

scAnt - an open-source platform for the creation of 3D models of arthropods and other small objects (#56600)

1

First submission

Guidance from your Editor

Please submit by **13 Jan 2021** for the benefit of the authors (and your \$200 publishing discount) .



Structure and Criteria

Please read the 'Structure and Criteria' page for general guidance.



Raw data check

Review the raw data.



Image check

Check that figures and images have not been inappropriately manipulated.

Privacy reminder: If uploading an annotated PDF, remove identifiable information to remain anonymous.

Files

Download and review all files from the [materials page](#).

9 Figure file(s)

6 Table file(s)



Structure and Criteria

Structure your review

The review form is divided into 5 sections. Please consider these when composing your review:

1. BASIC REPORTING
2. EXPERIMENTAL DESIGN
3. VALIDITY OF THE FINDINGS
4. General comments
5. Confidential notes to the editor

 You can also annotate this PDF and upload it as part of your review

When ready [submit online](#).

Editorial Criteria

Use these criteria points to structure your review. The full detailed editorial criteria is on your [guidance page](#).

BASIC REPORTING

-  Clear, unambiguous, professional English language used throughout.
-  Intro & background to show context. Literature well referenced & relevant.
-  Structure conforms to [Peerj standards](#), discipline norm, or improved for clarity.
-  Figures are relevant, high quality, well labelled & described.
-  Raw data supplied (see [Peerj policy](#)).

EXPERIMENTAL DESIGN

-  Original primary research within [Scope of the journal](#).
-  Research question well defined, relevant & meaningful. It is stated how the research fills an identified knowledge gap.
-  Rigorous investigation performed to a high technical & ethical standard.
-  Methods described with sufficient detail & information to replicate.

VALIDITY OF THE FINDINGS

-  Impact and novelty not assessed. Negative/inconclusive results accepted. *Meaningful* replication encouraged where rationale & benefit to literature is clearly stated.
-  All underlying data have been provided; they are robust, statistically sound, & controlled.
-  Speculation is welcome, but should be identified as such.
-  Conclusions are well stated, linked to original research question & limited to supporting results.

Standout reviewing tips

3



The best reviewers use these techniques

Tip

Support criticisms with evidence from the text or from other sources

Example

Smith et al (J of Methodology, 2005, V3, pp 123) have shown that the analysis you use in Lines 241-250 is not the most appropriate for this situation. Please explain why you used this method.

Give specific suggestions on how to improve the manuscript

Your introduction needs more detail. I suggest that you improve the description at lines 57- 86 to provide more justification for your study (specifically, you should expand upon the knowledge gap being filled).

Comment on language and grammar issues

The English language should be improved to ensure that an international audience can clearly understand your text. Some examples where the language could be improved include lines 23, 77, 121, 128 – the current phrasing makes comprehension difficult.

Organize by importance of the issues, and number your points

1. Your most important issue
2. The next most important item
3. ...
4. The least important points

Please provide constructive criticism, and avoid personal opinions

I thank you for providing the raw data, however your supplemental files need more descriptive metadata identifiers to be useful to future readers. Although your results are compelling, the data analysis should be improved in the following ways: AA, BB, CC

Comment on strengths (as well as weaknesses) of the manuscript

I commend the authors for their extensive data set, compiled over many years of detailed fieldwork. In addition, the manuscript is clearly written in professional, unambiguous language. If there is a weakness, it is in the statistical analysis (as I have noted above) which should be improved upon before Acceptance.

***scAnt* - an open-source platform for the creation of 3D models of arthropods and other small objects**

Fabian Plum¹, **David Labonte**^{Corresp. 1}

¹ Bioengineering, Imperial College London, London, United Kingdom

Corresponding Author: David Labonte
Email address: d.labonte@imperial.ac.uk

We present *scAnt*, an open-source platform for the creation of digital 3D models of arthropods and small objects. *scAnt* consists of a scanner and a Graphical User Interface (GUI) which enable the automated generation of multi-view Extended Depth Of Field (EDOF) images. These images are then masked with a novel approach combining random forest-based edge-detection, adaptive thresholding, and connected component labelling and can be processed further with a photogrammetry software package of choice, including open-source options such as Meshroom, to create high-quality, textured 3D models. We demonstrate how these 3D models can be “rigged” to enable realistic digital specimen posing, and introduce a novel simple yet effective method to include semi-realistic representations of approximately planar and transparent structures such as wings in 3D models. As a result of the exclusive reliance on generic hardware components, rapid prototyping, and open-source software, *scAnt* costs only a fraction of available comparable systems. The resulting accessibility of *scAnt* will (i) drive the development of novel and powerful methods for machine learning-driven behavioural studies, leveraging synthetic datasets; (ii) increase accuracy in comparative morphometric studies and extend the available parameter space with area and volume measurements; (iii) inspire novel forms of outreach; and (iv) aid in the digitisation efforts currently underway in several major natural history collections.

scAnt – An open-source platform for the creation of 3D models of Arthropods and other small objects

Fabian Plum¹, David Labonte¹

¹Department of Bioengineering, Imperial College London, London, United Kingdom

Abstract

We present *scAnt*, an open-source platform for the creation of digital 3D models of arthropods and small objects. *scAnt* consists of a scanner and a Graphical User Interface (GUI) which enable the automated generation of multi-view Extended Depth Of Field (EDOF) images. These images are then masked with a novel approach combining random forest-based edge-detection, adaptive thresholding, and connected component labelling and can be processed further with a photogrammetry software package of choice, including open-source options such as Meshroom, to create high-quality, textured 3D models. We demonstrate how these 3D models can be “rigged” to enable realistic digital specimen posing, and introduce a novel simple yet effective method to include semi-realistic representations of approximately planar and transparent structures such as wings in 3D models. As a result of the exclusive reliance on generic hardware components, rapid prototyping, and open-source software, *scAnt* costs only a fraction of available comparable systems. The resulting accessibility of *scAnt* will (i) drive the development of novel and powerful methods for machine learning-driven behavioral studies, leveraging synthetic datasets; (ii) increase accuracy in comparative morphometric studies and extend the available parameter space with area and volume measurements; (iii) inspire novel forms of outreach; and (iv) aid in the digitisation efforts currently underway in several major natural history collections.

Keywords

photogrammetry, morphometry, zoology, 3D digitisation, macro imaging

Introduction

The diversity of arthropods is unparalleled (Misof *et al.*, 2014). Key institutions such as the Natural History Museum in London, the Smithsonian National Museum of Natural History, or the Australian National Insect Collection house upwards of ten million insect specimens, and grow continuously, so archiving part of this diversity (Mantle, la Salle and Fisher, 2012). Clearly, the utility of these collections hinges on the accessibility of the specimens. However, specimen access typically requires to be either physically present on-site, or for specimens to be posted, so reducing the practical value of the collections. This issue is particularly severe for rare and valuable specimens such as holotypes, which can be difficult to access despite their scientific importance. In recognition of these limitations, significant efforts have been underway to digitise natural history collections (Beaman and Cellinese, 2012; Blagoderov *et al.*, 2012; Mantle, la Salle and Fisher, 2012; Nguyen *et al.*, 2017, 2014; Hudson *et al.*, 2015; Martins *et al.*, 2015;

Galantucci, Pesce and Lavecchia, 2016; Erolin, Jarron and Csetenyi, 2017; Ströbel *et al.*, 2018; Galantucci, Guerra and Lavecchia, 2018; Qian *et al.*, 2019; Brecko and Mathys, 2020). Such a “cybertaxonomy” has been predicted to revolutionise collaborative taxonomy, and fundamentally change formal and public taxonomic education (Zhang, Gao and Caelli, 2010; Wheeler *et al.*, 2012). However, the vast majority of these efforts have focused on high-throughput capturing of 2D images, and the convenient automatic inclusion of metadata such as barcodes, labels etc. Photographs are doubtlessly useful, but by definition contain substantially less information than the original 3D specimen, as they are restricted to a single image plane (Nguyen *et al.*, 2014; Ströbel *et al.*, 2018). Even obtaining simple 1D measurements from 2D images is error-prone, due to parallax errors and intra-observer variability, which is larger for measurements obtained from 2D photographs compared to 3D models (Ströbel *et al.*, 2018; Qian *et al.*, 2019; Brecko and Mathys, 2020). As a consequence, the “gold standard” for digitisation are photorealistic and anatomically accurate 3D models (Wheeler *et al.*, 2012).

Perhaps the most promising method for the creation of 3D models is photogrammetric reconstruction, which retains colour information, and represents an excellent compromise between portability, price and quality (Mathys, Brecko and Semal, 2013; Brecko and Mathys, 2020). To the best of our knowledge, there exist two photogrammetry devices specifically designed for the creation of 3D models of arthropods (Nguyen *et al.*, 2014; Ströbel *et al.*, 2018). However, both systems rely on specialised hardware and ~~crucially~~ commercial software for scanner control, image processing and 3D reconstruction, hampering widespread use ~~although the technology is in principle available~~.

Here, we address this limitation by introducing an open-source platform for the automated generation of 3D models of arthropods and small objects from a series of 2D images. This platform, *scAnt*, consists of (i) a low-cost scanner, built from generic structural and electronic components, (ii) an intuitive yet powerful Graphical User Interface (GUI) providing full control over (iii) a processing pipeline which combines several community-driven open-source applications to automate image capture and simultaneous image processing. *scAnt* runs on Windows and Linux operating systems, including full support for scanner control, image capture, image post-processing, additional scripts to facilitate reconstruction and is accompanied by a guide to manual mesh post-processing. All component drawings and assemblies are available on Thingiverse ([thingiverse.com/evobiomech/designs](https://www.thingiverse.com/evobiomech/designs), licensed under CC BY 4.0), and manuals and all code have been deposited on GitHub (github.com/evo-biomech/scAnt, made available under MIT License). Due to the combination of low-costs, open documentation, and system flexibility, *scAnt* enables scientists and dedicated amateurs alike to create high-quality 3D models of arthropods, ~~so~~ contributing to an increasing library of 3D models of arthropods which can be used in research, outreach, and conservation.

Material and Methods

scAnt is designed to automate the capturing and processing of 2D images from various viewing angles ~~as much as possible~~. It is built entirely with generic hardware components, and exclusively leverages recent developments in open-source image processing software, ~~so~~ remaining affordable, accessible, and flexible.

With *scAnt*, 3D models are produced through a series of five steps, described in detail below: (I) mounting a pinned specimen in an illumination dome; (II) configuring and conducting a “scan”, using a custom-made GUI; (III) processing captured images into Extended Depth Of Field (EDOF) images and creating masks for each EDOF image; (IV) generating textured 3D meshes from the EDOF images; and (V) post-processing of the created mesh.

