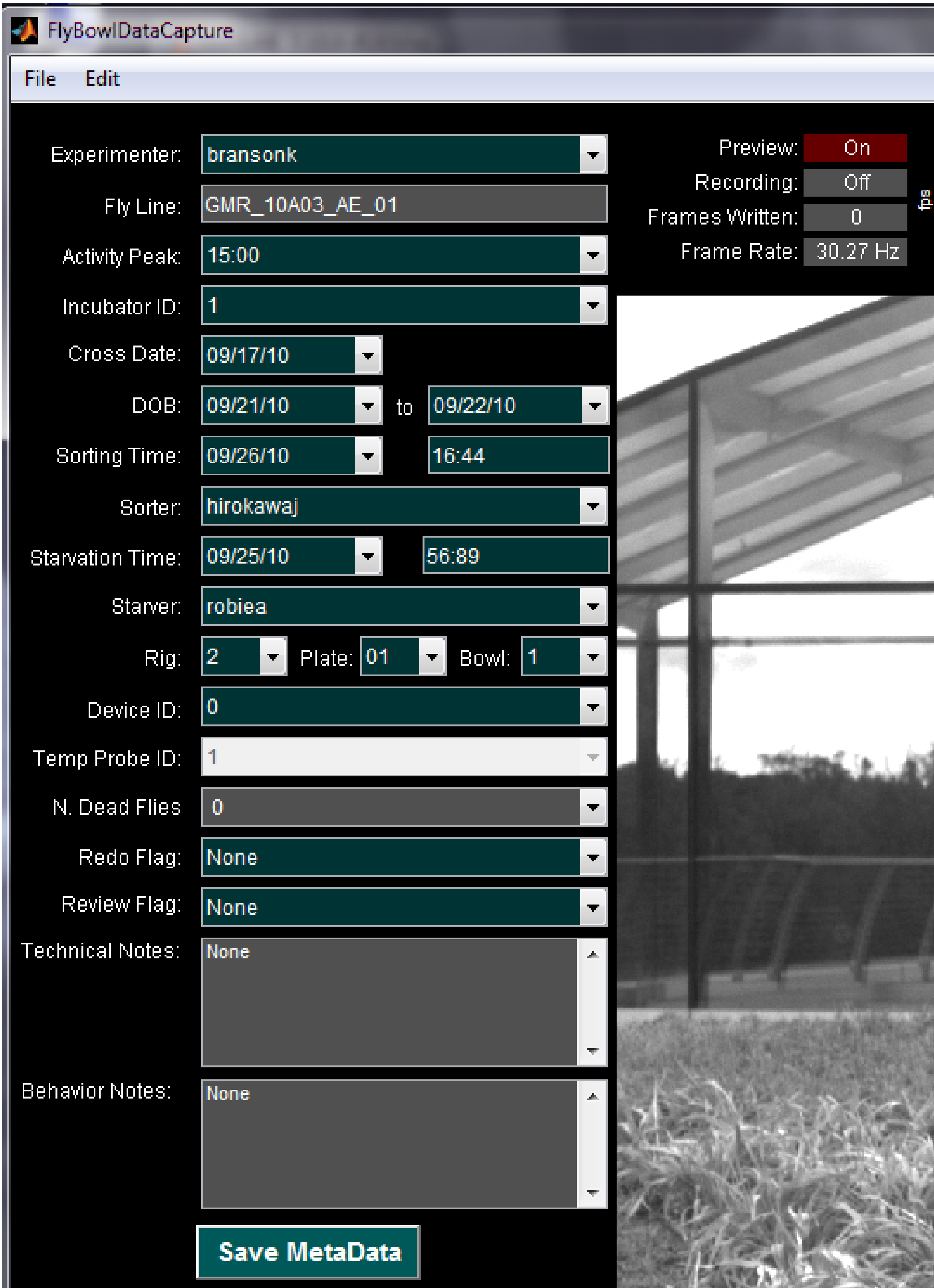
This page last changed on Sep 27, 2010 by bransonk.

