

## Automated tracking of aquatic crustaceans with potential application on the quantification of animals movement

JESÚS D. NUÑEZ<sup>1</sup>✉; OCTAVIO MASSONE<sup>1</sup> & JOSÉ A. GARCÍA<sup>2,3</sup>

<sup>1</sup>*Instituto de Investigaciones Marinas y Costeras (IIMyC), FCEyN, Universidad Nacional de Mar del Plata-CONICET. Mar del Plata, provincia de Buenos Aires, Argentina.* <sup>2</sup>*Universitat Oberta de Catalunya (UOC).* <sup>3</sup>*Institute of Marine Sciences, Spanish National Research Council (ICM-CSIC).*

**ABSTRACT.** Here, we present a set of algorithms using the Python programming language, that will allow using a routine for object detection and tracking in experimental videos. We developed a script, under the fundamentals of background subtraction and image thresholding (using the OpenCV package), that makes it possible to track a wide spectrum of animals under different conditions. We have validated this script through testing on semi-terrestrial and aquatic crustacean species and under different experimental scenarios (laboratory and field sampling and using video created under nocturnal and diurnal conditions). The open-source nature of the script allows for flexibility and scalability, so it can be easily customized and is thus transferable to other species/experiments in the context of behavioral ecology. The tracking script is easy customizable and free alternative to commercial video tracking systems and therefore, applicable to a wide variety of both educational and research programs.

[Keywords: Python, OpenCV, object detection, behavioral ecology, video tracking]

**RESUMEN.** Seguimiento automatizado de diferentes crustáceos acuáticos con potencial aplicación a la cuantificación de movimiento de animales. En este trabajo se describe un conjunto de algoritmos escritos en el lenguaje de programación Python, orientados a la detección y el seguimiento autónomo de objetos en videos experimentales. Esta rutina se desarrolló bajo los fundamentos de la sustracción de fondo y umbral de imagen (empleando la biblioteca OpenCV), lo cual posibilita seguir un espectro amplio de animales bajo diferentes condiciones. La rutina fue validada usándola en crustáceos acuáticos y semi-terrestres, y bajo diferentes escenarios experimentales (muestreo de laboratorio, de campo y utilizando videos creados en condiciones nocturnas y diurnas). Una de las ventajas de la rutina que presentamos es que fue construida empleando un *software* de código abierto y multiplataforma, por lo que se la puede emplear en cualquier sistema operativo. La naturaleza de código abierto de esta rutina permite flexibilidad y escalabilidad, y fácil personalización. Por lo tanto, es transferible a otras especies o experimentos en el contexto de la ecología del comportamiento. La presente rutina constituye una alternativa gratuita a los sistemas comerciales de seguimiento de video y, en consecuencia, se la puede aplicar a una gran variedad de programas, tanto educativos como de investigación.

[Palabras clave: detección de objetos, ecología del comportamiento, seguimiento por video]

## INTRODUCTION

Quantification of movement is fundamental to answer research questions in many branches of biology (e.g., ethology, behavioral ecology, ecophysiology, biomechanics, neuroscience and evolutionary ecology). Whether through field observations and/or laboratory experiments, quantification of movement is largely used as a proxy to assess their responses to changes in the environment. For example, in the last decades, movement parameters such as locomotion activity have been widely used to evaluate hypotheses focusing on ecological characteristics driven by environmental changes such as predator avoidance, social interaction (e.g., Audira et al. 2020), innate response to con- and heterospecific alarm chemicals (e.g., Ituarte et al. 2022), anthropogenic stresses (e.g., Thoré et al. 2021) and in routine chemical risk assessment procedures (e.g., Antunes and Biala 2012).

