Aug 30, 2024

🌐 Protocol for the preparation of Wings and Wing Interference Pattern Imaging from ethanol preserved specimens

Saskia B. Bartsch[1], Jonathan Vogel[1], Anne Ankermann[1], Christoph Mayer[1], Ralph S. Peters[1]

[1]Leibniz Institute for the Analysis of Biodiversity Change, Museum Koenig Bonn

Saskia B. Bartsch: Preparation of the first draft of the protocol
Jonathan Vogel: Adjusting the protocol with tweaks, preparation of final protocol
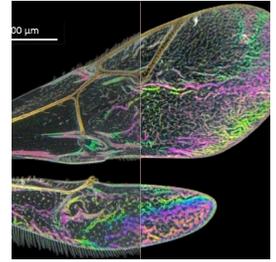Anne Ankermann: Contributing the automated clean-up with Python in Section 3
Christoph Mayer: Supervision
Ralph S. Peters: Supervision

Jonathan Vogel

Leibniz Institute for the Analysis of Biodiversity Change, M...

Create & collaborate more with a free account

Edit and publish protocols, collaborate in communities, share insights through comments, and track progress with run records.

Create free account

Protocol status: In development
We have been using this protocol yielding satisfactory results in many cases. However, there is most definitely room for improvement.

Created: December 07, 2023

Last Modified: August 30, 2024

Protocol Integer ID: 91948

Keywords: Wing interference patterns, WIPs, Flying insects, Preservation, Small insects, wing interference pattern imaging from ethanol, wing interference pattern imaging, wing specimen, easy cleaning of the wing specimen, secure reproducibility of the imaging, storage of the specimen, prepared wing, quality wip image, specimen, ethanol, protocol for the preparation, imaging, decades after preparation, wing, image, preparation

Disclaimer

Abstract

This protocol aims to provide standardisation for the necessary steps, specifically to a) produce standardised, artifact-free and high-quality WIP images, b) secure long-term-storage of the specimens, c) make the prepared wings re-traceable to the specimens they have been taken from d) secure reproducibility of the imaging even decades after preparation (i.e., easy cleaning of the wing specimens prior to imaging), and to e) allow reproduction of the protocol with minimal financial resources. In all these aspects, there is undeniably room for improvement.

Image Attribution

Preparation of *Anacharis* wing interference patterns: Before and After. By Jonathan Vogel.

Materials

Section 1 - Slide preparation
- Stereo microscope
- Ethanol-preserved specimens
- Size 0 insect pin
- Small petri dish or staining block
- Microscopic slide (1 per specimen)
- Coverslip (1 per specimen)
- Ethanol (few drops per specimen)
- Microfiber cloth
- Fine Forceps
- Clear nail polish
- Small 00- or 000-type brush
- 📄 Alignment sheet.pdf 13KB

Section 2 - Imaging
- Zeiss Discovery.V20 SteREO microscope (60mm focal width lens)
- AxioCam HR R3-ZEN pro 2012 software (CARL ZEISS AG)
- Custom ring light and dome system (see Kawada and Buffington (2016))
- Black Cardboard

Section 3 - Image Processing (req. software)
- GNU Image Manipulation Program (GIMP development team)
- Python package OpenCV (cv2, cf. Bradski 2000)
- Python package imutils v0.5.4 (cf. Rosebrock 2015)
- Python package NumPy (cf. Harris et al. 2020)
- Python package Pandas (Pandas Development Team 2020)

Troubleshooting

Before start

The protocol is tested for both DNA-lysed and unlysed samples in high-% EtOH. While both scenarios worked, we found it a little easier to detach the wings (step 1) from the lysed samples.

Slide preparation `5m`

1 Using the microscope, transfer the ethanol-preserved specimen into a small petri dish or staining block filled with ethanol.
Dissect the right fore- and hindwings of the specimen by grabbing the base of the forewing with the fine forceps and spinning the wing while pushing the specimen into the opposite direction until the wing detaches.
Repeat with the hindwing. `1m`

2 Transfer the remaining specimen back into its labelled tube or proceed with mounting the specimen. `30s`
Clean a microscopic slide with ethanol and a microfiber cloth.

Position the printed alignment sheet (📄 Alignment sheet.pdf 13KB) underneath the slide.

3 Place a small drop of ethanol in the centre of the slide. `30s`
Under the microscope, using forceps, transfer the fore- and hindwings into the ethanol drop.

