BEHAVE YOURSELF!

Animal Tracking: a DIY reference

Mazord

June 28, 2024

Contents

Everything should be made as simple as possible but not any simpler
—Albert Einstein

1 Open Source is the key

Also freely available is desirable... but you can't really always expect that much!

2 General concepts

2.1 The problem

- If you are reading this, then you are likely interested in video-tracking of animals for behavioural studies.
- This guide is an attempt to provide indications on how to approach this problem starting from the basics. As the title implies, the idea is to provide suggestions on how to implement everything that is required on a Do It Yourself basis, both from the hardware as well as from the software points of vew.
- Although this may seem a difficult undertaking, in reality it is relatively simple. All that is required is some programming experience as well as a basic knowledge of electronics.
- I will deal with the subject matter mainly referring to behavioural experimentation in the lab involving *Drosophila melanogaster* as the model organism. However, most of what will be exposed can be adapted to other laboratory model organisms with relative ease.

2.2 The solutions

- At the most basic level what will be needed is:
  1. a transparent enclosure (arena) into which to place the animal(s) to be tracked;
  2. a digital video camera with which to capture the movements of the subjects;
  3. a computer to which the camera will be attached and which will run the software being used to capture the video footage and/or perform the real time tracking.
- Some further considerations which could be made are related, for example, to the type of lighting to use during the video recording/tracking. Typically you would need to adequately illuminate the scene using some form of "normal" visible white light. While this may be fine for just tracking

animals in a visually unchanging environment, this may not be a good solution when the video recording/tracking is performed while providing the subject with visual stimuli (i.e. the visual environment surrounding the subject is not static). In this case, the visual stimuli will also be "seen" by the camera and by the tracking software and this will cause interferences with the tracking process. One solution is to illuminate the scene with infra-red light and to equip the video camera with a long pass Infra Red (IR) filter. This guarantees that the video camera will only be able to "see" the animals (illuminated by the IR light) and not the visual stimuli (which should be restricted to the visibile part of the light spectrum).

2.2.1 Further considerations on lighting

Since correct lighting is a key issue in this context, it is worth considering the possibility of housing the arena, camera and the UST projector (if one is being used to provide visual stimuli to the experimental animals) in a light-proof enclosure. There are many ways in which this can be constructed, with materials ranging from plywood to aluminium and black acrylic (i.e. the latter, as described here: MARGO).

3 The logic of the basic experimental setups

There are two basic scenarios which normally present themselves:

3.1 Tracking of freely moving animals in the absence of specific visual stimulation

In this context, there are different possibilities:

- 1. Single animal in a single arena
- 2. Multiple animals in a single arena
- 3. Multiple animals, each in a single arena

Given that tracking of the animals occurs in the absence of visual stimulation, it follows that illumination of the arenas need not be done with infrared light. Consequently video recording of the behaviour of the animal(s) can be performed with ordinary optics (i.e. no special light filters are required). Illumination is best done from beneath the arena, while the video camera records the backlit scene from above. In the first case, the software of choice for tracking could be MARGO, if one is interested in live-tracking of the animal. If on the other hand, live-tracking is not important, then Ctrax can be used for offline tracking.

In the case of multiple animals, if one is interested in maintaining the identity of the individuals distinct throughout the tracking process, then Ctrax would be the tracking software of choice, albeit that the tracking can only be performed offline. If, on the other hand, it is not important to keep the identity of the animals separate during the tracking, i.e. in the case one is only interested in knowing the total amount of movement (the total or the average distance travelled collectively by all the animals) during the tracking period, then the MARGO package can be used to perform live-tracking of the multiple animals.

3.2 Tracking of freely moving animals in the presence of specific visual stimulation

In this context it is more customary to track animals under one of the following two conditions:

- 1. Single animal in a single arena
- 2. Multiple animals, each in a single arena

In this case, the reason why the condition with "multiple animals in a single arena" is not considered is that in this type of setup the individuals are under the influence of one another, since they share the same arena. This would introduce a source of (possibly unwanted) disturbance, in particlar if the objective of the setup is to evaluate the response of the animals specifically to the visual stimuli.