Researchers need to acquire methods that enable non-invasive motion measurements under different scenarios and with high accuracy (Dell et al. 2014). Thus, the ability to process video plays a fundamental role in the development of methodologies related to measuring an animal's locomotor activity. In recent decades, with the advancement of technology and computer development, a wide range of tools have emerged for use in the autonomous tracking of animals in a variety of contexts. These techniques drastically reduce the time and effort needed to extract image-based data from videos, allowing researchers to generate large and highly detailed datasets on different types of behavior (e.g., García et al. 2019; Nuñez et al. 2016; Sbragaglia et al. 2013). Also, these techniques allow the quantification of behaviors that might go unnoticed by manual observation, such as rapid movements, infrequent behaviors or locomotion patterns that emerge over long periods of time (Noldus et al. 2001). Nowadays, autonomous tracking methods for experimental videos abound in the literature, but researchers use them infrequently, either because they lack the knowledge to implement these tools and adapt them to their specific needs or because of how expensive it can be to obtain licenses for their use (in the case of a paid software). Thus, the implementation of these techniques can be difficult in practice, leading to the use of traditional techniques of manual data collection through

direct observation. However, while direct observation can produce informative data, it is time-consuming, subjective and difficult to replicate. Therefore, there is a need to develop script/programs for autonomous monitoring of organisms that are: 1) free of charge, 2) easy to implement, and 3) reliable. In this context, the aim of this work is to describe and validate a set of algorithms (scripts) of simple, free and open source implementation, using the Python language and R for the detection and tracking of animals in experimental videos. This set of algorithms are developed by background subtraction technique in a background model image is created and then the changes in pixels of successive images were evaluated. The advantages of this type of algorithms are that they are simple to implement, give good results and do not require a great use of computational capabilities and previous training as the deep learning image techniques. Therefore, they are ideal for versatile usage and can be suitable as a simple solution for species tracking in the field and in the laboratory.

In order to demonstrate its high-efficiency with tracking, we used different species and stadium of crustacean as well as different experimental contexts (field and laboratory conditions) in order to have to cope with different difficulties when analyzing the videos. For example, we used adults of the semi-terrestrial crab *Neohelice granulata* in field and under night conditions (Nuñez et al. 2021). Also, under laboratory conditions, we used a larvae stadium of this specie which has the difficulty of being smaller than 2 mm. On the other hand, we use a freshwater shrimp *Palaemon argentinus* which has the difficulty that their body coloration has very little contrast with the background. Finally, we discuss our results and suggest future improvements.

## ALGORITHM AND COMMANDS

This paper describes a set of algorithms developed in Python using the OpenCV library (Bradski 2000) (opencv.org) for image processing and some R functions for further analysis. Basically, the Python scripts works by automatically detecting objects in the videos using adaptive thresholding and isolates animals from other objects based on their contrast with the environment. This script evaluates the location of an animal in the course of a single session and continues. For this, the principle of background subtraction is

used, which consists of taking an image of the scene without movement and subtracting the successive frames that we obtain from a video. In the last years, background subtraction algorithms have been improved in many aspects. For example, in our case, we use the `BackgroundSubtractorMOG2()` function of OpenCV (see Supplementary Material 1). This function has two parameter settings that can be modified to improve tracking accuracy under different conditions. The first one is the previous history parameter (first parameter in `fgbg` of the script) (Supplementary Material 1) that is the previous history of images used to generate a background model image. The second one is the statistical threshold (second parameter in `fgbg` of the script) (Supplementary Material 1) that evaluates the differences between the background model and the target image where we want to see if movement has occurred, and is calculated using Mahalanobis distance squared (Mahalanobis 1936). Thus, the adjustment of these two parameters will allow us to correct motion detection failures in different experimental situations and with different species. There is no definite rule and this adjustment will depend on the expertise acquired by performing different experiments under different situations and with different animals. Thus, in the example cases shown in this publication, the assigned values have been 1000 and 45 respectively (see Supplementary Material 1). Once a thresholded image is obtained, the routine draws contours around the target object (in our case a crustacean). Here, two filters are applied on binarized image, a previous static one in which the region of interest in the video is determined, leaving aside everything that is outside it, which saves us calculations and computational capacity of the machine by increasing the processing speed. And the other one, is applied in relation to the object size, since the organisms to be tracked have a certain size range and everything outside that size is not tracked, which is a classic computer vision techniques.