Scanner design

In the design of *scAnt*, we built on and benefited from several insights and innovations from previous studies: (1) We deploy focus stacking to overcome the limitations of single-focal plane images (Gallo, Muzzupappa and Bruno, 2014); (2) We use a two-axis gimballed system to maximise the number of possible viewing angles (Nguyen *et al.*, 2014); (3) We use an illumination dome to achieve “flat” lighting, thus minimising specular reflections and other artefacts arising from variations in appearance with viewing angle (Ströbel *et al.*, 2018).

All structural components of the scanner are fabricated via 3D printing and laser cutting, methods readily available in most laboratories and museums, but also accessible to the keen amateur. Technical drawings of all components, the assembly, and a complete parts list are available for download (thingiverse.com/evobiomech/designs licensed under CC BY 4.0). The hardware costs for *scAnt* are approximately £200, and – due to the exclusive use of open-source software – the only remaining costs are related to the camera and lens (for our system, this added another £300). The total costs are hence between a factor of five to ten lower than for existing systems (~£4500 for C. V. Nguyen *et al.*, 2014; ~£7000 for Ströbel *et al.*, 2018).

In order to enhance hardware durability and to minimise print post-processing, we used PLA Filament for all prints. An acrylic sheet, laser-cut from 4 mm thick acrylic sheets, serves as the base plate for the mounting of all printed elements, and for the routing of all wires (Fig. 1 A).

The scanner consists of three main components (Fig. 1): (I) an illumination dome which ensures flat and diffuse lighting (inspired by the design in Ströbel *et al.*, 2018), (II) a two-axis gimbal to change the orientation of specimen inside the dome (inspired by the design of C. V. Nguyen *et al.*, 2014), and (III) an actuated camera slider to alter the position of the focal plane.

(I) The illumination dome is comprised of two symmetrical semi-domes. As the image background needs to be as uniform as possible to achieve high-quality results in subsequent masking and reconstruction steps (see below), the inside of both semi-domes was sanded down and coated in a matt light grey spray paint, so ensuring flat and diffuse lighting regardless of specimen orientation. The dome is illuminated by two arrays of LEDs. Further specifications, as well as links to the parts list and 3D models, can be found in the Supplementary Material. In order to reduce sharp specular reflections, translucent, 3D-printed rings are mounted in each half of the illumination dome and act as diffusers (Fig. 1 B).

(II) Inside the dome, the specimens are mounted directly on a rod connected to the two-axis gimbal. The horizontal (X) and vertical (Y) axes of the gimbal are actuated by two 4-lead NEMA

17 stepper motors (0.7 A, 1.8° step size). In order to minimise the noise introduced by motor jitter, a counterweight is attached to the top of the gimbal, opposite to the vertically oriented stepper motor actuating the Y-axis. The gimbal is mounted such that the X and Y axes run directly through the centre of the illumination dome, enabling specimen rotation without translation. Mirroring slits on the underside of the dome provide the space required for the upwards facing pin of the gimbal to move freely (**Fig. 1 B & C**).

(III) The gimbal enables rotation of the specimen about two axes but does not alter the distance between specimen and camera, as its axes run through the centre of the camera's field of view. To extend the effective depth of field for each XY position, the camera is mounted on a linear slider (Z axes, **Fig. 1 A**), so that the focal plane can be moved relative to the specimen. The stepper-controlled camera slider is mounted on top of an elevated platform, manufactured from 4 mm acrylic sheets via laser cutting, and is placed outside the illumination dome such that the lens points at the centre of the dome. We used a 20 MP colour sensor camera (BFS-U3-200S6C-C: 20 MP, FLIR Systems, Wilsonville, USA) with custom made extension tubes and a compact 35 mm, F16 MPZ Computar lens (Computar, CBC Group, Phoenix Arizona, USA). Other cameras and lenses may be used, but changing camera manufacturer in particular, will require some adjustments in the automated scripts controlling capturing and image processing (see below).

All communication between the computer and electronic components of the scanner occurs directly via USB to alleviate the need for additional control hardware. All three stepper motors are controlled by separate USB driver boards (Tic T500, Pololu Corporation, Las Vegas, USA). In order to minimise the number of cables and ports running to ~~or being used at~~ the computer, ~~respectively~~, all stepper drivers are connected to a USB Hub; a single 2.0 connection from the hub to the computer is then sufficient for all communication apart from camera control. The camera requires higher bandwidth and is therefore connected via a USB 3.0 / 3.1 port (**Fig. 1 A**).

All stepper drivers, as well as the LEDs, are connected to a generic DC12V, 5A power supply. Due to the low load required to actuate the gimbal, the 0.7 A, four-wire steppers do not draw a current close to the peak of either driver or power supply, so that the voltage remains approximately constant. In order to further reduce the peak current draw, steppers are always actuated successively rather than simultaneously.

The stepper drivers controlling the X- and Z-axes are connected to a limit switch, which provides a reference position, and prevents gimbal and camera from moving to positions where they may physically interact (**Fig. 1 A, C**). The system is constructed such that the object of interest can be scanned from any orientation about the Y-axis, and an angular range of 100° about the X-axis (**Fig. 1 C**). The range of motion in the X-axis is limited to ensure that the back half of the illumination dome is the background throughout the entire imaging range. While an angular range of 100° may appear limiting, steeper viewing angles are not required to generate high-quality models as long as the dorsoventral axis of the specimen is approximately aligned with the

Y-axis: multiple images obtained for increasingly dorsal or ventral views then contain redundant information, due to the lateral symmetry of insects in the sagittal plane.

The physical dimensions of the scanner impose an upper size limit of approximately 8 cm for the longest specimen axis, whereas the lower size limit depends on the desired resolution, and can be controlled via an appropriate choice of camera and lens. Imaging larger specimens requires a geometric scaling of the scanner, which is labour-intense, but still simple, as all software components and subsequent processing steps (see below) remain identical. Hence, *scAnt* can be readily adjusted to suit the specific needs of the end-user.

Mounting pinned specimens

Photogrammetric reconstruction requires that the appearance of the captured subject does not change with its viewing angle (Mathys, Brecko and Semal, 2013; Galantucci, Guerra and Lavecchia, 2018). As a consequence, lighting must be as uniform as possible, and more importantly any motion of parts of the imaged object relative to each other needs to be avoided. Practically, this means that insect specimens should be fully dehydrated prior to imaging

In preparing a specimen for scanning, we recommend posing it with its legs and antennae spread in the frontal plane to minimise occlusion (see Walker *et al.*, 1999 for an in-depth review of the handling and pinning of arthropods). Specimens need to be connected to a pin, which can be glued onto the rod connector of the gimbal; we recommend UV glue as it is easy to remove, but other solutions, such as attaching additional clamps to hold the pinned specimen in place, are possible (**Fig. 1 C**).

Automated image capture and processing using the *scAnt* GUI

In order to automate the scanning process as much as possible, we developed a graphical user interface (GUI) in python, which provides full control over all relevant settings. Key python libraries which power the GUI include NumPy (Oliphant, 2006), scikit-image (der Walt *et al.*, 2014), OpenCV (Bradski, 2000), and Pillow (Clark, 2015). The GUI consists of five sections (see **Fig. 2**): (I) The video stream of the camera is shown in the “live-view”-section of the GUI. Overexposed areas, as well as a normalised histogram for each RGB colour channel, can be displayed as an overlay to aid the choosing of suitable settings further. From within the (II) “camera-settings” section, all relevant camera parameters such as exposure time, gain level, and white balance ratios can be adjusted. In addition, a function has been implemented to determine initial exposure- and gain levels automatically. All parameters, including camera settings such as gain level and exposure time, stepper positions, and processing options, can be saved as presets to be reloaded for subsequent scans. (III) The “scanner-setup”-section is used to configure the project output location on the connected PC. Defined presets of all relevant scanning parameters can be saved to be re-used at a later stage if desired. The scan is also started or stopped from within this sub-window. Simultaneous stacking and masking of captured images during the scanning process can be enabled, which may reduce the total time required for a single scan, provided that sufficient computational power is available. Note well that due to the size of the uncompressed images, processing speed does not only hinge on CPU core count and clock speed,

but also on reading and writing speeds of the hard drive on which the output is stored. (IV) The “stepper control” section allows users to move the scanner to any position within the available range, to home or reset the three axes, and to define the number of images to be taken, determined by the minimum and maximum position for the three axes, and the corresponding step size. These parameters are generally a function of specimen size and may have to be adjusted accordingly. (V) The “info section” displays the progress of the scan, information about the status of the scanner, and a timestamp.

Image Processing

The images recorded during a scan are processed further in three successive steps: (i) images taken at identical specimen orientation, but different focal planes, are “stacked” to produce a single Extended Depth Of Field (EDOF) image per unique XY position; (ii) each EDOF image is masked to remove the image background; and (iii) meta-data required for 3D reconstruction, such as sensor size, focal length, and camera model, are written to the EDOF image files. All image processing steps are either implemented directly in python or called from within python, in case of pre-compiled software packages (details below). All processing scripts can be run from within the *scAnt* GUI, but are also available as standalone python files available on our [GitHub](#) page.

Generating Extended Depth Of Field images

As the depth of field is typically small compared to the dimensions of the specimen normal to the image plane, multiple images per unique XY position are required to capture every part of the body “in focus” (Fig. 3, I). These multiple images are then processed into a single EDOF image, using a series of processing steps (Fig. 3, II). First, images are aligned to correct for minor movements in the XY-plane of the camera rail. Second, the aligned images are blended, so that the resulting EDOF image only contains body pixels which are in focus. Image alignment is performed with *Hugin* (by Pablo d’Angelo, License: GNU GPLv2), an open-source toolbox for panorama photo stitching and High dynamic range (HDR) merging. High-quality merging requires to exclude all images which do not contain any relevant in-focus pixels. As it is not feasible to programme the scanner such that imaging is automatically constrained to relevant imaging planes, some images will fall into this category. For convenience, these images are automatically removed prior to the stacking process. The variance of a Laplacian 3 x 3 convolutional kernel is computed for each image, and images with a variance below a specified threshold are discarded. The optimal threshold depends on image noise, Z-Axis step size, aperture, and magnification, and hence has to be determined empirically; we provide a set of suitable parameter combinations as configuration presets (see [GitHub](#)). The remaining images of a stack are passed to *Hugin*’s `align_image_stack` function with a set of modifiers (see Tab. 1).

Aligned images are exported to a temporary folder and subsequently merged into a single EDOF image using *Enblend-Enfuse* (licensed under GNU GPL 2+), initially developed by Andrew Mihal and now maintained by the *Hugin* developer team. As for image alignment, we provide a pre-defined set of modifiers, chosen to represent a sensible compromise between quality and processing time. Most modifiers alter luminance control, owing to the history of *Enblend-*

Enfuse, which was developed to perform exposure correction for HDR image processing. A set of modifiers (see **Tab. 2**) are passed to Hugin’s enfuse function.