In both cases, the software of choice for tracking is MARGO, especially if one is interested in live-tracking of the individuals. Ctrax could also be used in these cases, albeit the tracking can only be performed offline.

Within the context of tracking animals subjected to specific visual stimulation, we can consider two different experimental paradigms. Open-loop or closed-loop visual stimulation.

3.2.1 Open-loop conditions

Under open-loop conditions, the animals are subjected to visual stimuli, which can be provided by computer-controlled Light Emitting Diode (LED) panels or by Ultra Short Throw (UST) projector(s). In open-loop conditions, the tracked behavioural response of the animals has no effect on the visual stimuli.

3.2.2 Closed-loop conditions

Under closed-loop conditions the animals are subjected to visual stimuli, which can be provided by computer-controlled LED panels or by UST projector(s). In this case the tracked behavioural response of the animals can affect the visual stimuli: The way in which the behavioural response of the animals affects the visual stimuli is determined by programming the software which controls the visual stimuli to respond in predetermined ways to specific behavioural parameters. The behavioural parameters are obtained through live-tracking of the animal's movements.

4 What you'll need

4.1 Hardware

4.1.1 Electronics

1. Arduino

   Arduino electronic boards are open-source hardware platforms designed for creating digital devices and interactive objects that can sense and control physical elements. They are popular for their ease of use, versatility, and wide range of applications in electronics projects. Here are some key characteristics:

   (a) **Microcontroller-Based**: Each Arduino board comes with a microcontroller (e.g., ATmega328 on the Arduino Uno), which acts as the brain of the board, executing programmed instructions.

   (b) **Digital and Analog I/O Pins**: Arduino boards have multiple input and output pins that can be used to connect sensors, LEDs, motors, and other electronic components.

   (c) **Programming Environment**: Arduino uses a simplified version of C/C++ and comes with an integrated development environment (IDE) that is user-friendly and easy to learn.

   (d) **USB Connectivity**: Most boards can be connected to a computer via USB for programming and power supply.

   (e) **Wide Compatibility**: Arduino is compatible with a vast array of sensors, modules, and shields that extend its functionality.

   Some typical applications for Arduino boards:

   - **Prototyping**: Rapid development of electronic prototypes.
   - **Education**: Teaching electronics and programming.
   - **Robotics**: Controlling robots and automation systems.
   - **IoT**: Creating Internet of Things devices that can communicate over networks.
   - **DIY Projects**: Building personal projects, such as home automation systems, wearable technology, and interactive art.

Other important points of strength of the Arduino project are:

- **Open Source**: The hardware and software are open-source, allowing for customization and community contributions.
- **Extensive Documentation**: A wealth of tutorials, forums, and examples are available online.
- **Libraries**: Numerous libraries are available to simplify the use of various sensors and modules.

Arduino boards offer a powerful and accessible platform for both beginners and experienced electronics enthusiasts to bring their ideas to life.

This is the link to the Arduino development board, software download and documentation pages: Arduino

2. LED Panels

LED-based panels can efficiently be employed to present visual stimuli to model organisms. In 2008, Michael Reiser and Michael Dickinson (Reiser and Dickinson, 2008) reported the design of a cylindrical modular display system for insect behavioral neuroscience. Following that report, the design was further updated, and all the technical details for the construction of such modular systems were made publicly available through the author's Github website Modular-LED-Display. The website documents everything required to implement a fully functional system, including commercial sources for the LED panels, the electronics, the hardware and the software required to control them. As of writing, the LED panels used are monochromatic (i.e. green LEDs), however this also results in very high speed refresh rates (up to 500Hz, with 256 brightness levels).

3. Ultra-Short-Throw Digital Projectors

UST projectors refer to "Ultra Short Throw" projectors. These are specialized projectors designed to project large images from a very short distance to the screen or wall. In general, UST projectors can project large images(e.g., 250cm, diagonal, or more) from just a few centimeters away from the screen or the surface on which to project the image. This is achieved through advanced lens and mirror systems. Nowdays, such projectors are widely available on the consumer market at relatively affordable prices, depending on the required quality of the projected image (typically resolution, brightness and contrast). However, the applications envisaged in the specific context being considered here, require the projection of high definition, high contrast images, having sizes in the range of 25cm (diagonal). The type of projectors that have such characteristics are those designed for applications requiring the short distance projection of relatively small images which can function as user interfaces (i.e. touch screen functions on common surfaces, such as walls or panels of any material, including glass).