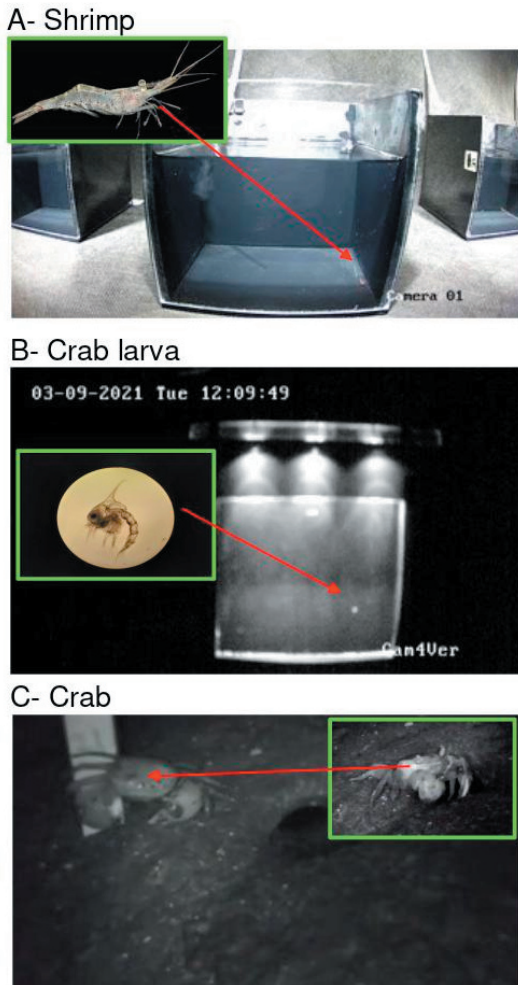
Then, it uses the centroid of the object as position and from there it starts to identify it in each frame by recording the X and Y position in space. With the data position obtained, several descriptive parameters of the locomotion of the animals can be calculated. For example, using some R functions we estimated the accumulated distance travelled, the average speed and the angles of movement (see Supplementary Material

1). The script was not developed to tracking multiple individuals. The tracking of various individuals using this script is possible as long as they have a physical boundary between them (different aquariums, separators, etc.), so that we can determine as many ROI as individuals we want to track (e.g., Nuñez et al. 2016). Besides the above-mentioned, we suggest having into account some final considerations before starting any video or photographic filming such as lighting, the type of light, the objects and environments targeted for filming or photography in order to minimize the 'noise' of video and maximized the object aim contrast.

#### *Script validation*

To validate the animal tracking performance of the Python script, results obtained autonomously were compared with those obtained by a trained observer. To further evaluate the plasticity of the script, we used videos under different scenarios and species: under artificial white light (simulating natural light) with a shrimp and infrared light with a crab larva (simulating night situation) under laboratory conditions and under nighttime conditions with a crab at field conditions (Figure 1).

For the cases under darkness conditions (both field and laboratory), the videos were recorded at standard rate (30 frames per second) using a high-resolution camera (HD 1024x720 pixels) sensitive to the infrared (IR) band. In the case of the laboratory, to improve contrast (due to the size of the larvae, smaller than 2 mm), an additional infrared light system was placed over the experimental aquarium. In all cases, the videos were made in .avi or .mp4 formats (although any video format can be used). The videos were edited by reducing the frame rates with `ffmpeg` library (v2.8.14) to 1 frames/second. With the output video, the position of the organisms per second was calculated manually using the free software ImageJ (Schneider et al. 2012) and autonomously with the Python script. For both cases (autonomous and manual), the euclidean distance was then calculated as described above, motion path parameters (the distances travelled and instantaneous velocities) were calculated and plotted (using the `ggplot2` package of the R software) and visually compared. The distances travelled were obtained by summing the euclidean distances calculated by both methods, while



**Figure 1.** Experimental set under different scenarios and species. A) Under artificial white light (simulating natural light) with a shrimp at laboratory conditions. B) Under infrared light with a crab larva (simulating night situation) at laboratory conditions. C) Under infrared light at night conditions with a crab at field.

**Figura 1.** El experimento bajo diferentes escenarios y especies. A) Bajo luz blanca artificial (simulando la luz natural) con un camarón en condiciones de laboratorio. B) Bajo luz infrarroja con una larva de cangrejo (simulando la situación nocturna) en condiciones de laboratorio. C) Bajo luz infrarroja en condiciones nocturnas con un cangrejo en el campo.

the instantaneous velocity was calculated as an average of the distance travelled for 10 consecutive frames. We evaluated the object detection against the movement data (manually annotated and autonomous) as described above and calculated Precision (Equation 1), Recall (Equation 2) and F1 (Equation 3). The Precision is the rate of true positives relative to total detections, and the Recall is the rate of detection of true positives. We used the F1 score (the harmonic mean of the Precision and Recall) to assess

the performance of our crustacean tracking detection under different conditions.