Image Masking

Once the EDOF images have been created, their background is removed (“masking”). Masking noticeably increases the quality of the mesh created during 3D reconstruction for at least three reasons. First, incorrectly matched background-features are excluded during the initial camera alignment. Second, the number of ‘floating artefacts’ is reduced. Third, the contours of the resulting model are retained more accurately. Masking is hence supported if not required by most of the tested software packages for 3D reconstruction. The masking process is conducted in a sequence of five steps (see **Fig. 3, III**): (1) Enhancing contours of the EDOF image, (2) approximating the specimen’s outline, (3) removing superfluous infill, (4) cleaning of the generated binary mask, and (5) applying it to the input image.

(1) The contours of each EDOF image are enhanced by increasing the local image contrast, using Contrast Limited Adaptive Histogram Equalisation (Zuiderveld, 1994). Subsequently, (2) contours within each EDOF image are identified using a pre-trained random forest-based edge detector (Dollar and Zitnick, 2013), which is implemented to extract the outline of the largest shape in the image – the specimen. In order to reduce the detector’s susceptibility to noise, Gaussian and median blurs are applied to each image prior to edge detection. Compared to a Sobel filter alone, the random forest edge detector returns more coherent outlines at comparable inference times, which is favourable when extracting the single largest shape in a given image (Bradski, 2000). The resulting outlines separate the specimen from its background, but they may include unwanted background areas, for example the area enclosed between a leg and the main body (**Fig. 3, III**). (3) These areas are removed with adaptive thresholding (akin to chroma- or luminance-keying to remove a specified colour region from an image), which works well as long as the lighting of the image centre is relatively uniform. (4) In order to identify both incorrectly retained or removed areas of a size below a threshold determined by the image resolution, the mask is then “cleaned”, using connected component-labelling (Fiorio and Gustedt, 1996; Wu, Otoo and Shoshani, 2005). Finally, the resulting mask of the EDOF image is exported as a binary *.png file, in which white areas represent parts of the specimen, and black areas represent the background. (5) Not all photogrammetry software natively supports the use of masks, and additional cut-outs can be generated by applying the binary mask to the input EDOF image, either as an alpha channel or simply by setting all background pixels to a value of zero. This process prevents feature extractors in subsequent steps from incorrectly detecting and matching landmarks of the background.

Adding metadata to processed images

In the last step, all relevant metadata is written into the generated EDOF image, using the open-source tool *ExifTool* (v. 12.00, developed by Phil Harvey, licenced under GPLv1+). Accurate 3D reconstruction relies on setup-specific image metadata, such as the camera’s sensor width and the focal length of the lens, used to “undistort” the EDOF images, match features between images, and approximate the dimensions of the scanned object. The parameters (see **Tab. 3**) are saved

automatically, and can readily be adjusted (the required parameters depend on the reconstruction software, see `config.yaml` in the Supplementary Material on [GitHub](#)).

Reconstruction

The result of a complete scan is a set of masked EDOF images from multiple orientations. These images now need to be combined to reconstruct a 3D model of the imaged specimen. Numerous photogrammetry software packages are available for this task, be they commercial (e.g. 3DSOM (CDSL Ltd., London United Kingdom), Metashape (Agisoft LLC, St. Petersburg, Russia), 3DF Zephyr (3DFLOW, Verona, Italy)), or open-source (e.g. Meshroom (AliceVision, 2018)). All these software packages have in common the stages in which a mesh is generated from the aligned images: (i) extraction of image features; (ii) matching extracted features between images; (iii) generating structure-from-motion (SfM), structure-from-silhouette (SfS), or a combination thereof; (iv) meshing and mesh-filtering; and (v) texturing. As a consequence, the procedure described below is, to a large extent, independent of the used software. SfM reconstructs 3D models by matching extracted features, or “descriptors”, between images; additional information such as data from accelerometers, gyroscopes, or GPS may aid in solving the “camera motion”. The resulting triangulated features, or “sparse point clouds”, form the basis for reconstructing the three-dimensional geometry of the captured scene, and are often combined with approximated depth maps (Seitz *et al.*, 2006; Jancosek and Pajdla, 2010). In contrast, SfS instead uses ~~silhouette, or “masked”~~ images to produce a visual hull (Laurentini, 1994; Jancosek and Pajdla, 2011; Nguyen *et al.*, 2014). SfM is particularly powerful when entire scenes are to be reconstructed, whereas SfS plays out its strength in the reconstruction of single objects in somewhat ‘sterile’ conditions. We choose SfM, as it outperforms SfS in retaining structural detail, such as concave shapes within the model that cannot be captured by image masks alone (Atsushi *et al.*, 2011; Nguyen *et al.*, 2014).

We did the bulk of our work with the open-source solution Meshroom 2020, to maximise the accessibility of *scAnt*, and hence focus the following description of the reconstruction process with Meshroom. In order to demonstrate both the competitiveness and limitations of open-source software, we reconstructed some models with the “lite” version of the commercial software 3DF Zephyr V5.009 (~£100).

In the first reconstruction step, images are aligned based on their visual content, which is represented by extracted feature vectors. In (i) feature extraction with descriptors such as SIFT (Lindeberg, 2012), ORB (Wang *et al.*, 2015), and AKAZE (Alcantarilla, Bartoli and Davison, 2012; Tareen and Saleem, 2018), there is a stereotypical trade-off between the number of descriptors (which determines the accuracy of the camera position matching and sampling density of triangulated features in subsequent steps) and the resulting processing times. Higher sampling densities consistently lead to higher model quality, and in particular, excel at retaining smaller structures such as antennae or thin leg segments. However, as long as each camera perspective is matched correctly in the subsequent step, the number of extracted features does not need to be excessively large. As the SfM approach assumes a constant scene and a moving camera, the background needs to be excluded using the generated masks. In Meshroom, image masks need to be applied directly to the input EDOF images prior to loading the “cut-outs” into

the software (**Fig. 3, IV**). A detailed description of the key parameters which require adjustment within Meshroom can be found on our [GitHub](#) page (see also Supplementary Material).

(ii) After aligning the images, they are matched, i.e. the feature vectors are compared to propose a set of likely neighbouring views. (iii) These view-pairs are then directly compared to reconstruct the location and rotation of each view iteratively and to triangulate extracted features – the essence of SfM reconstruction. The quality of the camera alignment is crucial for the final mesh quality but is strongly dependent on the number and quality of the previously extracted features. As a consequence, the reconstruction may fail, particularly when **scanning small objects** or objects which are visually similar across multiple viewing angles. In order to address this problem, we have provided a python script which computes an approximated structure-from-motion reconstruction of the camera positions to be used as a starting point. This script, `estimate_camera_positions.py`, takes the project configuration file generated during the scanning process as input and produces a `*.sfm` file with the solved camera intrinsics and transformation matrices for all EDOF images. This file can then be loaded into the open-source software Meshroom 2020 or later (**Fig. 4**).

Camera alignment may also be improved through the use of a textured reference object (Ströbel *et al.*, 2018). Although this approach can increase the quality of the alignment, the calibration is uniquely tied to the specific imaging positions used during the scan. In other words, the system would need to be recalibrated every time the step sizes are changed. In contrast, computing approximated positions from a new project file takes only a few seconds.

The result of the structure-from-motion reconstruction is a set of aligned camera positions, as well as a “sparse point cloud” of the scanned specimen. (iv) On the basis of this information, a depth map is then computed for each camera view, and a cohesive mesh is calculated. The meshing parameters are set to the highest levels of detail, favouring retention of small features over noise-reduction and smoothing. An in-depth explanation of the meshing process within Meshroom can be found in the literature Boykov and Kolmogorov, 2004; Labatut and Keriven, 2009; Jancosek and Pajdla, 2010, 2011, 2014. Several mesh-filtering steps, specific to Meshroom, are then applied to the generated mesh:

- **Mesh Filtering** is primarily used to remove unwanted elements of the resulting mesh by defining a size-threshold. Due to the masking process (see above), the background noise is minimal, so that the number of excluded elements is usually small.
- **Mesh Denoising** improves mesh topology and is best applied prior to **Mesh Decimation** or **Mesh Resampling**. Rather than reducing detail across the entire mesh, **Denoising** smoothens the input mesh across large surfaces, while leaving sharp edges intact, so decreasing noise without simplifying the mesh’s overall topology. Meshroom’s implementation of de-noising is based on the filtering methods described by Zhang *et al.*, 2018.

- **Mesh Resampling** can be used to reduce the number of vertices while retaining the overall shape, volume, and boundaries of the mesh. In contrast to **Mesh Decimation**, i.e. the deletion of redundant vertices, the mesh is rebuilt entirely using elements of predetermined dimensions. Generally, only one of the two methods should be used.

Last, (v) an image texture is generated by projecting the colour channel information of all undistorted camera views back onto the model. Depending on the size of the model and the number of faces, different unwrapping methods may be used to fit the colour information into a single image texture. We used the Least Squares Conformal Maps unwrapping method (Lévy *et al.*, 2002) to produce image textures of 4096 px * 4096 px for meshes with less than 600,000 faces. For larger meshes, we used mosaic texturing. In our trials, LSCM performed consistently better than mosaic texturing methods in retaining colour relationships between neighbouring regions. In addition, the resulting model textures can be edited more easily in subsequent manual post-processing steps.

We accurately replicated the lighting and camera setup of the scanner in Blender v2.8 to visually inspect the quality and accuracy of the final mesh. We also matched the position of exemplary EDOF and masked images, in order to demonstrate the absence of distortion and the level of detail retained both in topology and texture (see [Supplementary Material](#)).

Post-processing and Rigging

All manual post-processing was performed with Blender v2.8. First, the mesh of the mounting pin was removed, either by selecting the vertices connecting to the insect mesh and removing all connected faces, or by a simple Boolean intersection. Afterwards, the resulting hole is filled by collapsing the surrounding vertices to their centre. If desired, basic or Laplacian smoothing modifiers, as well as manual smoothing via sculpting, can be applied to the mesh to improve the local mesh quality (**Fig. 5, A**).