Currently we are using projectors from Texas Instruments (Texas Instruments, DLP LightCrafter™ Display 4710 EVM Gen2).

4. Digital Cameras

   When selecting video cameras for video tracking applications, certain characteristics are particularly desirable to ensure accurate and efficient tracking performance. Here are some key attributes to consider:

   ### 1. **High Frame Rate**:

   - **Importance**: High frame rates (60 fps or higher) are essential for capturing smooth motion, which is crucial for accurately tracking fast-moving objects.
   - **Benefit**: Reduces motion blur and provides more frames for the tracking algorithm to work with, leading to more precise tracking.

   ### 2. **High Resolution**:

   - **Importance**: High resolution allows for more detailed images, which can improve the accuracy of the tracking system.
   - **Benefit**: Enables the detection of small or distant objects and provides more data points for the tracking algorithm.

   ### 3. **Low Latency**:

   - **Importance**: Low latency ensures that the video feed is processed in real-time, which is critical for responsive tracking.
   - **Benefit**: Minimizes delays between the actual movement and the tracked data, which is important in applications like robotics, sports analysis, and surveillance.

   ### 4. **Global Shutter**:

   - **Importance**: Global shutters capture the entire frame at once, as opposed to rolling shutters which capture the frame line by line.
   - **Benefit**: Eliminates motion artifacts and distortions, making it easier to track fast-moving objects accurately.

   ### 5. **High Dynamic Range (HDR)**:

   - **Importance**: HDR capability allows the camera to capture details in both very bright and very dark areas of the scene.
   - **Benefit**: Improves tracking performance in varying lighting conditions, ensuring that objects are clearly visible even in challenging environments.

   ### 6. **Good Low-Light Performance**:

- **Importance**: Many tracking applications may occur in low-light or variable lighting conditions.
- **Benefit**: Cameras with good low-light sensitivity can capture clear images in dim lighting, ensuring reliable tracking.

7. **Robust Connectivity**:

- **Importance**: Fast and reliable data transfer is essential for high-performance tracking.
- **Benefit**: Interfaces like USB 3.0, GigE, and PCIe offer high bandwidth and low latency, which are beneficial for transmitting high-resolution video at high frame rates.

8. **Compact and Lightweight Design**:

- **Importance**: Depending on the application, such as drones or robotics, the size and weight of the camera can be critical.
- **Benefit**: Easier to integrate into systems where space and weight are limited.

9. **Software and SDK Support**:

- **Importance**: Comprehensive software support ensures that the camera can be easily integrated into existing systems.
- **Benefit**: Access to robust SDKs and APIs facilitates the development of custom tracking algorithms and applications.

10. **Durability and Reliability**:

- **Importance**: Cameras used in industrial or outdoor environments need to be durable and reliable.
- **Benefit**: Withstands harsh conditions and continues to perform accurately over long periods.

11. **Multi-Camera Synchronization**:

- **Importance**: For applications requiring multiple cameras, such as 3D tracking or wide-area surveillance.
- **Benefit**: Synchronizes multiple video feeds for coherent and accurate multi-angle tracking.

Applications of Video Tracking Cameras:

- **Surveillance**: Monitoring and tracking individuals or objects in security applications.
- **Sports Analytics**: Tracking players and objects to gather performance data.

- **Robotics**: Enabling robots to track and interact with their environment.
- **Healthcare**: Tracking patient movements for diagnostics and rehabilitation.
- **Automotive**: Tracking vehicles and objects for autonomous driving systems.

Example Cameras for Video Tracking:

- **FLIR Blackfly S**: Known for high frame rates, global shutter, and high resolution.
- **Basler ace**: Offers a range of resolutions and frame rates, and is compact and lightweight.
- **XIMEA xiQ Series**: Compact design with high frame rates and low latency, ideal for embedded systems.