FlyBowlDataCapture

Overview

FlyBowlDataCapture is a Matlab program for collecting video and experimental metadata for the FlyBowl Olympiad assay. The user selects which devices to record from, enters metadata about the flies in the left panel of the GUI, clicks to record the timestamps of events, and finally clicks to start recording. A preview of the frames recorded, the frame rate, and temperature stream is shown after the devices are initialized. The program creates a directory for each experiment, and the video, video diagnostics, video log, temperature log, experiment log, and metadata are written to this directory.

Program Usage



Experimenter: bransonk

Fly Line: GMR_10A03_AE_01

Activity Peak: 15:00

Incubator ID: 1

Cross Date: 09/17/10

DOB: 09/21/10 to 09/22/10

Sorting Time: 09/26/10 16:44

Sorter: hirokawaj

Starvation Time: 09/25/10 56:89

Starver: robiea

Rig: 2 Plate: 01 Bowl: 1

Device ID: 0

Temp Probe ID: 1

N. Dead Flies: 0

Redo Flag: None

Review Flag: None

Technical Notes: None

Behavior Notes: None

Preview: On

Recording: Off

Frames Written: 0

Frame Rate: 30.27 Hz

Run Command

To run the program, in Matlab in the directory FlyBowlDataCapture containing all the relevant code, run `FlyBowlDataCapture`

GUI Initialization

The GUI will first prompt the user for the parameters file. Choose the parameters file tailored to your experiment. We are currently using
`FlyBowlDataCaptureParams_bransonlab-ww2_udcam_Basler622f_data1.txt`
See Parameters for information on how to create the parameters file.

The GUI will then pop up. The layout has been optimized to have two GUIs running in a 1080x1920 monitor in portrait mode. Immediately, a log file will be created in the temporary data directory TmpOutputDirectory with the name "TmpLog_<datetime>.txt", for example `TmpLog_20100927T094126.txt.`

A few parameters, such as the OutputDirectory, depend on which instance of the FlyBowlDataCapture program this is. For instance, we will usually be running two instances per computer, and will want to be writing to two separate disks. To allow multiple instance of Matlab and FlyBowlDataCapture to share resources and communicate, there are various semaphore files created. When the GUI is initialized, a semaphore file is created to communicate that a GUI instance is claimed. If this is the first GUI initialized, it will create a file
`.GUIInstances/GUIInstance_1.mat`
or in general
`.GUIInstances/GUIInstance_<instance>.mat`

Metadata Panel

The panel on the left allows the user to input experimental metadata such as the fly line and cross date, see Metadata Entry for a description of each field. The controls will be blue when the control has not yet been used to set the metadata value, and the default value is used. The default value is chosen intelligently, usually using the data entered in the previous experiment. They will turn gray once data has successfully been entered. They will be orange if there is an error in data entry, for instance if the entered starvation time is before the sorting time.

Fly Line Edit Box

The **Fly Line** edit box always begins orange, as the user should enter this piece of data in every experiment. It is an auto-complete edit box: when the user begins typing in the line name, a drop-down menu will appear showing all possible valid line name completions, every line name that begins with the characters before the cursor. If no line names begin with the entered prefix, then all line names will be shown in the drop-down menu. The up and down arrows can be used to navigate the list, and hitting Enter will select the highlighted choice.

Select Camera

In this left panel, the user can also select which devices to record from. The **Device ID** is used to select which camera to record from, the number corresponding to the order that the computer discovers them. Once the camera device is selected, the camera can be initialized by clicking **Initialize Camera**, after which the video preview axes will be updated with a stream from the camera. This preview is of the raw video input from the camera. The same camera device **cannot** be selected in multiple GUIs running simultaneously on the same computer. This is ensured using another semaphore file.

If you plug in cameras after the GUI has started, you can attempt to detect new cameras by selecting the "Edit -> Detect Cameras" menu choice.