Thin and transparent structures, such as wings, are challenging for SfM-based reconstruction and often appear fragmented in the final reconstruction (**Fig. 5, B** and see Nguyen *et al.*, 2014; Ströbel *et al.*, 2018). However, structures such as wings may be approximately described as two-dimensional, provided that their thickness and curvature is small compared to the resolution of the scans. In such instances, there exists an easy yet high-quality option for reconstruction: wings can be added to the model during post-processing, by re-projecting the wing structure from EDOF images onto image planes, using the corresponding mask as an alpha-channel (**Fig. 5, B**). An orthogonal top-down view of the fragmented structure or sparse point cloud of the scanned specimen serves as the wing outline. The cut-out structure is then manually transformed in scale and rotation to match the original structure, using an image editing program such as GIMP (licensed under GPLv3+), and the scaled wing is then set as the image texture of a plane object with corresponding dimensions. By blending a diffuse shader for the colour information with a transparent shader for the opacity, the wings are added to the textured mesh, using the image plane's alpha channel as an input. Subsequently, all unconnected vertices of the fragmented wing are removed, using the process outlined above for the removal of the mounting pin. This process retains key visual information of the wing such as colour and venation patterns, but of course

does not correspond to a ‘true’ 3D reconstruction, as the wing is simplified as a single-plane 2D structure. This approximation may be valid in some cases (see **Fig. 6**, bottom row, *Orthomeria versicolor*), but will correspond to a significant simplification in others. For a better overview of the resulting model quality, refer to our [Sketchfab Model collection](#).

In a final step, the mesh may be “rigged”, by which we mean the assignment of virtual joints and rigid body parts, subsequently allowing users to pose the **animals** (**Fig. 5, C**); this process is possible in arthropods, because their bodies are reasonably approximated as a linked series of rigid bodies. As long as joint-location and type can be defined with reasonable accuracy, rigging allows for anatomically correct posing. In order to rig a scan, we create an “armature” inside the mesh and define joint types and locations. In order to ensure that labelled segments are treated as rigid bodies, we use weight painting, which avoids incorrect deformation commonly arising from smoothing algorithms which deform tissue locally.

 In order to demonstrate that “rigging” of 3D models is not only aesthetically pleasing, but also scientifically useful, we asked seven individuals of varying degree of experience (undergraduate students, PhD students, Postdocs) to measure the linear dimension of a set of three anatomical parameters of a specimen of *Sungaya inexpectata*: Head width (HW), femur length (FL, right front leg), and abdomen length (AL) (see also Ströbel *et al.*, 2018). These measurements were taken either on the physical specimen, using a LEICA Z6 Microscope (Leica Camera AG, Wetzlar Germany), or on the final 3D model using internal measuring tool in **blender** v2.8. In both cases, participants were allowed to rotate the specimen and choose their preferred magnification freely. We included HW and FL as they are relatively easy to measure and unlikely to be subject to large parallax error. As a consequence, we expect the difference between measurements from **2D** vs 3D data to be small. In contrast, the abdomen of the original specimen was curved in two axes (**Fig. 5, C**); we removed this curvature via rigging, straightening the abdomen to a position similar to that observed in live animals, so enabling us to test if the flexibility provided by rigging can help to increase measurement precision .

Accuracy evaluation

In order to quantify the reconstruction accuracy achievable with **scAnt**, we performed a set of measurements on certified gauge blocks, as described by Gallo, Muzzupappa and Bruno, 2014. Grade 0 gauge blocks (uncertainty of measurement within ± 0.00008 mm, UKAS certified) of 1.50 mm, 1.10 mm, 1.05 mm, and 1.00 mm were scanned in pairs in a 3D printed tray with a rectified surface to create step-sizes of 500 μ m, 100 μ m, and 50 μ m, respectively. The gauge cubes were then reconstructed using the parameters described in the Meshroom Guide on our [GitHub Page](#). The step-sizes were measured in Blender v2.8, using a custom-written python script. In brief, the vertical distance between every vertex of the top plane of the step cube (1.50 mm, 1.00 mm, 1.01 mm) was compared to the 1.00 mm reference gauge cube.

Scanned species

In order to demonstrate the versatility and quality of *scAnt*, we created 3D models for a series of arthropods, selected to present various challenges such as size, transparency, hairy surfaces, or iridescence (see **Tab. 4**).

Results and Discussion

We created several models to demonstrate the quality achievable with *scAnt* (see **Fig. 6** and [Sketchfab](#)). The error of the reconstructions, as estimated from the reconstruction of pairs of grade 0 gauge cubes, is approximately 12-15 μm across step sizes (see **Tab. 5**). In the following, we briefly address some key features of these models to demonstrate their quality, highlight key challenges and difficulties which users may encounter, and outline the current performance limits of *scAnt*. We then proceed to describe how 3D models generated with *scAnt* may be used in research, outreach, and conservation.

Quality, challenges, and limitations

The quality of the final models is determined by the quality of the EDOF images and the reconstruction process. The former hinges on the quality of the used camera and lens (to first-order; image processing during stacking introduces second-order effects on quality); high-quality cameras and lenses are available where the budget allows, and the resolution and scanner dimensions can easily be adjusted to specific needs. Hence, model quality is mostly limited by the quality of the 3D reconstruction, for which we identified four major challenges:

First, there exists a lower size limit for features that can be retained during reconstruction. The smallest features that can be retained depend on the resolution of the input images, the quality of the generated masks, the number of camera positions, the quality and breadth of meshing approaches of available in reconstruction software of choice, and – crucially - the size of the scanned animal (**Fig. 6, a**): The larger the animal, the coarser the absolute resolution. As an illustrative example, the thinnest preserved structures of the *Camponotus gigas* specimen with a body length of 27.5 mm are the claws, with a width of approximately 30 μm (they are likely even thinner towards their tips; **Fig. 6, b & c**). The choice of reconstruction software also notably contributes to the smallest retained features: Using the same input images, the low-cost commercial software 3DF Zephyr Lite generated a higher level of topology detail than the open-source option Meshroom (see **Fig. 6, f & g**), and thin structures such as tarsi were less likely to be fragmented. We provide a number of direct comparisons between the two software options on [Sketchfab](#).

The reconstruction of small features is particularly challenging where large size differences between feature and body size are combined with high local curvature, such as for the proboscis of *Leptoglossus zonatus* (**Fig. 6**, second row and see [Sketchfab](#) as well as Supplementary Material). The reconstructed proboscis was partially fragmented, which required manual post-processing ~~in the form of manually~~ joining the segments into a coherent mesh with Blender. Generally, structures with high-aspect ratios, such as hairs, may be visible in EDOF images, but fail to reconstruct. The most straightforward way to address this issue is to increase the number

of camera positions during the scan, which increases the number of matched descriptors, but comes at the cost of longer processing times. As SfM-based approaches require a sufficient number of clearly defined features to preserve a structure, increasing the contrast using additional filtering methods such as Wallis filters during the feature extraction stage may also help to address this issue (Gallo, Muzzupappa and Bruno, 2014). Alternative meshing methods, such as Visibility-Consistent Meshing, may retain thin features without the need for manual post-processing (see Ströbel *et al.*, 2018), but are currently only available as an experimental feature in commercial software such as of PhotoScan Pro (Agisoft LLC).

Second, highly reflective surfaces, such as the head capsule of the leaf-cutter ant, *Atta vollenweideri* (**Fig. 6**, first row), or of the cockroach, *Blatta orientalis* (**Fig. 6**, third row) may ~~ironically~~ lead to a comparably rough mesh topology, as the reflection introduces noise in the reconstruction process. This noise is caused by a combination of small variations in appearance with viewing angle and low surface detail, ~~so~~ reducing the number of features matched between views, and leading to incorrect depth maps. In contrast, the reconstructions of seemingly more challenging reflective and even iridescent structures with high levels of surface detail are far less noise ridden (**Fig. 6, d & e**). Noise may be reduced using Mesh de-noising, manual smoothing, or similar post-processing techniques, but the specimen may then appear artificially smooth, and it is difficult to achieve a photorealistic appearance. Two alternatives exist. First, the application of fine powder to the specimen's surface prior to scanning may reduce surface reflectivity, but it may come at the cost of a less realistic optical impression when the model is viewed in variable lighting conditions. Second, the uniformity of the lighting conditions may be improved further, but some reflections from the camera lens or the dome opening are impossible to avoid.

Third, transparent structures, such as some wings, can be difficult to reconstruct in 3D for at least three reasons. First, because they may not be recognised as part of the body, second, because their appearance varies strongly with viewing angle, and third, because pixels may be wrongly assigned to the dorsal or ventral side of the wing. Two solutions to this problem exist: First, the generated masked EDOF images can be used to re-project the fragmented structures with relatively little effort (**Fig. 5, B**). However, this approach is limited to thin and approximately planar wings, as it is unable to capture any out-of-plane variation in wing morphology. Second, alternative meshing methods such as SfS, or Visibility-Consistent Meshing can be used to enforce the inclusion of such regions, but are currently only implemented in commercial software, such as Agisoft PhotoScan Pro. To provide some perspective, the cost of this software exceeds the costs of the scanner, including camera and lens, by a factor of three.

Fourth, model creation is still time-intense. High-throughput model creation requires (i) scripting of the complete reconstruction process, which is possible with most photogrammetry software, including Meshroom; and (ii) a reduction in the total processing time. As an illustrative example, all models in **Fig. 6** were created with 180-360 EDOF images recorded at 20 MP resolution; Capturing and simultaneously processing the EDOF images using our GUI takes between 2 to 5 hours, depending on the number of camera positions (intel core i7, 8 Core processor at 4.3 GHz,

32 GB DDR4 RAM, NVIDIA RTX 2070 QMax). The reconstruction of 3D models took between ~~three to twelve~~ hours (intel core i9, 12 Core processor at 4.0 GHz, 64 GB DDR4 RAM, 2 x NVIDIA RTX 2080 Ti). Total model creation time may be reduced by sacrificing resolution (in step size or pixel density), by using more powerful computers, or by further improvements in the algorithms underlying the reconstruction.

Notwithstanding these challenges and limitations, the quality of the generated models is sufficient to enable their use in a number of different activities, including research, outreach, and conservation, as we briefly outline in the following.

Research

Morphometry

The development of increasingly sophisticated comparative and geometric morphometric methods has revived the popularity of morphometry in entomology (see e.g. Tatsuta, Takahashi and Sakamaki, 2018). The immediate appeal of the 3D models in morphometric studies is two-fold:

First, 3D models enable measurements which are difficult or even impossible to obtain from 2D images. Such measurements include area measurements of structures with complex shape or measurements of volumes. For example, it is straightforward to extract body surface area and body volume from our 3D models in blender v2.8, using its default measurement toolbox and 3D-Print addon (we assigned a uniform thickness to the wings, equivalent to the thickness of the reconstructed fragmented wing). An ordinary least squares regression of $\log_{10}$ -transformed data yields a slope of 0.61 (95% CI (0.33; 0.98)), consistent with isometry, and the results from Ströbel *et al.*, 2018 (data extracted with WebPlotDigitizer, slope 0.59, 95% CI (0.49, 0.68)). Notwithstanding the ease with which such data can be obtained from 3D models, two sources of error require attention. (i) The relative measurement error, defined as the ratio between the measured quantity and the resolution, is usually small for linear measurements, but it is additive, and can hence become sizeable for area and volume measurements, so reducing statistical power. In practice, this error is nevertheless only relevant for small structures, and is unlikely to bias scaling analyses conducted across multiple orders of magnitude, as there is no *a priori* reason to assume that it is systematic. (ii) Variations in fractal dimension across specimens may introduce a potential systematic error: area (and volume) measurements may rely on perimeter estimates, and these estimates can vary strongly with image resolution (Mandelbrot, 1967, 1982). Arthropods are unlikely to have a large fractal dimension, and occupy only a small range of physical length scales (from atomic dimensions to a few centimetres, say). However, body surfaces are often highly structured, so that a relationship between resolution and estimated area/volume is likely. Methods to estimate fractal dimension exist (e.g. Neal and Russ, 2012), and this issue needs further attention to enable a robust comparison of areas or volumes across scans or species.