Selecting the right video camera for tracking depends on the specific requirements of your application, such as the speed of the objects being tracked, lighting conditions, and integration needs.

Currently in our labs we use two types of digital cameras:

- Chameleon3 USB3 digital cameras by FLIR Systems (formerly Point Grey Research). This is the link to FLIR.
- Ximea digital cameras. This is the link to XIMEA

The Chameleon3 USB3 digital cameras, are a line of high-performance, compact digital cameras designed for a variety of applications including industrial inspection, scientific imaging, and machine vision. Here are some key features and details about the Chameleon3 USB3 digital cameras:

(a) **USB 3.0 Interface**: These cameras utilize the USB 3.0 interface, providing high data transfer rates of up to 5 Gbps, which allows for faster image capture and transfer.

(b) **High-Quality Sensors**: Chameleon3 cameras are equipped with high-quality CMOS and CCD sensors from leading manufacturers like Sony and On Semiconductor, offering resolutions ranging from VGA to several megapixels.

(c) **Compact and Lightweight Design**: The cameras are designed to be compact and lightweight, making them suitable for integration into various systems where space is limited.

(d) **Flexible Configuration**: They come with flexible configuration options including adjustable frame rates, exposure times, and gain settings, allowing customization to suit specific application needs.

(e) **Image Quality**: These cameras provide high image quality with features like low noise, high dynamic range, and excellent sensitivity, making them suitable for capturing detailed and accurate images in various lighting conditions.

(f) **Software and SDK Support**: The Chameleon3 cameras are supported by FLIR's FlyCapture software and SDK, which provide comprehensive tools for camera control, image capture, and image processing. This makes it easier for developers to integrate the cameras into their applications.

(g) **Versatility**: These cameras are used in a wide range of applications including machine vision, robotics, medical imaging, microscopy, and scientific research due to their reliable performance and versatility.

(h) **Compatibility**: The cameras are compatible with a variety of operating systems including Windows, Linux, and macOS, ensuring broad usability across different platforms.

The XIMEA MQ013RG-ON is a model from XIMEA's xiQ series of USB3 Vision cameras, known for their compact size, high performance, and versatility. Here are some key features and details:

(a) **Sensor**:
- The MQ013RG-ON uses the ON Semiconductor AR0134 CMOS sensor.
- It is a 1.2-megapixel sensor with a resolution of 1280 x 960 pixels.
- The sensor size is 1/3 inch, which is a standard size for many industrial and scientific imaging applications.

(b) **Frame Rate**:
- The camera supports a high frame rate of up to 60 frames per second (fps) at full resolution.
- This makes it suitable for applications requiring fast image capture, such as high-speed industrial inspection or real-time monitoring.

(c) **Interface**:
- The camera features a USB 3.0 interface, providing high data transfer rates up to 5 Gbps.
- USB 3.0 ensures fast and reliable data transmission, making it ideal for high-resolution and high-speed imaging applications.

(d) **Compact Design**:
- The MQ013RG-ON is known for its extremely compact and lightweight design.
- The small form factor (26.4 mm x 26.4 mm x 21.6 mm) makes it easy to integrate into tight spaces or portable systems.

(e) **Low Power Consumption**:

- The camera is designed to operate with low power consumption, which is beneficial for portable and embedded systems where power efficiency is crucial.

(f) **Global Shutter**:

- The camera is equipped with a global shutter, which captures the entire image at once, eliminating motion artifacts and distortions.
- This is particularly important for capturing fast-moving objects accurately.

(g) **Image Quality**:

- The MQ013RG-ON offers good image quality with low noise and high sensitivity, making it suitable for various lighting conditions.
- It also supports multiple image enhancement features such as gain control, exposure control, and white balance.

(h) **Software Support**:

- XIMEA provides comprehensive software support, including drivers and SDKs for major operating systems like Windows, Linux, and macOS.
- The xiAPI SDK facilitates easy camera integration and control, and supports various programming languages.