When the camera is set, a semaphore file
`.DetectCameras/IsCameraRunning_<AdaptorName>_<DeviceID>.mat`
is created. This file contains the output of `imaqhwinfo` because calling this command in one Matlab will freeze the preview in another Matlab. When one camera is already running, when `imaqhwinfo` needs to be called, we instead load it from this file.

We could not prevent all calls to `imaqhwinfo`. When the camera is set in a second instance of Matlab after it is set in another instance, the first instance's preview will freeze temporarily, but should be restarted automatically.

Select Temperature Probe Channel

The **Temp Probe ID** is used to select which of the allowed channels of the Pico temperature probe to record from. Once the temperature channel is selected, the temperature probe can be initialized by clicking **Init Temp Probe**, after which the recorded temperature will be shown in the top right plot.

Because only one Matlab instance can read from the Pico temperature probe at a time, if multiple Matlabs are running then one Matlab will be the master. It will record the last temperature readings for each channel to a file, and all other Matlabs will read from these files. To indicate that there is a master temperature probe Matlab, a semaphore file
`.TempRecordData/IsMaster.mat`
is created. The temperature for each channel is recorded to the file
`.TempRecordData/Channel_<Channel>`
for example
`.TempRecordData/Channel_01`.

Experiment Events

Three buttons must be pushed in order. First, the time that the flies are brought into the hot room must be recorded by clicking the **Shift Fly Temp** button. Next, the time that the flies are put into the FlyBowl must be recorded by clicking the **Flies Loaded** button. Finally, the **Start Recording** button must be pushed to start recording data. The remaining record time is shown in the "Done" button.

Save Metadata

Before, while, and after video is recorded, metadata in the left panel can be modified. Clicking the "Save MetaData" button will output the current metadata to the metadata file. Metadata is also saved when "Start Recording" is clicked, and when the "Done" button is clicked.

Saving metadata involves first creating a name for the experiment, with the template `<FlyLine>_<Effector>_Rig<Rig>Plate<Plate>Bowl<Bowl>_<StartRecordingTime>`, for example `P0163_TrpA_Rig1Plate01Bowl1_20100925T123228`. If an experiment name is created (Save MetaData pressed) before recording has begun, the name will be `<FlyLine>_<Effector>_Rig<Rig>Plate<Plate>Bowl<Bowl>_notstarted_<GUIInitializationTime>`, for example `P0163_TrpA_Rig1Plate01Bowl1_notstarted_20100925T125156`. The **Experiment Directory** will be created with the experiment name in the OutputDirectory. If this is not the first time metadata has been saved, the old experiment directory will be moved to the new directory name. After the experiment is finished, all relevant data for this experiment should be within in this directory. When metadata is saved, the log file will be immediately placed in the experiment directory and renamed according to the LogFileName parameter (we are currently using "Log.txt"). A metadata XML file will also be saved in the experiment directory with the name set by the MetaDataFileName parameter (we are currently using "Metadata.xml").

Experiment Start

When recording is started, besides files created while saving metadata, two files will be opened and placed in the TmpOutputDirectory to record streamed experiment data. These files are placed in the temporary directory and given temporary names so that the "Save MetaData" button can remain active during recording, and the experiment name can be changed.

Temporary Video File

The program will begin logging all collected video frames to the **temporary video file** in TmpOutputDirectory with the name template:
`FBDC_movie_<StartRecordingTime>_<RandomNumber>.<MovieType>`
where <StartRecordingTime> is the time when the experiment started, <RandomNumber> is a random number associated with the current experiment, and <MovieType> is the extension for the type of video recorded. Example:
`FBDC_movie_20100925T124453_9571.ufmf`
When recording is finished, this file will be placed within the experiment directory, and renamed `movie.<MovieType>`.

Temporary Temperature File

Also when recording is started, the program will begin logging the temperature stream to a **temporary temperature file** in TmpOutputDirectory with the name template:

`FBDC_temperature_<StartRecordingTime>_<RandomNumber>.txt`. The temperature will be recorded every TempProbePeriod seconds. The format of this file will be one line per recording. Each line consists of the timestamp, a comma, then the temperature in degrees Celsius.