$$\text{Precision} = \frac{\text{truepositives}}{(\text{truepositives} - \text{falsepositives})} \quad \text{Equation 1}$$

$$\text{Recall} = \frac{\text{truepositives}}{(\text{truepositives} - \text{falsenegatives})} \quad \text{Equation 2}$$

$$\text{F1} = 2 * (\text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall}) \quad \text{Equation 3}$$

Results showed that the routine was able to track the position of the animals in all trials under all experimental scenarios (laboratory, field, dark, natural light and different species). Validation of the stand-alone routine with a manual method for both (estimation of accumulated distances travelled and instantaneous speeds) showed excellent performance in all cases; both methods showed similar results (Table 1, Figure 2). It is especially noteworthy that the script worked effectively in field conditions and with animals smaller than 2 mm (i.e., crab larva) (Table 1, Figure 2).

By using the Python script to detect individuals, we obtained 81.5% of Precision value and an F1 score of 81% for crabs adults (Table 1). The object detection model missed 7 crabs (false negatives) and misidentified 8 objects as crabs (false positives) out of the 39 manually registered individuals. In the case of larvae phase, we obtained 82.8% Precision value and an F1 score of 100% (Table 1). The object detection model missed 0 larvae (false negatives) and misidentified 5 objects as larvae (false positives) out of the 21 manually registered individuals. Finally, we obtained 89.4% Precision value and an F1 score of 81.1% to adult shrimp (Table 1). The object detection model missed 6 shrimp (false negatives) and misidentified 1 objects as shrimp (false positives) out of the 21 manually registered individuals.

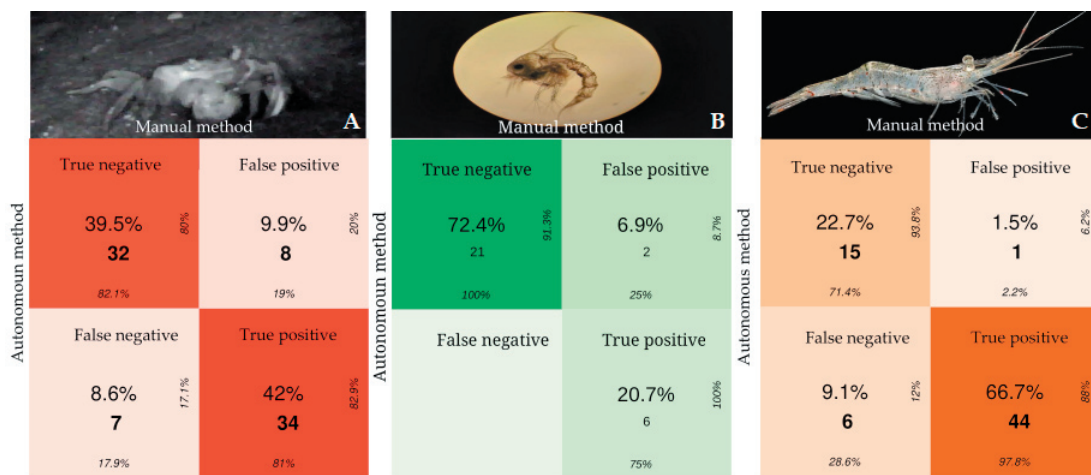
#### *An example*

To describe how the Python/R script works, a 30-seconds experimental video was selected where a crab larva is continuously filmed under darkness conditions (using a camera and infrared light illumination). The video was pre-treated to reduce the frame rate (1 fps) and increase the tracking accuracy. We used the Python script to track the position of the individual during the filming period (see Material supplementary 2), and with the

**Table 1.** Validation of the autonomous script with a manual method using a confusion matrix and accumulative estimation of distances travelled for the three situations (i.e., adult and larvae crab and adult shrimp), which are shown as individuals count, where the true positive means an individual tracked was detected when it appeared on a video frame; otherwise, it was false negative. False positives were misidentified objects that were detected and tracked.