(i) **Versatility**: These cameras are used in a wide range of applications including machine vision, robotics, medical imaging, microscopy, and scientific research due to their reliable performance and versatility.

4.1.2 Arenas

The design and construction of arenas suitable for containing the animals in a defined space during tracking (perhaps in the presence of visual stimulation of the individuals) requires the consideration of some key aspects such as:

- the material used to construct the arena should be transparent, in particular the top, which is the part through which the video camera will be recording the behaviour of the enclosed animals;
- the shape of the arena should be considered carefully, according to the intentions of the experimenter. In particular, circular shaped arenas are generally used in order to avoid having corners into which the animals may tend to find "refuge", which may result in the animals remaining immobile for most of the experiment time. On the other hand, circular open spaces may be a source of stress, and this should be kept in due consideration. In this respect, throughout the years, many other geometries for arenas used to study particular aspects of *Drosophila melanogaster* behaviour have

been proposed (see for example, Soibam et al., 2012 or Tom Mekdara et al., 2012).

- In the case of experiments involving insects, such as Drosophila, the height of the enclosing arena (i.e. floor to ceiling) requires a careful design in order to restrict the vertical motion of the flies by not allowing them to fly, but only walk. Also the height should decline towards the edges of the arena, so that the flies can only adopt an upright standing position throughout the arena, as described in the paper by Simon et al. (Simon and Dickinson, 2010).

4.2 Software

4.2.1 General Software

1. MATLAB

   MATLAB (Matrix Laboratory) is a powerful commercial software environment used for numerical computing, data analysis, algorithm development, and visualization. Here are its key features:

   (a) **Numerical Computing**:
   - Handles matrix and vector operations efficiently.
   - Supports complex arithmetic and advanced mathematical functions.

   (b) **Programming and Algorithm Development**:
   - Allows for the creation of scripts, functions, and classes using its own programming language.
   - Includes a vast library of built-in functions for mathematical and engineering calculations.

   (c) **Data Analysis and Visualization**:
   - Provides tools for data import, manipulation, and visualization.
   - Supports 2D and 3D plotting functions for creating graphs and charts.
   - Interactive apps for curve fitting, signal analysis, and other tasks.

   (d) **Toolboxes and Add-Ons**:
   - Extensive range of toolboxes for specific applications like signal processing, image processing, machine learning, and more.
   - Custom toolboxes can be created or downloaded from the MATLAB File Exchange.

   (e) **Simulink**:
   - An integrated environment for modeling, simulating, and analyzing dynamic systems.

- Used for multidomain simulation and Model-Based Design.

(f) **Interfacing and Integration**:

- Interfaces with other programming languages like C, C++, Java, Python, and more.
- Integrates with hardware for real-time data acquisition and control.

(g) **Performance**:

- High-performance capabilities with support for parallel computing, GPU acceleration, and large data sets.
- Optimization tools for improving code performance.

(h) **Documentation and Support**:

- Comprehensive documentation, tutorials, and examples.
- Active user community and technical support services.

Applications:

- **Engineering and Scientific Research**: Used extensively in academia and industry for research and development.
- **Data Science and Machine Learning**: Provides tools and functions for data preprocessing, machine learning, and deep learning.
- **Finance**: Used for quantitative analysis, modeling, and algorithmic trading.

This is the link to the MATLAB software download and documentation pages: MATLAB

2. Python

Python is a versatile and powerful open source programming language known for its simplicity and readability. Here are some key features:

(a) **Easy to Learn and Use**:

- Simple and clean syntax similar to English, making it beginner-friendly.

(b) **Interpreted Language**:

- Executes code line by line, which simplifies debugging.

(c) **Versatile and General-Purpose**:

- Suitable for web development, data analysis, scientific computing, automation, and more.

(d) **Extensive Standard Library**:

- Comprehensive library that supports many common programming tasks like file I/O, system calls, and data manipulation.

(e) **Cross-Platform Compatibility**:
   - Runs on various operating systems including Windows, macOS, and Linux.

(f) **Dynamic Typing**:
   - No need to declare variable types, which are determined at runtime.