4 Align the fore- and hindwings according to the alignment sheet (see step 5) for standardised positioning and easier comparison. Make sure the fore- and hindwings do not overlap and are stretched out as good as possible. If too little ethanol is used on the slide, the wings are prone to rip apart when moved. `1m` ⚠️

5 Gently place the coverslip on top to protect the wings from dust and damage for long-time storage. `30s` ⚠️
Add an additional drop of ethanol on the coverslip's edge and gently push the slip down with the forceps right above the wings to push potential air bubbles to the side. Avoid pushing too hard to not break the coverslip and not adhese the glasses. The latter is important as the WIPs will not show otherwise.
Allow the ethanol to evaporate.

6 Gently apply a small amount of clear nail polish (enough to stabilize the slip but without it reaching the wings) to the coverslip edges using a small 00- or 000-type brush and let the nail polish harden. `30s`

7 Label the slide to link the wings to the corresponding specimen. `1m`

Imaging

8 See Materials for hard- and software used and prepare accordingly.
Configure the camera settings:
   - Exposure time: 100ms

- Saturation: 20 (maximum)
- Contrast: Maximum
- Magnification: 34.5X
- Colour mode: RGB
- Histogram: Best fit

9   Put the slide with the wings on the ring light, the black painted cardboard behind it and the dome on top

10 Bring the wing in focus, adjust white balance (Auto) and snapshot the image

11 Apply a scalebar to the image and export the image in a file format of choice

Image Processing

12 In GIMP, adjust luminance values by navigating to Colours/Levels. A window will pop up. Drag the min input level slider (black triangle) to the lowest visible margin of the input level curve and drag the max input level slider (white triangle) to the highest visible margin of the input level curve (see image). The visibility of the curve can be improved by dragging the window to full screen.
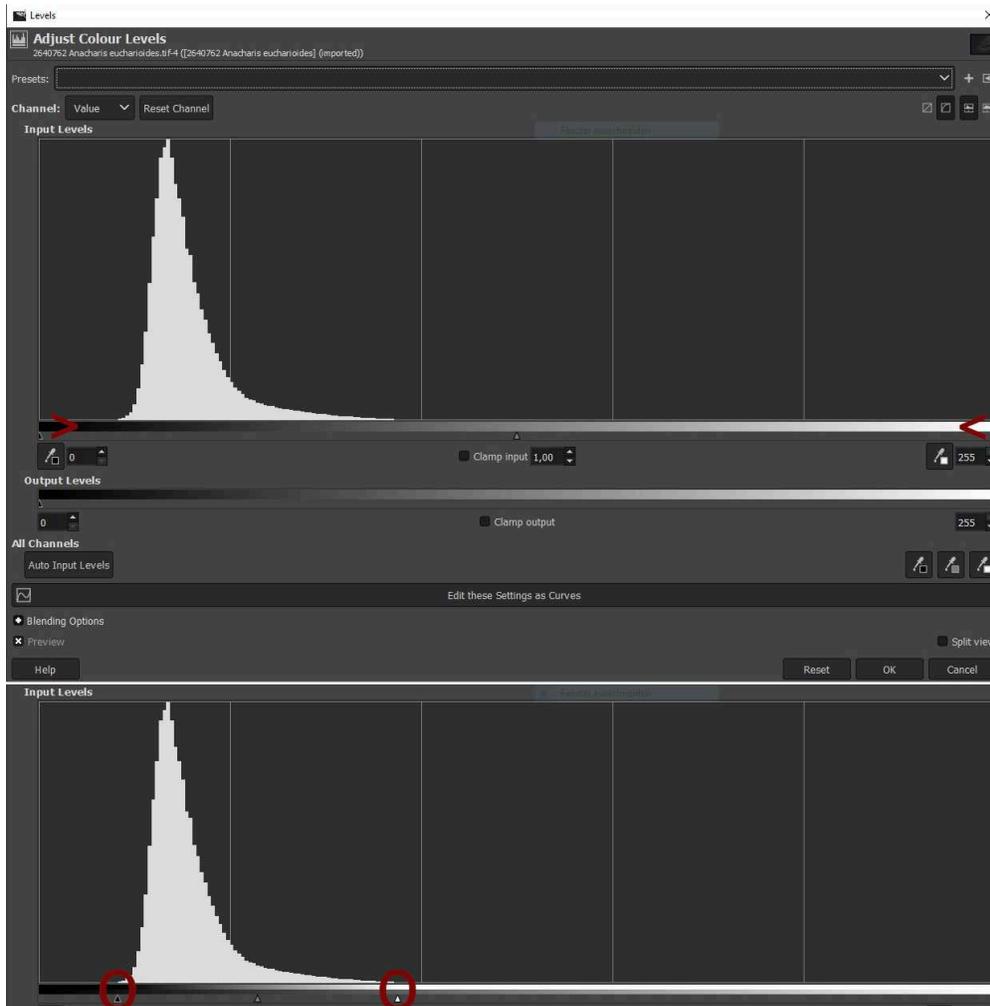
Adjust the colour levels (luminance values) by dragging the black and white triangle closer to the input curve.