**Tabla 1.** Validación de la rutina autónoma con un método manual mediante la de una matriz de confusión y las distancias acumuladas recorridas para las tres situaciones (i.e., cangrejo adulto y larva y camarón adulto), que se muestran como recuentos de individuales, donde el verdadero positivo significa que un individuo rastreado fue detectado cuando apareció en un fotograma de vídeo; de lo contrario, fue falso negativo. Los falsos positivos fueron objetos mal identificados, detectados y rastreados.

Species	Confusion matrix values			Distances travelled (m)	
	Precision	Recall	F1	Autonomous tracking	Manual tracking
Crab	0.815	0.821	0.810	18.32	17.05
Larva	0.828	1	0.894	11.87	10.97
Shrimp	0.894	0.714	0.811	5.78	6.13



**Figure 2.** The confusion matrix is used to summarize the performance of the autonomous routine compared to manual method. A) In dark red colored diagonal of the matrix are the correct adult crab classifications, whereas all other entries are misclassifications. B) In dark green colored diagonal of the matrix are the correct larvae classifications, whereas all other entries are misclassifications. C) In orange blue colored diagonal of the matrix are the correct shrimp classifications, whereas all other entries are misclassifications.

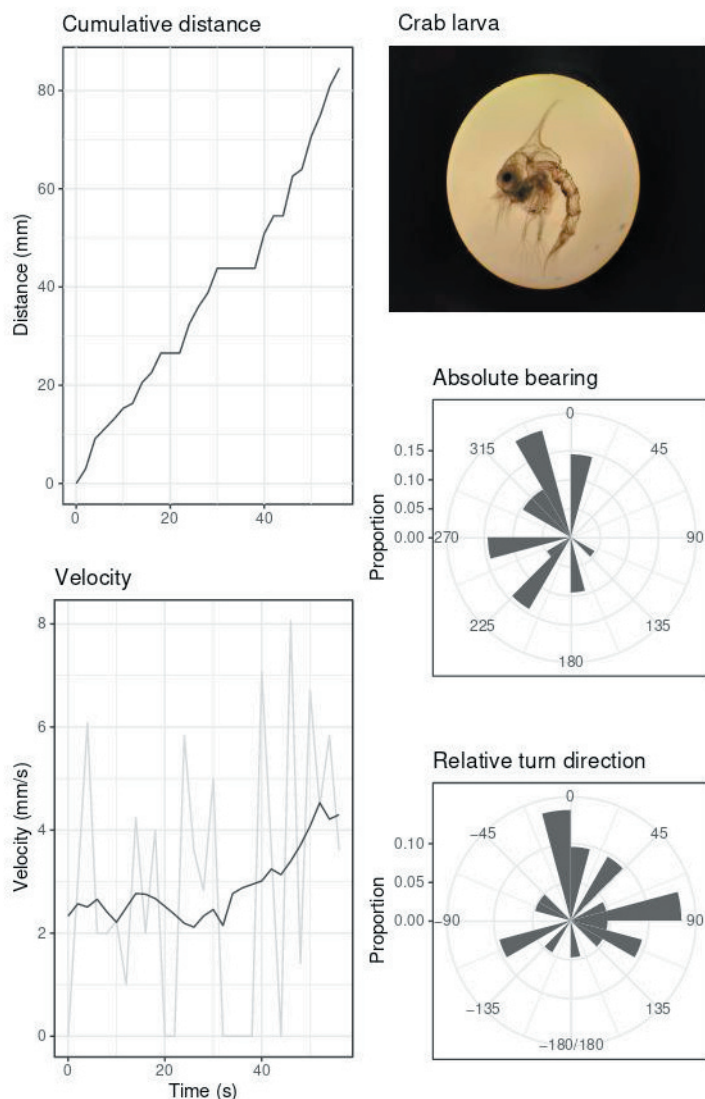
**Figura 2.** La matriz de confusión se utiliza para resumir el rendimiento de la rutina autónoma en comparación con el método manual. A) La diagonal de color rojo oscuro de la matriz representa las clasificaciones correctas de cangrejos adultos, mientras que todas las demás entradas son las clasificaciones erróneas. B) La diagonal de color verde oscuro de la matriz representa las clasificaciones correctas de las larvas, mientras que todas las demás entradas son las clasificaciones erróneas. C) La diagonal de color naranja oscuro de la matriz representa las clasificaciones correctas de camarones, mientras que todas las demás entradas son las clasificaciones erróneas.

R script, the accumulated distance travelled, the average speed and the angles of movement (absolute and relative) were calculated (Figure 3).