Second, 3D models may increase measurement accuracy and precision even for measurements which can in principle be taken from 2D images (see, e.g. Ströbel *et al.*, 2018). Indeed, intra-observer errors can be large even for seemingly straightforward measurements (Viscardi, Sakamoto and Sigwart, 2010). We replicated the finding of Ströbel *et al.*, 2018 that the coefficient of variation may be reduced in some (but not all) cases when measurements are taken from 3D models instead of 2D images (see **Tab. 6**). We demonstrated further that the increase in precision is substantial for structures which are curved in the “original” but can be straightened via rigging in the 3D model (see **Tab. 6** and **Fig. 5, C**).

As a consequence of both advantages outlined above, morphometric data extracted from 3D models may be more versatile, accurate, increase statistical power, and reduce the ambiguity present in 2D images (Roth, 1993; Cardini, 2014; Gould, 2014; Fruciano, 2016; Buser, Sidlauskas and Summers, 2018; Bas and Smith, 2019 but see Courtenay *et al.*, 2018; McWhinnie and Parsons, 2019). While the use of 3D data is still somewhat limited by the power of available statistical methods for subsequent analyses (Polly and Motz, 2016), increasingly advanced methods are available (Bardua *et al.*, 2019), and 3D morphometry is expected to grow in importance.

Machine learning-based segmentation, detection, and tracking

Computer vision and machine learning are quickly finding their way into the standard methodological toolbox deployed in behavioural and kinematic studies of animals (Branson, 2014; Dell *et al.*, 2014; Egnor and Branson, 2016; Robie *et al.*, 2017). Recent advances in the use of pre-trained networks have reduced the quantity of required training data – and hence tedious hand-labelling – substantially (Mathis *et al.*, 2018; Redmon, Farhadi and Ap, 2018; Graving *et al.*, 2019; Pereira *et al.*, 2019; Bochkovskiy, Wang and Liao, 2020). We anticipate that 3D models which are photorealistic at least at low resolutions may further improve the generalisability and precision of these approaches, as they enable the creation of automatically labelled “synthetic datasets” (De Souza *et al.*, 2017; Tobin *et al.*, 2017; Varol *et al.*, 2017; Kar *et al.*, 2019; Mehta *et al.*, 2019; Stout *et al.*, 2019). Combining the power of free software engines such as *Unreal* or *Unity3D* with the flexibility of rigged 3D models, we can freely pose and position individuals and groups of individuals, control image background, lighting conditions, image noise, and degree of occlusion, so providing the opportunity to create a virtually unlimited variety of labelled training images. As illustrative examples, we provide some animations and still images of some of our models in the Supplementary Material.

Clearly, the effort required to generate synthetic data may outweigh the effort to create sufficient hand-labelled training data if all experiments occur in well-defined and controlled environments. However, we hope that the development of an integrated pipeline to create synthetic training data from 3D models may render network performance robust and accurate enough to be used in “unseen” (and unpredictable) conditions in the field. Such “synthetic datasets” may be a particularly powerful method to increase the performance of detection networks on complex

backgrounds, or in the presence of extreme occlusion/overlap between individuals, as occurs regularly in studies on social insects (Gal, Saragosti and Kronauer, 2020).

Conservation and outreach

High-quality 3D models of arthropods may aid in both conservation and outreach, as they bring with them the following advantages (Nguyen *et al.*, 2014; Galantucci, Pesce and Lavecchia, 2015; Erolin, Jarron and Csetenyi, 2017; Ströbel *et al.*, 2018; Cobb *et al.*, 2019; Brecko and Mathys, 2020):

First, digital models can readily be made available online, so reducing the need for physical visits to collections or specimen exchange between collections. Therefore, they maximise collection utility and accessibility, and minimise the risk of specimen damage. Second, in contrast to physical specimens, digital models do not deteriorate. Both points may be particularly relevant for the study and availability of valuable holotypes, provided that the digital models are of sufficient accuracy. Third, 3D models add significant value to online encyclopaedias which currently are almost exclusively populated by 2D images. Fourth, 3D models can be animated, so increasing the information content which can be stored and communicated. For example, we are working on an extension of our pipeline to animate our models directly with kinematic data recorded with live animals during natural locomotion. Such animations may be used in teaching, for example to demonstrate different gaits, but also in computer games or educational TV programmes. Fifth, 3D models may be used in “digital exhibitions”, enabling an unprecedented possibility of visitors to interact with the exhibits, and increasing accessibility of rare specimen.

Conclusion

We introduced *scAnt* 3D, a low-cost, open-source photogrammetry pipeline to create digital 3D models of arthropods and small objects. The process of 3D model creation is largely automated, and can be easily controlled via a user-friendly GUI; all required code, technical drawings and component lists are freely available. We achieve state-of-the-art model quality at a fraction of the cost of comparable systems, so paving the way for the widespread creation of near photorealistic 3D models of arthropods to be used in research, conservation, and outreach.

Acknowledgements

The authors are grateful for the kind support and expert input from René Bulla, in particular in respect to 3D modelling and processing of visual data.

Funding

This study was funded by the Imperial College’s President’s PhD Scholarship (to FP) and is part of a project that has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (Grant agreement No. 851705, to DL).

618 Supplementary Material

- 619 • [GitHub Page](#) including installation, setup, and scanning instructions
- 620 • Links to all software / used versions (listed on [GitHub Page](#))
- 621 • Example files ([Sketchfab](#) links listed with [Scanned species](#))
- 622 • [Thingiverse](#) printing files and construction instructions

References

- Alcantarilla, P. F., Bartoli, A. and Davison, A. J. (2012) ‘KAZE Features’, in Fitzgibbon, A. et al. (eds) *Computer Vision -- ECCV 2012*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 214–227.
- AliceVision (2018) ‘Meshroom: A 3D reconstruction software.’ Available at: <https://github.com/alicevision/meshroom>.
- Atsushi, K. et al. (2011) ‘System for reconstruction of three-dimensional micro objects from multiple photographic images’, *CAD Computer Aided Design*. Elsevier Ltd, 43(8), pp. 1045–1055. doi: 10.1016/j.cad.2011.01.019.
- Bardua, C. et al. (2019) ‘A Practical Guide to Sliding and Surface Semilandmarks in Morphometric Analyses’, *Integrative Organismal Biology*, 1(1). doi: 10.1093/iob/obz016.
- Bas, A. and Smith, W. A. P. (2019) ‘What Does 2D Geometric Information Really Tell Us About 3D Face Shape?’, *International Journal of Computer Vision*. Springer US, 127(10), pp. 1455–1473. doi: 10.1007/s11263-019-01197-x.
- Beaman, R. S. and Cellinese, N. (2012) ‘Mass digitization of scientific collections: New opportunities to transform the use of biological specimens and underwrite biodiversity science’, *ZooKeys*, 209, pp. 7–17. doi: 10.3897/zookeys.209.3313.
- Blagoderov, V. et al. (2012) ‘No specimen left behind: Industrial scale digitization of natural history collections’, *ZooKeys*, 209, pp. 133–146. doi: 10.3897/zookeys.209.3178.
- Bochkovskiy, A., Wang, C.-Y. and Liao, H.-Y. M. (2020) ‘YOLOv4: Optimal Speed and Accuracy of Object Detection’.
- Boykov, Y. and Kolmogorov, V. (2004) ‘An Experimental Comparison of Min-Cut / Max-Flow Algorithms for Energy Minimization in Vision’, 26(9), pp. 1124–1137.
- Bradski, G. (2000) ‘The OpenCV Library’, *Dr. Dobb's Journal of Software Tools*.
- Branson, K. (2014) ‘Distinguishing seemingly indistinguishable animals with computer vision’, *Nature Publishing Group*. Nature Publishing Group, 11(7), pp. 721–722. doi: 10.1038/nmeth.3004.
- Brecko, J. and Mathys, A. (2020) ‘Handbook of best practice and standards for 2D+ and 3D imaging of natural history collections’, *European Journal of Taxonomy*, 2020(623), pp. 1–115. doi: 10.5852/ejt.2020.623.
- Buser, T. J., Sidlauskas, B. L. and Summers, A. P. (2018) ‘2D or Not 2D? Testing the Utility of 2D Vs. 3D Landmark Data in Geometric Morphometrics of the Sculpin Subfamily Oligocottinae (Pisces; Cottoidea)’, *Anatomical Record*, 301(5), pp. 806–818. doi: 10.1002/ar.23752.
- Cardini, A. (2014) ‘Missing the third dimension in geometric morphometrics: How to assess if 2D images really are a good proxy for 3D structures?’, *Hystrix*, 25(2), pp. 73–81. doi: 10.4404/hystrix-25.2-10993.
- Clark, A. (2015) ‘Pillow (PIL Fork) Documentation’. readthedocs. Available at: <https://buildmedia.readthedocs.org/media/pdf/pillow/latest/pillow.pdf>.
- Cobb, N. S. et al. (2019) ‘Assessment of North American arthropod collections: Prospects and challenges for addressing biodiversity research’, *PeerJ*, 2019(11), pp. 1–29. doi: 10.7717/peerj.8086.
- Courtenay, L. A. et al. (2018) ‘Testing accuracy in 2D and 3D geometric morphometric methods for cut mark identification and classification’, *PeerJ*, 2018(7), pp. 1–24. doi: 10.7717/peerj.5133.
- Dell, A. I. et al. (2014) ‘Automated image-based tracking and its application in ecology’, 29(7), pp. 417–428. doi: 10.1016/j.tree.2014.05.004.
- Dollar, P. and Zitnick, C. L. (2013) ‘Structured forests for fast edge detection’, *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1841–1848. doi: 10.1109/ICCV.2013.231.
- Egnor, S. E. R. and Branson, K. (2016) ‘Computational Analysis of Behavior’, pp. 217–238. doi: 10.1146/annurev-