13   The python scripts of the following steps can be accessed via https://github.com/aankm/WIP_auto-img.

Isolate the wings: In order to remove extraneous signal from the images, isolate the wings from the
background. As is, the script will save the forewing and hindwing for each individual in separate files. The two wings are processed separately and recombined to a final image in a standardised way after the following sub-steps.

Note: To maintain the wings' orientation to each other, combine both contours.

→ 01-extract_foreground.py

13.1   Create a mask from the contours detected in the image with the edge detection function cv2.Canny() by overlaying the edges with small circles of the same solid colour (grey). The size of the circles should be selected so that all edges within the desired foreground

area connect into one shape. Fill all areas outside this shape with a solid colour. If the background is too noisy, this step may not work as desired.

13.2 Find all contours (read: continuous objects) within the mask using cv2.findContours(), and sort them by area size in decreasing order.

13.3 Fill the largest contour of the mask (the forewing) in a different solid colour (white). Use this mask to black out all areas of a copy of the original image that are not part of the foreground object.

13.4 Repeat the preceding step for the second largest contour (the hindwing).
Note: If the forewing and hindwing areas intersect, or one part is missing (due to lost or generally missing wing as in, for example, Diptera), make these steps conditional on the size of the contours.

14 If magnification and resolution differ within the series of images: Re-size images to put them to scale.

15 Standardize orientation: Rotate the images so that the back margin of the forewing and the front margin of the hindwing were horizontal. The rotation function was customised from an OpenCV interactive grabcut algorithm available at https://github.com/opencv/opencv/blob/master/samples/python/grabcut.py The program opens a new window and prompts the user to draw a line in the image indicating the edge that should be horizontally aligned. The difference between the angle of this line and the 0° horizontal is calculated and the image rotated accordingly. Note that it is important to keep the direction (start and end points) of the guiding line consistent. Use any two consistent landmarks across the dataset.
→ 02-interactive_rotation.py

16 Crop isolated and rotated images to their minimum possible size and combine the fore- and hindwings into one image.
→ 03-crop_combine.py

16.1 Create a mask from the contours detected in the image with the edge detection function cv2.Canny() by overlaying the edges with small circles of the same solid colour (grey). The size of the circles should be selected so that all edges within the desired foreground area connect into one shape. Fill all areas outside this shape with a solid colour.

16.2 Find all contours (read: continuous objects) within the mask using cv2.findContours(). There should be one contour in each image, since the wings were already isolated.

16.3 Find the minimum dimensions of the wing contours (cv2.boundingRect()), and crop the image to the corresponding coordinates.

16.4 Using the bounding rectangle dimensions, place the wings into a blank background with specified distance from the edges and each other. A standardised scale can be placed into the images.

17 If necessary, adjust the brightness of the images to the same level. Do this after removing the background and exclude black pixels from the calculation.
→ 04-adjust_brightness.py

Protocol references

- Bradski G. 2000. The OpenCV Library. Dr. Dobb's Journal of Software Tools 25(11), p. 120-123.
- Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, Wieser E, Taylor J, Berg S, Smith NJ, Kern R. 2020. Array programming with NumPy. Nature 585(7825), p. 357-362.
- Kawada R, Buffington ML (2016) A Scalable and Modular Dome Illumination System for Scientific Microphotography on a Budget. Esposito A (Ed.). PLOS ONE 11 (5): e0153426. https://doi.org/10.1371/journal.pone.0153426
- Pandas Development Team (McKinney et al.). 2020. pandas-dev/pandas: Pandas (v1.3.3). DOI 10.5281/zenodo.3509134
- Rosebrock A. 2015. Imutils: A series of OpenCV convenience functions. https://pyimagesearch.com/2015/02/02/just-open-sourced-personal-imutils-package-series-opencv-convenience-functions/ - https://github.com/PyImageSearch/imutils
- Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D, Burovski E, Peterson P, Weckesser W, Bright J, van der Walt SJ, Brett M, Wilson J, Millman, KJ, Mayorov N, Nelson ARJ, Jones E, Kern R, Larson E, Carey CJ, Polat İ, Yu F, Moore EW, VanderPlas J, Laxalde D, Perktold J, Cimrman R, Henrikson I, Quintero EA, Harris CR, Archibald AM, Ribeiro AH, Pedregosa F, van Mulbregt P, SciPy 1.0 Contributors. 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. Nature Methods, 17(3), 261-272.