USB3 vision cameras recording in `Mono8` format are supported and only the model `acA1920-40um` actually has been tested.

3 Overview



Exposure Time [μs] 20000

□ Automatic Exposure

Frame Rate [fps] 40.00000

Gamma Correction 1.00000

Gain [dB] 0.00000

Black Level 0.00000

□ Subtract Background Recalculate Background 100

□ Cutoff Threshold 0

□ Stretch Histogram 0.00000

Extract every n-th frame from Video: 1

Will record 800 frames.

Figure 1: The main UI components of *FIMrecorder*

(1) Loads parameters ((7) & (8)) from a previously saved `.json` file.
(2) Saves parameters ((7) & (8)) to a new `.json` file.
(3) Starts recording for the in (7) specified duration.
(4) Saves a snapshot to `Snapshot Directory`
(5) Extracts frames as single images from a specified video file. (Required for FIMTrack)
(6) settings.
(7) This tab houses information relevant to your experiment such as measurement duration, species or genotype.
(8) Parameters in this tab modify the image signal your camera is acquiring.
(9) model name of the camera you're using.
(10) Live preview of the acquired image signal.
(11) Zooms by scrolling with your mouse or trackpad.
(12) When recording or extracting, the progress is displayed here.
(13) Some relevant status messages appear in this area.

4 Configuration

To adjust settings not visible in the UI, click the button labelled *Settings* (6). This will launch your favourite text editor allowing you to edit the main configuration.

4.1 `settings.json`

`settings.json` is the primary configuration file and can be edited using any text editor. It contains data (`"Parameters"`) relevant to your measurement (see Measurement Annotations for a detailed description). More importantly, it contains a `"Settings"` section controlling the behaviour of the application. The following options can be modified:

`Background Frames to average:` Number of Frames that should be used to construct an averaged static background image for background subtraction.
This setting *is* available from the UI.
`default: 100`

`Configuration Directory:` Path of the directory where additional configurations files should be stored.
`settings.json` is **not** stored here.
`default: "config"`

`Default Camera Parameters:` This is the default `.pfs` file. See [#pfsfiles] for further information
`default: "FIM_NodeMap.pfs"`

`Extract every n-th Frame:` This value determines how many frames are skipped when extracting from video files.
Usually, a value half the framerate of the recorded footage is a good choice, therefore extracting a frame every 0.5 seconds.
This setting *is* available from the UI.
`default: 1`

`Logging Configuration:` The location of your logging configuration. There shouldn't be any need to modify this. See [#loggingconf] for more details.
`default: "loggingconf.json"`

`Single Image Format:` The format snapshots and extracted frames are saved in.
FIMTrack officially supports all three of these, however, `".png"` may be problematic for FIMTrack in some cases.
`supported: ".tif", ".tiff", ".png"`
`default: ".tif"`

`Snapshot Directory:` The location where snapshots are being saved automatically.
`default: "snapshots"`

`Video Codec:` The codec in which recorded video files should be saved.
In principle, every `fourCC`[6] installed on your system *and* supported by `ffmpeg` and `opencv` should work.

[6] `https://fourcc.org/codecs.php`

So far, only `"XVID"` has been tested and determined to be a safe choice.
`default: "XVID"`

`Video Container Format:` The container format of your recorded video footage. Only the default value has been tested. `opencv` has some limitations[7].
`default: ".avi"`

FIMrecorder will fall back to hard-coded defaults and create a new configuration file if you happen to delete it.

4.2 `.pfs` Files

In addition to `settings.json` there are `.pfs` files in your `Configuration Directory` for every camera model you've used in *FIMrecorder*. Those text files are being generated when you use a device for the first time with *FIMrecorder* and include all the parameters of the specific model.

Those files can be used to modify the resolution, offset and binning parameters of your device. If in doubt, take a look at your `Default Camera Parameters` for a comparison. It is recommended to not change any other parameters in these files unless you've read the documentation for your camera model provided by Basler.

5 Basic Workflow

5.1 Pre-Recording

5.1.1 Checking Setup

1. Adjusting field of view.
2. Adjust aperture.
3. Adjust focus. Use the magnifying feature by scrolling on the preview for more control.

[7] `https://docs.opencv.org/4.0.1/d7/d9e/tutorial_video_write.html`

5.1.2 Measurement Annotations

5.1.3 Applying Camera parameters

5.1.4 Real-Time Signal Modifications

5.2 Recording

5.3 Post-Recording

5.3.1 Locating Recorded Data

5.3.2 Extracting Frames from Video Files

6 Troubleshooting

Feel free to open an issue on github[8] if you have any trouble. Please make sure to include a recent debug log.

6.1 Known Limitations

Besides the limited set of supported devices, there are a few more limitations to this date:

- **Only an aspect ratio of 1:1 is displayed correctly in the preview.** This does not affect recording.
- Changing resolution requires editing `.pfs` files
- Only `Mono8` image formats are supported
- Only the default codec and container format are supported

6.2 Logging

FIMrecorder generates `debug.log` files which may help track down possible culprits if you encounter any problems. Please make sure to copy your debug logs as soon as you encounter a problem. Otherwise the important information will be overwritten.

Older debug logs may still be helpful and are suffixed by ascending digits (e.g. `debug.log.1`).

6.2.1 `loggingconf.json`

To change the logging behaviour, `loggingconf.json` in your `Configuration Directory` can be edited. In most cases, you shouldn't be required to do so.

See `logging.config`[9] for further information.

[8] `https://github.com/fncnt/fimrecorder/issues/new`

[9] `https://docs.python.org/3.6/howto/logging-cookbook.html#an-example-dictionary-based-configuration`