- 665 neuro-070815-013845.
- 666 Erolin, C., Jarron, M. and Csetenyi, L. J. (2017) 'Zoology 3D: Creating a digital collection of specimens from the
667 D'Arcy Thompson Zoology Museum', *Digital Applications in Archaeology and Cultural Heritage*. Elsevier Ltd,
668 7(September), pp. 51–55. doi: 10.1016/j.daach.2017.11.002.
- 669 Feltz, C. J. and Miller, G. E. (1996) 'An asymptotic test for the equality of coefficients of variation from k
670 populations', *Statistics in Medicine*, 15(6), pp. 647–658. doi: 10.1002/(SICI)1097-0258(19960330)15:6<647::AID-
671 SIM184>3.0.CO;2-P.
- 672 Fiorio, C. and Gustedt, J. (1996) 'Two linear time Union-Find strategies for image processing', *Theoretical
673 Computer Science*, 154.
- 674 Fruciano, C. (2016) 'Measurement error in geometric morphometrics', *Development Genes and Evolution*, 226. doi:
675 10.1007/s00427-016-0537-4.
- 676 Gal, A., Saragosti, J. and Kronauer, D. J. (2020) 'anTraX, a software package for high-throughput video tracking of
677 color-tagged insects', *eLife*, 9. doi: 10.7554/eLife.58145.
- 678 Galantucci, L. M., Guerra, M. G. and Lavecchia, F. (2018) 'Photogrammetry Applied to Small and Micro Scaled
679 Objects: A Review', *Lecture Notes in Mechanical Engineering*. Springer International Publishing, (978-3-319-
680 89563-5), pp. 57–77. doi: 10.1007/978-3-319-89563-5.
- 681 Galantucci, L. M., Pesce, M. and Lavecchia, F. (2015) 'A stereo photogrammetry scanning methodology, for precise
682 and accurate 3D digitization of small parts with sub-millimeter sized features', *CIRP Annals - Manufacturing
683 Technology*. CIRP, 64(1), pp. 507–510. doi: 10.1016/j.cirp.2015.04.016.
- 684 Galantucci, L. M., Pesce, M. and Lavecchia, F. (2016) 'A powerful scanning methodology for 3D measurements of
685 small parts with complex surfaces and sub millimeter-sized features, based on close range photogrammetry',
686 *Precision Engineering*. Elsevier Inc., 43, pp. 211–219. doi: 10.1016/j.precisioneng.2015.07.010.
- 687 Gallo, A., Muzzupappa, M. and Bruno, F. (2014) '3D reconstruction of small sized objects from a sequence of
688 multi-focused images', *Journal of Cultural Heritage*. Elsevier Masson SAS, 15(2), pp. 173–182. doi:
689 10.1016/j.culher.2013.04.009.
- 690 Gould, F. D. H. (2014) 'To 3D or not to 3D, that is the question: Do 3D surface analyses improve the
691 ecomorphological power of the distal femur in placental mammals?', *PLoS ONE*, 9(3). doi:
692 10.1371/journal.pone.0091719.
- 693 Graving, J. M. *et al.* (2019) 'Fast and robust animal pose estimation', *bioRxiv*, p. 620245. doi: 10.1101/620245.
- 694 Hudson, L. N. *et al.* (2015) 'Insect: Automating the digitization of natural history collections', *PLoS ONE*, 10(11),
695 pp. 1–15. doi: 10.1371/journal.pone.0143402.
- 696 Jancosek, M. and Pajdla, T. (2010) 'Hallucination-free Multi-View Stereo'.
- 697 Jancosek, M. and Pajdla, T. (2011) 'Multi-view reconstruction preserving weakly-supported surfaces', *Proceedings
698 of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 3121–3128. doi:
699 10.1109/CVPR.2011.5995693.
- 700 Jancosek, M. and Pajdla, T. (2014) 'Exploiting Visibility Information in Surface Reconstruction to Preserve Weakly
701 Supported Surfaces', *International Scholarly Research Notices*, 2014, pp. 1–20. doi: 10.1155/2014/798595.
- 702 Kar, A. *et al.* (2019) 'Meta-sim: Learning to generate synthetic datasets', *Proceedings of the IEEE International
703 Conference on Computer Vision*, 2019-Octob, pp. 4550–4559. doi: 10.1109/ICCV.2019.00465.
- 704 Labatut, P. and Keriven, R. (2009) 'Robust and efficient surface reconstruction from range data', 0(0), pp. 1–15.
- 705 Laurentini, A. (1994) 'The Visual Hull Concept for Silhouette-Based Image Understanding', *IEEE Transactions on
706 Pattern Analysis and Machine Intelligence*, 16(2), pp. 150–162. doi: 10.1109/34.273735.
- 707 Lévy, B. *et al.* (2002) 'Least squares conformal maps for automatic texture atlas generation', *ACM Transactions on*

- 708 *Graphics*, 21(3), pp. 362–371. doi: 10.1145/566654.566590.
- 709 Lindeberg, T. (2012) ‘Scale Invariant Feature Transform’, *Scholarpedia*, 7(5), p. 10491. doi:
710 10.4249/scholarpedia.10491.
- 711 Mandelbrot, B. (1967) ‘How Long Is the Coast of Britain? Statistical Self-Similarity and Fractional Dimension’,
712 *Science*. American Association for the Advancement of Science, 156(3775), pp. 636–638. doi:
713 10.1126/science.156.3775.636.
- 714 Mandelbrot, B. . . (1982) ‘The Fractal Geometry of Nature’, *Berichte der Bunsengesellschaft für physikalische*
715 *Chemie*, 89(2), p. 209. doi: <https://doi.org/10.1002/bbpc.19850890223>.
- 716 Mantle, B. L., la Salle, J. and Fisher, N. (2012) ‘Whole-drawer imaging for digital management and curation of a
717 large entomological collection’, *ZooKeys*, 209, pp. 147–163. doi: 10.3897/zookeys.209.3169.
- 718 Martins, A. F. *et al.* (2015) ‘R2OBIE-3D, a fast robotic high-resolution system for quantitative phenotyping of
719 surface geometry and colour-texture’, *PLoS ONE*, 10(6), pp. 1–18. doi: 10.1371/journal.pone.0126740.
- 720 Marwick, B. and Krishnamoorthy, K. (2019) ‘cvequality: Tests for the Equality of Coefficients of Variation from
721 Multiple Groups’. Available at: <https://github.com/benmarwick/cvequality>.
- 722 Mathis, A. *et al.* (2018) ‘DeepLabCut: markerless pose estimation of user-defined body parts with deep learning’,
723 *Nature Neuroscience*, 21(9), pp. 1281–1289. doi: 10.1038/s41593-018-0209-y.
- 724 Mathys, A., Brecko, J. and Semal, P. (2013) ‘Comparing 3D digitizing technologies: What are the differences?’,
725 *Proceedings of the DigitalHeritage 2013 - Federating the 19th Int’l VSMM, 10th Eurographics GCH, and 2nd*
726 *UNESCO Memory of the World Conferences, Plus Special Sessions fromCAA, Arqueologica 2.0 et al.* IEEE,
727 1(Table 1), pp. 201–204. doi: 10.1109/DigitalHeritage.2013.6743733.
- 728 McWhinnie, K. C. and Parsons, K. J. (2019) ‘Shaping up? A direct comparison between 2D and low-cost 3D shape
729 analysis using African cichlid mandibles’, *Environmental Biology of Fishes*. Environmental Biology of Fishes,
730 102(7), pp. 927–938. doi: 10.1007/s10641-019-00879-2.
- 731 Mehta, B. *et al.* (2019) ‘Active Domain Randomization’, pp. 1–15.
- 732 Misof, B. *et al.* (2014) ‘Phylogenomics resolves the timing and pattern of insect evolution’, *Science*, 346(6210), pp.
733 763–767. Available at: <http://www.sciencemag.org/content/346/6210/763.abstract>.
- 734 Neal, B. and Russ, J. C. (2012) *Measuring Shape*. 1st edn. Boca Ranton: CRC Press.
- 735 Nguyen, C. *et al.* (2017) ‘Towards high-throughput 3d insect capture for species discovery and diagnostics’,
736 *Proceedings - 13th IEEE International Conference on eScience, eScience 2017*, pp. 559–560. doi:
737 10.1109/eScience.2017.90.
- 738 Nguyen, C. V. *et al.* (2014) ‘Capturing natural-colour 3D models of insects for species discovery and diagnostics’,
739 *PLoS ONE*, 9(4), pp. 1–11. doi: 10.1371/journal.pone.0094346.
- 740 Oliphant, T. E. (2006) *A guide to NumPy*. Trelgol Publishing USA.
- 741 Pereira, T. D. *et al.* (2019) ‘Fast animal pose estimation using deep neural networks’, *Nature Methods*. Springer US,
742 16(1), pp. 117–125. doi: 10.1038/s41592-018-0234-5.
- 743 Polly, P. D. and Motz, G. J. (2016) ‘Patterns and processes in morphospace: Geometric morphometrics of three-
744 dimensional objects’, *The Paleontological Society Papers*. 2017/04/27. Cambridge University Press, 22, pp. 71–99.
745 doi: DOI: 10.1017/scs.2017.9.
- 746 Qian, J. *et al.* (2019) ‘Large-scale 3D imaging of insects with natural color’, *Optics Express*, 27(4), p. 4845. doi:
747 10.1364/oe.27.004845.
- 748 Redmon, J., Farhadi, A. and Ap, C. (2018) ‘YOLOv3: An Incremental Improvement’, *CoRR*, abs/1804.0. Available
749 at: <http://arxiv.org/abs/1804.02767>.