(g) **Object-Oriented Programming (OOP)**:
   - Supports OOP concepts such as classes and inheritance, facilitating code reuse and organization.

(h) **Large Ecosystem and Community**:
   - Vast number of third-party libraries and frameworks (e.g., NumPy, Pandas, Django, Flask).
   - Active community providing support and development of new tools.

(i) **Integration Capabilities**:
   - Easily integrates with other languages (C/C++, Java) and technologies.

(j) **Support for GUI Programming**:
   - Libraries like Tkinter, PyQt, and Kivy for developing graphical user interfaces.

(k) **Strong Support for Scientific and Numeric Computing**:
   - Libraries like SciPy, NumPy, and Matplotlib for scientific research and data visualization.

(l) **Web Development**:
   - Frameworks such as Django and Flask for creating robust web applications.

(m) **Automation and Scripting**:
   - Frequently used for automating repetitive tasks and writing scripts to enhance productivity.

(n) **Data Science and Machine Learning**:
   - Popular libraries like TensorFlow, Keras, and scikit-learn for data analysis and machine learning projects.

This is the link to the Python programming software download and documentation pages: Python

3. R and RStudio

   R is a powerful scripting language specifically designed for statistical computing and data analysis. Here are its key features:

(a) **Statistical Analysis**:

- Extensive library of statistical functions for descriptive statistics, hypothesis testing, regression analysis, time-series analysis, and more.

(b) **Data Manipulation**:

- Tools for data manipulation and transformation, including the ‘dplyr‘ package for data wrangling.

(c) **Data Visualization**:

- Advanced graphics capabilities with packages like ‘ggplot2‘ for creating detailed and customizable plots.

(d) **Extensive Package Ecosystem**:

- Comprehensive repository (CRAN) of packages for various applications, including bioinformatics, machine learning, econometrics, and more.

(e) **Scripting and Automation**:

- Ability to write scripts for automating repetitive tasks and workflows.

(f) **Integration with Other Languages**:

- Interfaces with C, C++, Python, and SQL for extending functionality.

(g) **Open Source**:

- Free to use and open-source, with active contributions from a global community of developers and researchers.

(h) **Reproducible Research**:

- Supports literate programming and reproducible research with tools like R Markdown and Knitr for integrating code, text, and results in a single document.

(i) **Interactive Development Environment (IDE)**:

- RStudio provides a robust IDE with features like syntax highlighting, debugging tools, and version control integration.

(j) **Strong Community Support**:

- Active community forums, user groups, and extensive documentation and tutorials available online.

This is the link to the R programming software download and documentation pages: R

This is the link to the RStudio IDE for R download and documentation pages: RStudio

4. Inkscape

   Inkscape is a versatile and open-source vector graphics editor. Here are its key features:

   (a) **Vector Graphics Creation**: Uses Scalable Vector Graphics (SVG) as its primary format.
   (b) **Drawing Tools**: Includes a variety of tools for drawing, shaping, and manipulating objects, including freehand, Bezier curves, and calligraphy tools.
   (c) **Node Editing**: Allows for detailed manipulation of vector paths and nodes.
   (d) **Text Tools**: Provides text manipulation with support for various fonts and styles.
   (e) **File Compatibility**: Supports multiple formats like SVG, PDF, EPS, AI, and PNG.
   (f) **Extensions and Plugins**: Customizable with numerous extensions and plugins for added functionality.
   (g) **Multi-platform**: Available on Windows, macOS, and Linux.

   These features make Inkscape suitable for tasks such as logo design, technical illustrations, and web graphics.

   This is the link to the Inkscape download and documentation pages: Inkscape

5. OpenSCAD

   OpenSCAD is a unique 3D design software tailored for creating precise 3D models through scripting. Here are its key features:

   (a) **Script-Based Modeling**: Users create models using a scripting language, offering precise control over the design.
   (b) **Parametric Design**: Supports parametric design, allowing for easy adjustments and scalability of models.
   (c) **CSG (Constructive Solid Geometry)**: Uses CSG for combining primitive shapes into complex models.
   (d) **Preview and Rendering**: Provides real-time preview and high-quality rendering of designs.
   (e) **Cross-Platform**: Available on Windows, macOS, and Linux.
   (f) **Open Source**: Free to use and modify, with an active community contributing to its development.