Precon Temperature and Humidity Reading

At the start of the experiment, NPreconSamples samples are taken of the temperature and humidity recorded by the Precon sensor. The serial port to be recorded from is set by the PreconSensorSerialPort parameter. The average readings are stored in the metadata file. Again, to prevent multiple Matlabs from requiring the resource simultaneously, a semaphore file
`.PreconRecordData/InUse_<SerialPort>.txt`
exists while samples are being recorded. If this instance finds this file, then it waits and tries to read later. In addition, to avoid many readings from the sensor, we save the average readings in
`.PreconRecordData/PreconTempHumid_<SerialPort>.txt`.
This file is read from instead of the sensor if it is less than 30 seconds old.

Abort Experiment

An experiment can be canceled at any time by clicking the **Abort** button. Data is *not* automatically deleted if an experiment is canceled. The experiment directory will remain. If currently recording, the program will stop recording immediately, and attempt to move the temporary video and temperature files to the experiment. This may take a few seconds, and occasionally may fail to rename because of timing issues between multiple threads. A file named "ABORTED" will be created in the experiment directory.

Status Log

The log in the bottom left corner shows warning messages and status updates. Check here for information about errors caught during the experiment. The same information is written to the experiment log file. The color of the text in the log will be green for the first GUI opened, and blue for the second GUI opened.

Experiment Completion

Once an experiment is finished, click the **Done** button to finalize all entered and recorded data. Metadata will be saved one final time.

Starting a New Experiment

To start a new experiment in the same GUI, use the **File -> New Experiment** menu choice.

The output files from an experiment are described in Output Data.

Semaphores

As discussed above, there are four different resources that must be shared and semaphore files to allow multiple Matlabs to communicate. The semaphore directories are

- .DetectCamerasData
- .TempRecordData
- .GUIInstances
- .PreconRecordData

If the GUI crashes for some reason, some of the semaphore files may not be correctly cleared. You can list all existing semaphores with the command
`ListSemaphores`
and you can delete all semaphores with the command
`ClearSemaphores`.
Clearing these while another GUI is running will potentially cause bad problems.

Parameters

The following parameters must be set in the parameters file input during GUI Initialization. Example values are shown in brackets.

- `Assay_Experimenters`: List of possible experimenters [`hirokawaj,robiea`].
- `NFlies`: Expected number of flies in the bowl [`20`].

- Rearing_IncubatorIDs: List of possible Incubator ID values. [1,2]
- Rearing_ActivityPeaks: List of possible Activity Peak times. These should correspond to the ordered list of Incubator IDs [10:00,15:00].
- Rearing_LightsOns: List of times the lights go on in the incubator. These should correspond to the ordered Incubator IDs list [08:00,13:00].
- Rearing_LightsOffs: List of times the lights go off in the incubator. These should correspond to the ordered Incubator IDs list [15:00,21:00].
- MetaData_RearingProtocols: Names of the rearing protocol files. These should correspond to the ordered Incubator IDs list [RearingProtocol0001_Morning,RearingProtocol0002_Afternoon].
- PreAssayHandling_CrossDate_Range: Range of Cross Dates to show – minimum and maximum number of days before today [4,10].
- PreAssayHandling_DOBStart_Range: Range of DOB Starts to show – minimum and maximum number of days before today [1,7].
- PreAssayHandling_DOBEnd_Range: Range of DOB Ends to show – minimum and maximum number of days before today [0,3].
- PreAssayHandling_SortingDate_Range: Range of Sorting Dates to show – minimum and maximum number of days before today [0,2].
- PreAssayHandling_SortingHandlers: List of possible Sorting Handler values [hirokawaj,robiea].