- Robie, A. A. *et al.* (2017) ‘Machine vision methods for analyzing social interactions’, pp. 25–34. doi: 10.1242/jeb.142281.
- Roth, V. (1993) ‘On three-dimensional morphometrics, and on the identification of landmark points’, in Marcus, L., Bello, E., and Carcia-Valdecasas, A. (eds) *Contributions to Morphometrics*, pp. 41–46.
- Seitz, S. M. *et al.* (2006) ‘A comparison and evaluation of multi-view stereo reconstruction algorithms’, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1, pp. 519–526. doi: 10.1109/CVPR.2006.19.
- De Souza, C. R. *et al.* (2017) ‘Procedural generation of videos to train deep action recognition networks’, *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017-Janua, pp. 2594–2604. doi: 10.1109/CVPR.2017.278.
- Stout, A. *et al.* (2019) ‘The development of synthetic thermal image generation tools and training data at FLIR’, 1098814(May), p. 38. doi: 10.1117/12.2518655.
- Ströbel, B. *et al.* (2018) ‘An automated device for the digitization and 3D modelling of insects, combining extended-depth-of-field and all-side multi-view imaging’, *ZooKeys*, 2018(759), pp. 1–27. doi: 10.3897/zookeys.759.24584.
- Tareen, S. A. K. and Saleem, Z. (2018) ‘A comparative analysis of SIFT, SURF, KAZE, AKAZE, ORB, and BRISK’, *2018 International Conference on Computing, Mathematics and Engineering Technologies: Invent, Innovate and Integrate for Socioeconomic Development, iCoMET 2018 - Proceedings*. IEEE, 2018-Janua, pp. 1–10. doi: 10.1109/ICOMET.2018.8346440.
- Tatsuta, H., Takahashi, K. H. and Sakamaki, Y. (2018) ‘Geometric morphometrics in entomology: Basics and applications’, *Entomological Science*, 21(2), pp. 164–184. doi: 10.1111/ens.12293.
- Tobin, J. *et al.* (2017) ‘Domain randomization for transferring deep neural networks from simulation to the real world’, *IEEE International Conference on Intelligent Robots and Systems*, 2017-Septe, pp. 23–30. doi: 10.1109/IROS.2017.8202133.
- Varol, G. *et al.* (2017) ‘Learning from synthetic humans’, *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017-Janua, pp. 4627–4635. doi: 10.1109/CVPR.2017.492.
- Viscardi, P., Sakamoto, M. and Sigwart, J. D. (2010) ‘How long is a piece of Strix? Methods in measuring and measuring the measurers’, *Zoomorphology*, 129(3), pp. 185–194. doi: 10.1007/s00435-010-0111-y.
- Walker, A. K. *et al.* (1999) *Insects and Other Invertebrates*. Oxford: Butterworth Heinemann.
- der Walt, S. *et al.* (2014) ‘scikit-image: image processing in Python’, *PeerJ*. PeerJ Inc., 2, p. e453.
- Wang, R. *et al.* (2015) ‘License plate localization in complex scenes based on oriented FAST and rotated BRIEF feature’, *Journal of Electronic Imaging*. SPIE, 24(5), pp. 1–10. doi: 10.1117/1.JEI.24.5.053011.
- Wheeler, Q. *et al.* (2012) ‘Nomenclatural benchmarking: The roles of digital typification and telemicroscopy’, *ZooKeys*, 209, pp. 193–202. doi: 10.3897/zookeys.209.3486.
- Wu, K., Otoo, E. and Shoshani, A. (2005) ‘Optimizing connected component labeling algorithms’, *Medical Imaging 2005: Image Processing*, 5747, p. 1965. doi: 10.1117/12.596105.
- Zhang, J. *et al.* (2019) ‘Static/Dynamic Filtering for Mesh Geometry’, *IEEE Transactions on Visualization and Computer Graphics*, 25(4), pp. 1774–1787. doi: 10.1109/TVCG.2018.2816926.
- Zhang, X., Gao, Y. and Caelli, T. (2010) ‘Primitive-based 3D structure inference from a single 2D image for insect modeling: Towards an electronic field guide for insect identification’, *11th International Conference on Control, Automation, Robotics and Vision, ICARCV 2010*, (December), pp. 866–871. doi: 10.1109/ICARCV.2010.5707814.
- Zuiderveld, K. (1994) ‘Contrast Limited Adaptive Histogram Equalization’, *Graphics Gems*. Academic Press, pp. 474–485. Available at: <https://ci.nii.ac.jp/naid/10031105927/en/>.

Figure 1

CAD Drawings and a photograph of the assembled 3D scanner.

(A) Top-down drawing of the assembled scanner, including wiring. The stepper motors for each axis as well as their corresponding stepper drivers are labelled **X** (horizontal, red wires), **Y** (vertical, orange wires), and **Z** (camera slider, yellow wires) respectively. Two end-stops (limit switches) are attached to the actuated camera rail to provide reference positions for the **X** and **Z** axes. Navy blue and maroon wires indicate connections from the 12V power supply to the stepper drivers and LEDs, respectively. Blue wires indicate USB cables, connecting stepper drivers and camera to the computer, using USB 2.0 and USB 3.0 ports, respectively. (B) Photograph of the open illumination dome, comprised of two half-domes, with a specimen mounted inside (*Sympetrum striolatum*). The LEDs are mounted on the inside of the half-domes and are covered by 3D printed diffusers. The slits at the bottom of each semi-sphere allow the pin of the X-axes stepper to move, as indicated in (C). (C) Side-view drawing of the assembled scanner, illustrating the range of motion of the **X**, **Y**, and **Z** axes. The **Y**-axis stepper is unlimited. Additional photographs of the assembled scanner, as well as links to the parts lists and 3D models, can be found in the supplementary material.

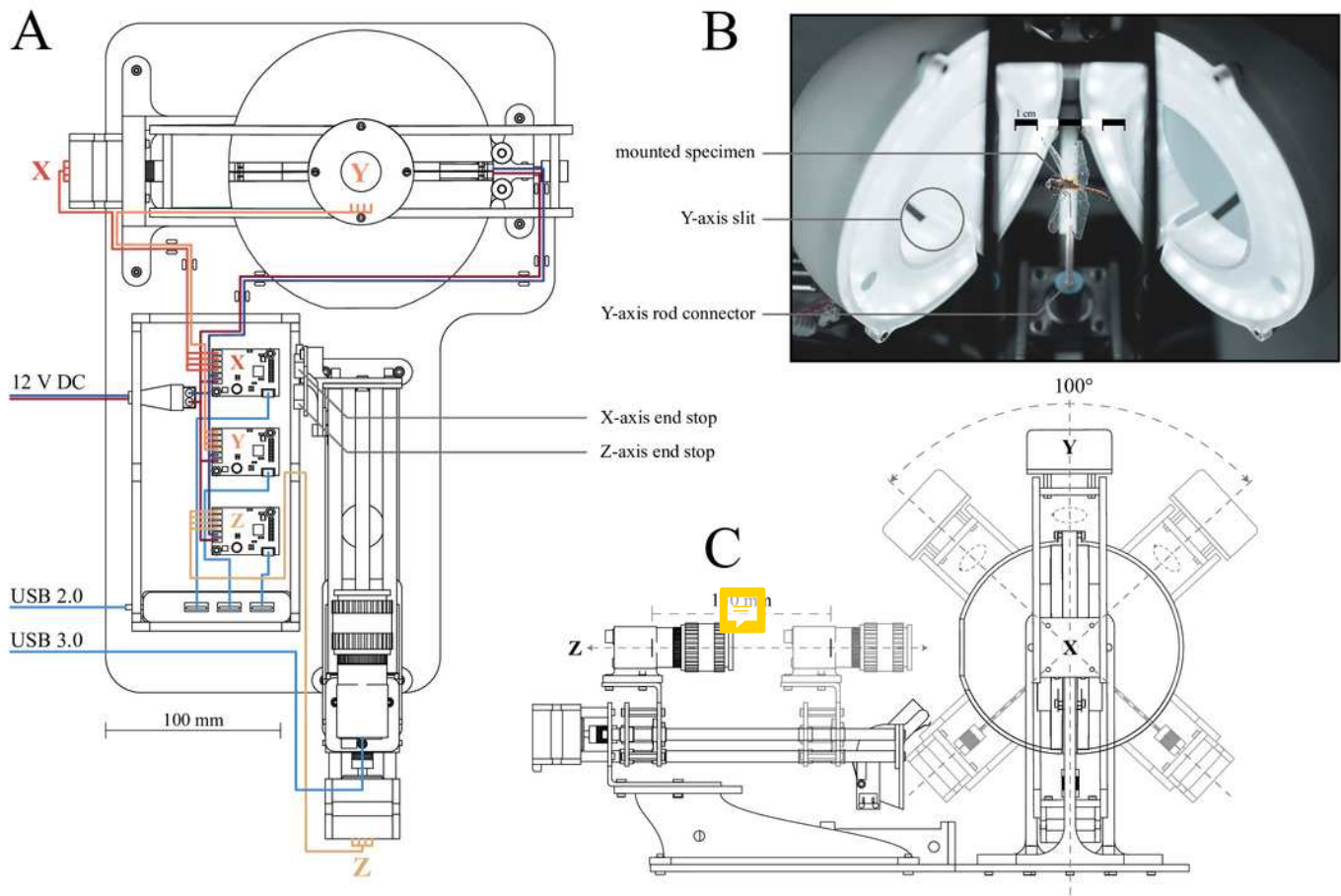


Figure 2

Graphical user interface of *scAnt*.

The **Live View (maroon)** displays the camera video stream. Bright-red areas indicate over-exposed pixels within the image to aid in the adjustment of the camera settings; a histogram of the RGB colour channels is displayed in the lower right corner. All relevant **Camera Settings (red)** such as exposure time and gain can be adjusted, the live view can be started, and images can be captured. The **Scanner Setup-section (yellow)** allows users to define an output folder, load existing project configurations ("presets"), write the scanner configuration to the specified output folder, choose stacking and masking methods, and start/abort the scanning process. The **Stepper Control-section (navy)** is used to change and **config.** the positions of the camera and mounted specimen during live view ~~and to define limits for the scanning process, respectively.~~ The X- and Z-axes (see **Fig. 1 A & C**) of the scanner can be returned to their home positions, and the zero-position of the Y-axis can be reset. The **Info-section (blue)** displays the progress of the current scan and all events logged by the scanner in chronological order.