## MAIN CONSIDERATIONS

Here, we describe a simple to implement set of algorithms (routine), developed in Python language, for object detection and tracking in experimental and field videos to quantify locomotor behaviors using tools from the Open Source Computer Vision (OpenCV) package. The script can be used from multiple platforms (Spyder, Jupyter Notebook or simply from the terminal from any Linux or Windows device).

Here we use the R reticulate package (Ushey et al. 2021) and run it in the Rstudio environment (R Core Team 2020). The code in Python format is organized into discrete, ordered sections that can be run independently by the user (from the computer terminal or interpreters mentioned above). Each section has a brief explanation that informs the user what each command does from a conceptual point of view, also indicating what can be modified according to the user's needs (see Supplementary Material 1). This allows the user to understand the algorithms conceptually without having to read the whole code and, at the same time, the user is free to modify the algorithms if desired.



**Figure 3.** Estimated locomotory parameters. A) The cumulative distance reveals the total distance travelled in the observation period. B) Speed indicates the changes in locomotor activity during the observation period. C) The absolute bearing, that is the sum of the absolute angles between each movement vector of the crab larva. D) The relative turn direction, that is the sum of the relative angles between each movement vector of the crab larva.

**Figura 3.** Parámetros estimados de locomoción. A) Distancia acumulada, que es la distancia total recorrida en el periodo de observación. B) Velocidad, que indica los cambios en la actividad locomotora durante el periodo de observación. C) Orientación absoluta, que es la suma de los ángulos absolutos entre cada vector de movimiento. D) Orientación relativa, que es la suma de los ángulos relativos entre cada vector de movimiento de la larva de cangrejo.

The script was designed to process and analyze video acquired from virtually any device (including mobile phones, notebooks and video cameras). In order to demonstrate its high-compatibility tracking, it was validated by using it on aquatic and semi-terrestrial crustaceans and under different experimental scenarios (laboratory and field sampling and using videos created under night and day conditions). The video tracking results show that the proposed script is an effective, accurate (in all the cases  $F1 > 0.8$ ) and non-invasive way to detect and track small-scale movement of crustaceans under different environments condition. The different scenarios proposed in this work demonstrate the great plasticity of this routine by obtaining similar results when compared to a manual method. It also suggests a great potential for the quantification of other animals and conditions. Moreover, this script

has already been successfully implemented in works with infrared lighting systems in studies of circadian rhythms (Sbragaglia et al. 2013; Nuñez et al. 2016). Particularly to works made in the natural environment, some limitations should be considered: the script would be effective if made with focal videos (one individual), and although it is adjustable (see materials and methods), the background should have enough distinctive texture to allow feature-based image warping and avoiding background 'noise' (such as other individuals, animals, or plants in movement, water reflection). Despite these limitations, there are currently few if any existing computational methods to tracking natural behavior in the field that can be plastic enough to be used in both laboratory and natural conditions such as the one presented here.

One great advantage of the script here presented is that it was built using open source and cross-platform software (it can be used on any operating system). The open source nature of this script allows for flexibility and scalability, making it easily customizable and thus transferable to other species/experiments in the context of behavioral ecology and therefore applicable to a wide variety of educational and research programs.

## FINAL CONSIDERATIONS

The routine described in this paper offers a simple, fast and versatile solution to many of the video-based tracking problems commonly encountered in the field of ethology. Autonomous video-based motion quantification allows for high sampling rate, wide spatial resolution and measurements over long periods of time, as well as eliminating

sources of error related to observer bias and allowing for transparent and reproducible workflows. One of the disadvantages of this script is that in currently version, is only able to track a single animal at a time. We plan to include the ability to track multiple animals simultaneously in the future. This excludes the use of this script in work with objectives related to interactions between animals. However, García et al. (2019) report a detailed description of how to implement a similar script (based on the same concepts and algorithms) in work on social interactions following several animals simultaneously.

**ACKNOWLEDGEMENTS.** This research was funded by Agencia Nacional de Promoción Científica y Tecnológica, Project PICT-2017-373. We are also grateful to the two highly dedicated reviewers and the Associate Editor for their constructive comments, which helped us to improve the manuscript.