1. Range of Starvation Dates to show – minimum and maximum number of days before today [0,2]. PreAssayHandling_StarvationDate_Range

- PreAssayHandling_StarvationHandlers: List of possible Starvation Handler values [hirokawaj,robiea].
- Assay_Rigs: List of possible Rig names [1,2].
- Assay_Plates: List of possible Plate names [01,02].
- Assay_Bowls: List of possible Bowl names [1,2,3,4].
- RedoFlags: List of possible Redo flags [None,Rearing problem,Flies look sick,See behavioral notes,See technical notes].
- ReviewFlags: List of possible Review flags [None,Rearing problem,Flies look sick,See behavioral notes,See technical notes].
- Imaq_Adaptor: Camera adaptor name [dcam].
- Imaq_DeviceName: Camera name [A622f].
- Imaq_VideoFormat: Video format. For gdcam, udcam, this should be Format 7, Mode 0. For dcam, it should be F7_Y8_1280x1024.
- Imaq_ROIPosition: Video ROI Position: xmin, ymin, width, height [0,0,1024,1024].
- Imaq_FrameRate: Frame rate we expect [30.4].
- Imaq_MaxFrameRate: Maximum frame rate we expect to get [31].
- Imaq_Shutter: Shutter period [100].
- Imaq_Gain: Gain [200].
- FileType: Extension of video file, such as fmf, ufmf, avi [ufmf].
- RecordTime: Number of seconds of video to record [1000].
- PreviewUpdatePeriod: Minimum time in between preview window updates [0].
- gdcamPreviewFrameInterval: For gdcam, number of frames between updates [2].
- OutputDirectory: List of directories to put all the experiment directories within. Which element of this list is selected based on which instance of the GUI this is [C:\Users\labadmin\Documents\FlyBowl\data1,D:\data2].
- HardDriveName: List of hard drive names, corresponding to the ordered OutputDirectory list [Internal_C,HD3].
- MovieFilePrefix: Prefix of movie file to store [movie].
- TmpOutputDirectory: List of directories to store temporary data within. Which element of this list is selected based on which instance of the GUI this is. These should be on the same disk as the corresponding OutputDirectory to avoid having to rewrite data when renaming from temporary to permanent file names [C:\Users\labadmin\Documents\FlyBowl\tmpdata1,D:\tmpdata2].
- MetaDataFileName: Name of file to store metadata to [Metadata.xml].
- LogFileName: Name of file to store log to [Log.txt].
- MetaData_AssayName: Assay name [FlyBowl].
- MetaData_SortingHandlingProtocols: Name of sorting protocol file [SortingProtocol0001].
- MetaData_StarvationHandlingProtocols: Name of starvation protocol file [StarvationProtocol0001].
- MetaData_ExpProtocols: Name of experiment protocol file [ExperimentProtocol0001].
- MetaData_Effector: Name of effector [TrpA].
- MetaData_RoomTemperatureSetPoint: Set point temperature for the experiment [29].
- MetaData_RoomHumiditySetPoint: Set point humidity for the experiment [60].