   These features make OpenSCAD ideal for engineers, designers, and hobbyists who require precision and flexibility in their 3D modeling work.

Furthermore, the 3D models produced can be rendered into real-world objects because OpenSCAD models can be exported as files suitable for 3D printing.

This is the link to the Inkscape download and documentation pages: OpenSCAD

4.2.2 Specific Software

1. **Tracking software**

   (a) Ctrax: The Caltech Multiple Walking Fly Tracker

   Live (real time) unsupervised tracking of multiple moving objects contained in a single defined area of interest is a relatively complex task. In particular it is a task which requires supervision and a large amount of computing power, making it at best a slow process. Consequently, closed loop experiments based on the response to the tracked behaviour of multiple animals contained in a single delimited arena is not really feasible given the above mentioned difficulties.

   On the other hand, software packages such as Ctrax are specifically designed to efficiently perform offline supervised tracking of multiple animals contained in a single delimited arena. In this case operator supervision is necessary to aid in correcting small tracking errors.

   Ctrax is a freely available, cross-platform, open source standalone software package.

   This is the link to the Ctrax software download and documentation pages: Ctrax.

   (b) MARGO: Massively Automated Real-time GUI for Object-tracking

   MARGO is scripted in the MATLAB programming environment. The GUI-based application allows to simultaneously track (in real time) up to several hundred individuals, each contained within a delimited arena. As an example, in the case of Drosophila this would typically be done with individual flies housed within the wells of 48 or 96-well flat-bottomed cell culture plates.

   Interestingly, this type of setup **does** allow for closed-loop experiments involving multiple animals. This is because simultaneous real time tracking of even hundreds of individuals is possible when the single individuals are confined to "private" non-overlapping arenas.

   In fact the main issue with tracking many animals within the same arena boils down to the computer intensiveness of the algorithms designed to resolve trajectories of animals whose paths cross during the tracking period. The task becomes a lot less computer-intensive if it is not necessary to keep the identity of the individuals separate during the tracking. Thus MARGO also allows to (live-) track multiple individuals housed in a single arena, albeit without keeping the identity of the individuals separate.

This is the link to the MARGO script download and documentation pages: MARGO.

2. **Visual Stimulation software**

   (a) PsychoPy®

   PsychoPy® is a very powerful open-source package for running experiments in Python (a real and free alternative to Matlab). PsychoPy® combines the graphical strengths of OpenGL with simple Python syntax to give scientists a free and simple stimulus presentation and control package. It is used by many labs worldwide for psychophysics, cognitive neuroscience and experimental psychology.

   This is the link to the PsychoPy® download and documentation pages: PsychoPy

5 Bibliography

Reiser, Michael B. and Michael H. Dickinson (Jan. 2008). "A modular display system for insect behavioral neuroscience". In: *Journal of Neuroscience Methods* 167.2, pp. 127–139. ISSN: 0165-0270. DOI: 10.1016/j.jneumeth.2007.07.019. URL: https://www.sciencedirect.com/science/article/pii/S0165027007003755 (visited on 06/22/2024).

Simon, Jasper C. and Michael H. Dickinson (Jan. 2010). "A New Chamber for Studying the Behavior of Drosophila". en. In: *PLOS ONE* 5.1. Publisher: Public Library of Science, e8793. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0008793. URL: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0008793 (visited on 06/24/2024).

Soibam, Benjamin et al. (2012). "Modeling Drosophila positional preferences in open field arenas with directional persistence and wall attraction". eng. In: *PloS one* 7.10, e46570. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0046570.

Tom Mekdara, Nalong et al. (July 2012). "A novel lenticular arena to quantify locomotor competence in walking fruit flies". eng. In: *Journal of experimental zoology. Part A, Ecological genetics and physiology* 317.6, pp. 382–394. ISSN: 1932-5231. DOI: 10.1002/jez.1731.