## REFERENCES

- Antunes, M., and G. Biala. 2012. The novel object recognition memory: neurobiology, test procedure, and its modifications. *Cogn Process* 13:93-110. <https://doi.org/10.1007/s10339-011-0430-z>.
- Audira, G., N. T. Ngoc Anh, B. T. Ngoc Hieu, N. Malhotra, P. Siregar, O. Villalobos, O. B. Villaflores, T. R. Ger, I. C. Huang, K. H. Chen, and C. D. Hsiao. 2020. Evaluation of the adverse effects of chronic exposure to donepezil (an acetylcholinesterase inhibitor) in adult zebrafish by behavioral and biochemical assessments. *Biomolecules* 10(9): 1340. <https://doi.org/10.3390/biom10091340>.
- Bradski, G. 2000. The OpenCV Library | Dr Dobb's. URL: [drdobbs.com/open-source/the680opencv-library/184404319](http://drdobbs.com/open-source/the680opencv-library/184404319).
- Dell, A. I., J. A. Bender, K. Branson, I. D. Couzin, G. G. de Polavieja, L. Noldus, A. Pérez-Escudero, P. Perona, A. D. Straw, M. Wikelski, and U. Brose. 2014. Automated image-based tracking and its application in ecology. *Trends in Ecology and Evolution* 29(7):417-428. <https://doi.org/10.1016/j.tree.2014.05.004>.
- García, J. A., V. Sbragaglia, D. Masip, and J. Aguzzi. 2019. Long-term Video Tracking of Cohoused Aquatic Animals: A Case Study of the Daily Locomotor Activity of the Norway Lobster (*Nephrops norvegicus*). *Journal of Visual Experiment* 146:e58515. <https://doi.org/10.3791/58515>.
- Iriarte, R., Nuñez, J. D., Spivak, E. D., and C. C. Bas. 2022. Innate response to con- and heterospecific alarm chemicals in juveniles of a freshwater shrimp. *Journal of Zoology* 318(2):104-113. <https://doi.org/10.1111/jzo.13001>.
- Mahalanobis, P. C. 1936. On the Generalized Distance in Statistics. *Journal of Zoology*. Proceedings of the National Institute of Science of India 2:49-55.
- Noldus, L., J. Spink, and R. Tegelenbosch. 2001. EthoVision: A versatile video tracking system for automation of behavioral experiments. *Behavior Research Methods, Instruments and Computers* 33(3):398-414. <https://doi.org/10.3758/BF03195394>.
- Nuñez, J. D., V. Sbragaglia, J. A. García, J. B. Company, and J. Aguzzi. 2016. First laboratory insight on the behavioral rhythms of the bathyal crab *Geryon longipes*. *Deep Sea Research I* 116:165-173. <https://doi.org/10.1016/j.dsr.2016.08.007>.
- Nuñez, J. D., V. Sbragaglia, P. D. Ribeiro, E. Spivak, and T. A. Luppi. 2021. The magnitude of behavioral responses to artificial light at night depends on the ecological context in a coastal marine ecosystem engineer. *Marine Environmental Research* 165:105238. <https://doi.org/10.1016/j.marenvres.2020.105238>.
- R Core Team. 2020. A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria.
- Sbragaglia, V., J. Aguzzi, J. A. García, D. Sarriá, S. Gomariz, C. Costa, P. Menesatti, M. Vilaró, A. Manuel, and F. Sardá. 2013. An automated multi-flume actograph for the study of behavioral rhythms of burrowing organisms. *Journal of Experimental Marine Biology and Ecology* 446:177-185. <https://doi.org/10.1016/j.jembe.2013.05.018>.
- Thoré, E. S. J., L. Brendonck, and T. Pincheel. 2021. Natural daily patterns in fish behaviour may confound results of ecotoxicological testing. *Environmental Pollution* 276:116738. <https://doi.org/10.1016/j.envpol.2021.116738>.
- Ushey, K., Allaire J. J., and Y. Tang. 2021. reticulate: Interface to 'Python'. R package version 1.20. URL: [cran.r-project.org/package=reticulate](http://cran.r-project.org/package=reticulate).
- Schneider, C. A., W. S. Rasband, and K. W. Eliceiri. 2012. NIH Image to ImageJ: 25 years of image analysis. *Nature Methods* 9:671-675. <https://doi.org/10.1038/nmeth.2089>.