- FrameRatePlotYLim: Limits of y-axis for the frame rate plot [0,35].
- TempPlotYLim: Limits of y-axis for temperature plot [15,45].
- DoQuerySage: Whether to try to connect over the network to Sage to query line names, 1 to query, 0 not to query [1].
- TempProbePeriod: Time in between temperature readings from the Pico temperature probe [1].
- TempProbeChannels: Which channels of the Pico probe can be recorded from [1,2,3].
- TempProbeTypes: Thermocouple types for each channel of the Pico probe. This should correspond to the ordered TempProbeChannels list [K,K,K].
- TempProbeReject60Hz: If 1, Pico temperature probe will reject 60 Hz, otherwise will reject 50 Hz [0].
- DoRecordTemp: Whether to record a temperature stream from the Pico probe. 1 means record, 0 means do not. [1].
- NPreconSamples: Number of temperature and humidity samples to read from the Precon probe to compute average start temperature and humidity. Set to 0 if no recording should be taken [5]. PreconSensorSerialPort: Name of serial port to connect to [COM3].
- ExtraLineNames: Line Names to add to the list read from Sage [DL-wildtype].
- UFMFLogFileName: Log file for writing ufmf [ufmf_log.txt].
- UFMFStatFileName: Name of file to write UFMF diagnostic stats to {ufmf_diagnostics.txt}}].
- UFMFPrintStats: Whether to compute UFMF diagnostic statistics. 1 means compute, 0 means don't compute [1].
- UFMFStatStreamPrintFreq Number of frames between outputting per-frame compression statistics: 0 means don't print, 1 means every frame [30].
- UFMFStatComputeFrameErrorFreq: Number of frames between computing statistics of compression error. 0 means don't compute, 1 means every frame [30].
- UFMFStatPrintTimings: Whether to print information about the time each part of the computation takes. 1 means do print, 0 means don't print [1].
- UFMFMaxFracFgCompress: Maximum fraction of pixels that can be foreground to try compressing frame. Otherwise, the raw frame will be output [.2].
- UFMFMaxBGNFrames: Approximate number of frames the background model should be based on. This means the background model is based on a window of size UFMFMaxBGNFrames * UFMFBGUpdatePeriod. This is actually not used in the current median implementation [100].
- UFMFBGUpdatePeriod: Number of seconds between updates to the background model [1].
- UFMFBGKeyFramePeriod: Number of seconds between spitting out a new background model [100].
- UFMFMaxBoxLength: Maximum box width and height stored during compression [5].
- UFMFBackSubThresh: Threshold for background subtraction [15].
- UFMFNFramesInit: First nFramesInit we always update the background model [100].
- UFMFBGKeyFramePeriodInit: While ramping up the background model, use this list of keyframe periods. There can be at most 5 numbers in this list [1,10,25,50,75].

Metadata Entry

The following metadata must be entered using the left panel in the GUI:

- **Experimenter**: LDAP name of person conducting the experiment. Possible values are set by Assay_Experimenters parameter.
- **Fly Line**: Name of fly line registered in Sage.
- **Activity Peak**: 24-hour time of peak activity for the flies. The possible values for this are set by the Rearing_ActivityPeaks parameter. One activity peak should be used in the morning, the other in the afternoon. [We should maybe remove this or Incubator ID, as this will always correspond to incubator ID?].
- **Incubator ID**: ID of the incubator the flies were raised in. Possible values for this are set by the Rearing_IncubatorIDs parameter.
- **Cross Date**: The date the cross was performed on to generate these flies. Possible values are set by the range of dates set in the PreAssayHandling_CrossDate_Range parameter.
- **DOB**: Range of dates of birth for the flies. Possible values are set by PreAssayHandling_DOBStart_Range and PreAssayHandling_DOBEnd_Range. [Is there more information for this available than just the Cross Date?]
- **Sorting Time**: Date and time that the files were sorted on. The range of possible dates is set by PreAssayHandling_SortingDate_Range.
- **Sorter**: LDAP name of person who sorted the files. Possible values are set by the PreAssayHandling_SortingHandlers parameter.
- **Starvation Time**: Date and time that the flies were moved to starvation material. The range of possible dates is set by PreAssayHandling_StarvationDate_Range.

- **Starver**: LDAP name of person who moved the files to starvation material. Possible values are set by the PreAssayHandling_StarvationHandlers parameter.
- **Rig**: ID of the rig (which cart) the current bowl is on. Possible values are set by the Assay_Rigs parameter.
- **Plate**: ID of the plate the current bowl is in. Possible values are set by the Assay_Plates parameter.
- **Bowl**: ID of current bowl. Possible values are set by the Assay_Bowls parameter.
- **Device ID**: Which camera device we should record from. Possible values are autodetected using imaqhwinfo.
- **Temp Probe ID**: Channel of the Pico temperature probe to record the temperature stream from. Possible values are set by the TempProbeChannels parameter.
- **N. Dead Flies**: Number of flies that are dead in the bowl.
- **Redo Flag**: Reason the experiment should be redone.
- **Review Flag**: Reason the experiment should be reviewed.
- **Technical Notes**: Notes from the experimenter about possible technical problems during the experiment.
- **Behavior Notes**: Notes from the experimenter about possible behavior problems or observations during the experiment.

Output Data

All data is output to the Experiment Directory within the OutputDirectory. The following files will be in this directory:

- Log File
- Metadata File
- UFMF Video
- UFMF Diagnostics File
- UFMF Log File

Log File

The log file Log.txt contains a record of all events during the experiment, as well as any errors or warnings. It is the same as the information output to the status window. Here is an example log file:

```
FlyBowlDataCapture v. 0.1
--------------------------------------
14:51:56: GUI instance 1, writing to
C:\Users\labadmin\Documents\FlyBowl\data1, random number
9594.
14:51:56: SAGE code directory
../SAGE/MATLABInterface/Trunk could not be added to the
path.
14:51:57: Read line names from file.
14:51:57: GUI initialization finished.
14:51:59: DeviceName =
Adaptor_udcam___Name_A622f___Format_Format 7, Mode
0___DeviceID_0___UniqueID_0053300063A94001
14:51:59: Video preview started.
14:52:05: Shifted fly temperature.
14:52:06: Flies loaded.
14:52:07: Experiment name initialized to
pdf-2-GAL4_TrpA_Rig1Plate01Bowl1_20100927T145207
14:52:07: Creating experiment directory
C:\Users\labadmin\Documents\FlyBowl\data1\pdf-2-GAL4_TrpA
_Rig1Plate01Bowl1_20100927T145207

14:52:07: Saved MetaData to file
C:\Users\labadmin\Documents\FlyBowl\data1\pdf-2-GAL4_TrpA
_Rig1Plate01Bowl1_20100927T145207\Metadata.xml.
14:52:07: Started recording to file
C:\Users\labadmin\Documents\FlyBowl\tmpdata1\FBDC_movie_2
0100927T145207_9594.ufmf.
14:54:45: Renaming movie file from
C:\Users\labadmin\Documents\FlyBowl\tmpdata1\FBDC_movie_2
```

```
0100927T145207_9594.ufmf to
C:\Users\labadmin\Documents\FlyBowl\data1\pdf-2-GAL4_TrpA
_Rig1Plate01Bowl1_20100927T145207\movie.ufmf
14:54:45: Finished recording. Video file moved from
C:\Users\labadmin\Documents\FlyBowl\tmpdata1\FBDC_movie_2
0100927T145207_9594.ufmf to
C:\Users\labadmin\Documents\FlyBowl\data1\pdf-2-GAL4_TrpA
_Rig1Plate01Bowl1_20100927T145207\movie.ufmf.
```

Metadata File

The metadata file `Metadata.xml` is an XML file containing all the metadata about the current experiment. It contains the following information:

- `experiment`
    - `assay`: Name of assay, from `MetaData_AssayName` parameter.
    - `protocol`: Name of protocol file, from `MetaData_ExpProtocols` parameter.
    - `datetime`: Date and time recording was started.
    - `aborted`: Whether the experiment was aborted early (1) or not (0).
    - `experimenter`: LDAP name of person conducting the experiment, set by Experimenter control.
    - `shiftflytemp_time`: Seconds between bringing the flies into the hot room and starting recording. Set by clicking the Shift Fly Temp button.
    - `fliesloaded_time`: Seconds between loading the flies into the bowl and starting recording. Set by clicking the Flies Loaded button.
    - `apparatus`
        - `rig_id`: Name of rig, set by Rig control.
        - `plate_id`: Name of plate, set by the Plate control.
        - `bowl_id`: Name of the bowl, set by Bowl control.
        - `camera`
            - `adaptor`: Name of the imaq adaptor, set by `Imaq_Adaptor` parameter.
            - `device_name`: Name of the device, set by `Device_Name` parameter.
            - `format`: DCAM format code, set by `Imaq_VideoFormat`.
            - `device_id`: ID of the camera, as detected by `imaqhwinfo`. Set by `Device ID` control.
            - `unique_id`: Unique ID for this camera, queried from device.
        - `computer`
            - `id`: ID of the computer, queried using `system('hostname'}`. T
            - `harddrive_id`: ID of the harddrive the data was stored on, set by GUI instance and `HardDriveName` parameter.
            - `output_directory`: Path to output directory, set by GUI instance and `OutputDirectory` parameter.
        - `flies`
            - `line`: Line name, set by Fly Line control.
            - `effector`: Name of effector, set by `MetaData_Effector` parameter.
            - `crossdate`: Date the cross for these flies was performed, set by the Cross Date control.
            - `age`: Range of flies' ages in days, computed from DOB range control.
            - `count`: Number of flies. This is currently set to 0 as we have not yet counted the number of flies.
            - `rearing_protocol`
                - `id`: Name of rearing protocol file, set by activity peak and `MetaData_RearingProtocols` parameter.
                - `incubator`: ID of the incubator the flies were raised in, set by Incubator ID control.
            - `handling_protocol`, `type = "sorting"`
                - `handler`: LDAP of person who sorted the flies, set by Sorter control.
                - `time`: Hours between recording start and sorting time, computed from Sorting Date and Time controls.
            - `handling_protocol`, `type = "starvation"`
                - `handler`: LDAP of person who moved the flies to starvation material, set by Starver control.
                - `time`: Hours the flies have been starved, computed from Starvation Date and Time controls.
        - `environment`

        - `temperature`: Temperature recorded from the Precon sensor at the start of the experiment, in degrees Celsius.
        - `humidity` Humidity recorded from the Precon sensor at the start of the experiment, in percent.
    - `note`, `type = "behavioral"`: Observations about the flies' behavior, entered by experimenter during the experiment.
    - `note`, `type = "technical"`: Observations about technical aspects of the experiment, entered by the experimenter during the experiment.
    - `flag`, `type = "review"`: If this flag is set, then the experimenter's opinion is that this experiment should be reviewed, otherwise the flag will be absent.
        - `reason`: Why the experiment should be reviewed, set by the Review Flag control.
    - `flag`, `type = "redo"`: If this flag is set, then the experimenter's opinion is that this experiment should be redone, otherwise the flag will be absent.
        - `reason`: Why the experiment should be redone, set by the Redo Flag control.

Here is an example metadata file:

```
<?xml version="1.0"?>
<experiment assay="FlyBowl" protocol="ExperimentProtocol0001" datetime="2010-09-27T15:34:59"
 aborted="1" experimenter="bransonk" shiftflytemp_time="4.225002" fliesloaded_time="3.640998" >
  <apparatus rig_id="2" plate_id="01" bowl_id="1">
    <camera adaptor="udcam" device_name="A622f" format="Format 7, Mode 0" device_id="0"
 unique_id="0053300063A94001" />
    <computer id="bransonlab-ww2" harddrive_id="Internal_C" output_directory="C:\Users\labadmin
\Documents\FlyBowl\data1"/>
    <flies line="GMR_34A04_AE_01" effector="TrpA" crossdate="2010-09-17" age="4.000000,7.000000"
 count="0">
      <rearing_protocol id="RearingProtocol0002_Afternoon" incubator="1" />
      <handling_protocol id="SortingProtocol0001" type="sorting" handler="hirokawaj"
 time="22.849824" />
      <handling_protocol id="StarvationProtocol0001" type="starvation" handler="robiea"
 time="54.099824" />
    </flies>
    <environment temperature="NaN" humidity="60.000000" />
    <note type="behavioral">Here is a note. </note>
    <note type="technical">None</note>
    <flag type="redo" reason="FLIES LOOK SICK"/>
  </apparatus>
</experiment>
```

UFMF Video

The primary piece of data output is the video record. The format is described at UFMF File Description. See bransonlab:udcam for information on how udcam creates this video. The video can be previewed with the `playfmf` or `showufmf` functions in JCtrax.

UFMF Diagnostics File

udcam outputs diagnostics about the video compression performed to the file `ufmf_diagnostics.txt`. See bransonlab:udcam for information about these diagnostics.

UFMF Log File

udcam outputs status, warning, and error messages to the log file `ufmf_log.txt`. See bransonlab:udcam for information about this file.