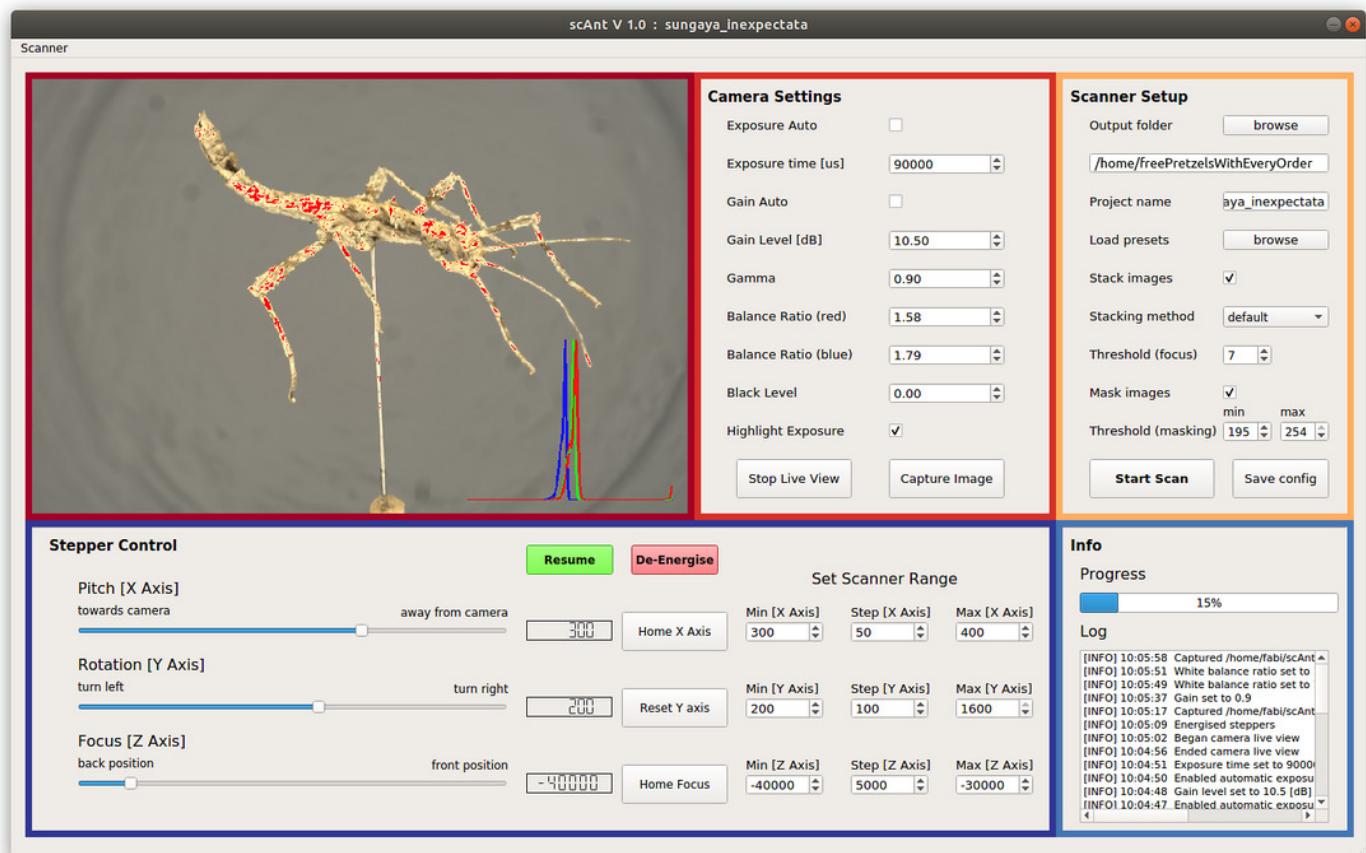


Figure 3

Generating EDOF images and masks.

(I) Images of different focal planes are automatically captured by moving the camera along the Z axes for each unique XY position (“viewing angle”). Between 20 and 50 images for each viewing angle are usually sufficient to cover the extent of a specimen of typical size normal to the imaging plane. The *Leptoglossus zonatus* specimen shown in I is captured at a step size of 18° about the Y-axis, but finer spatial resolutions can be configured from within the GUI. (II) The images of each stack are then aligned, and focus masks are generated to combine the sharpest regions of each image into a single stacked EDOF image. (III) These EDOF images are then “masked” to remove the image background. Noise is removed by applying a 5x5 Gaussian kernel, and edges are enhanced using contrast limited adaptive histogram equalisation (Zuiderveld, 1994). Subsequently, the largest contour within the image is extracted using a random-forest-based edge detector (Dollár and Zitnick, 2013). The resulting outline may include unwanted areas (highlighted in red) which are removed using adaptive thresholding. The noise introduced by the thresholding is reduced with connected component labelling (Fiorio and Gustedt, 1996; Wu, Otoo and Shoshani, 2005). (IV) The generated mask is then applied to the EDOF image to create a single cut-out image per XY-orientation.

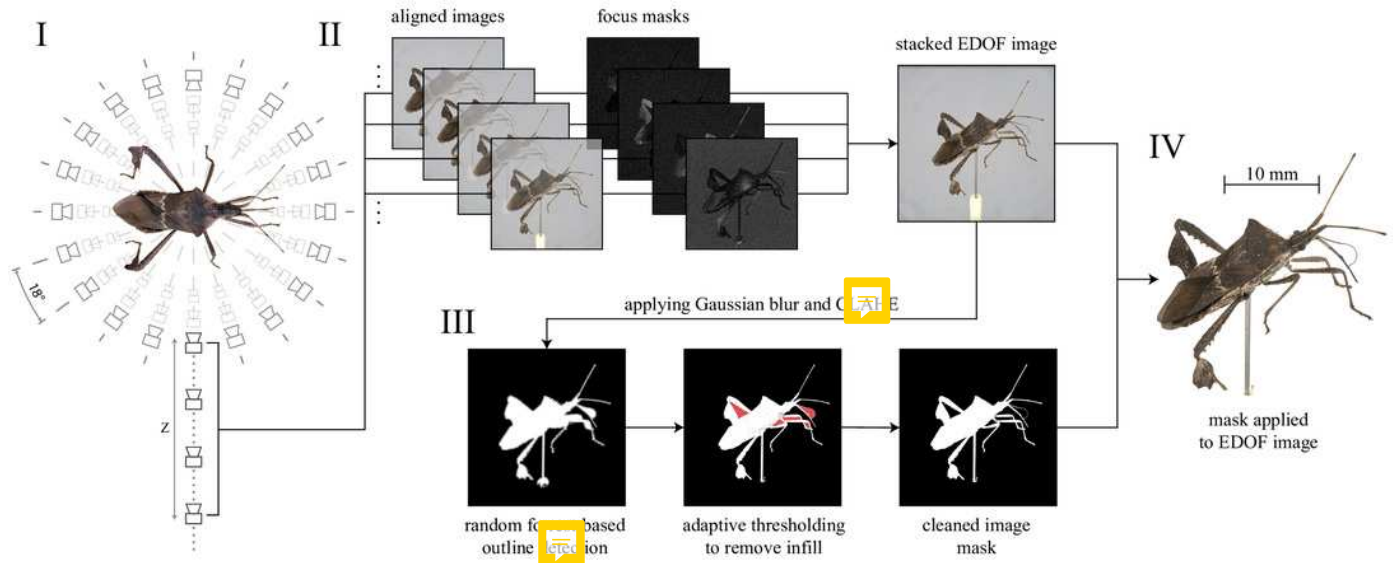


Figure 4

Camera positions of a completed scan.

(I) Approximated camera positions computed from the settings provided by the project configuration file and the scanner dimensions. A transformation matrix for each camera is calculated and displayed as a vector. (II) Solved camera positions of the structure-from-motion (SfM) within Meshroom based on matched features between undistorted EDof images.

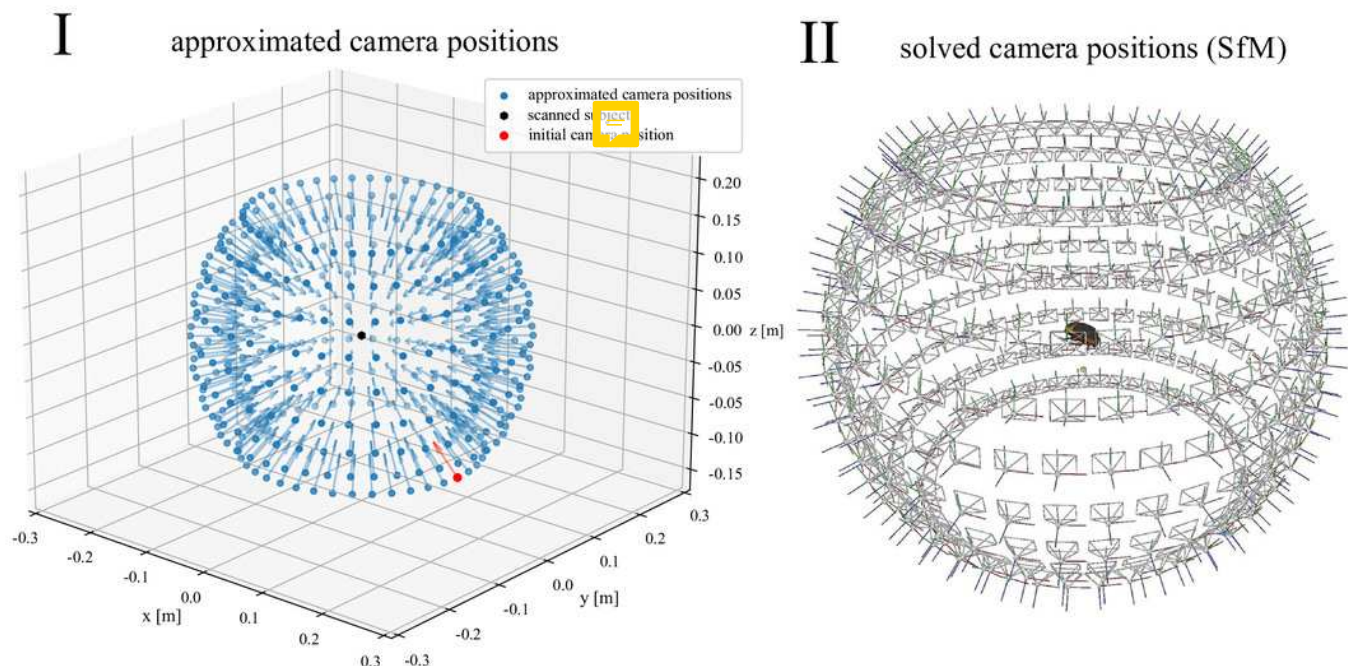


Figure 5

Mesh post-processing steps.

(A) ~~Mesh~~ cleaning process (~~*Pachnoda marginata*~~). (i) The mounting pin and any floating artefacts are removed from the generated mesh, and (ii) the topology is cleaned, using Laplacian smoothing, before (iii) producing the final mesh. (B) Thin, planar structures, such as some wings, are difficult to reconstruct but may be added to the models by approximating them as 2D objects. (i) Original, generated mesh. (ii) cut-outs of each wing are taken from masked EDOF images (~~*Sympetrum striolatum*~~). (iii) The cut-outs are used as image textures for planes of the corresponding size, containing an alpha layer, and are merged with the cleaned mesh. (C) Rigging process. Insets show close-ups of the legs and abdomen at each stage of the process (*Sungaya inexpectata*). (i) Reconstructed mesh pose as-scanned. (ii) Assigned armature superimposed over the original, untextured mesh. Joints are placed at each abdomen intersection as well as at every identified joint location to create a fully articulated model (in principle, the number of joints will be well defined for body parts such as legs, but may be harder to define for body parts such as antennae, where an appropriate number of joints may be determined by the desired flexibility and degree of accuracy for the posing). (iii) After assigning the surrounding vertices to their respective rigid bodies, the model can be posed arbitrarily. As an illustrative example, we extended the curled abdomen and posed the legs to reflect the posture of a freely standing stick insect.

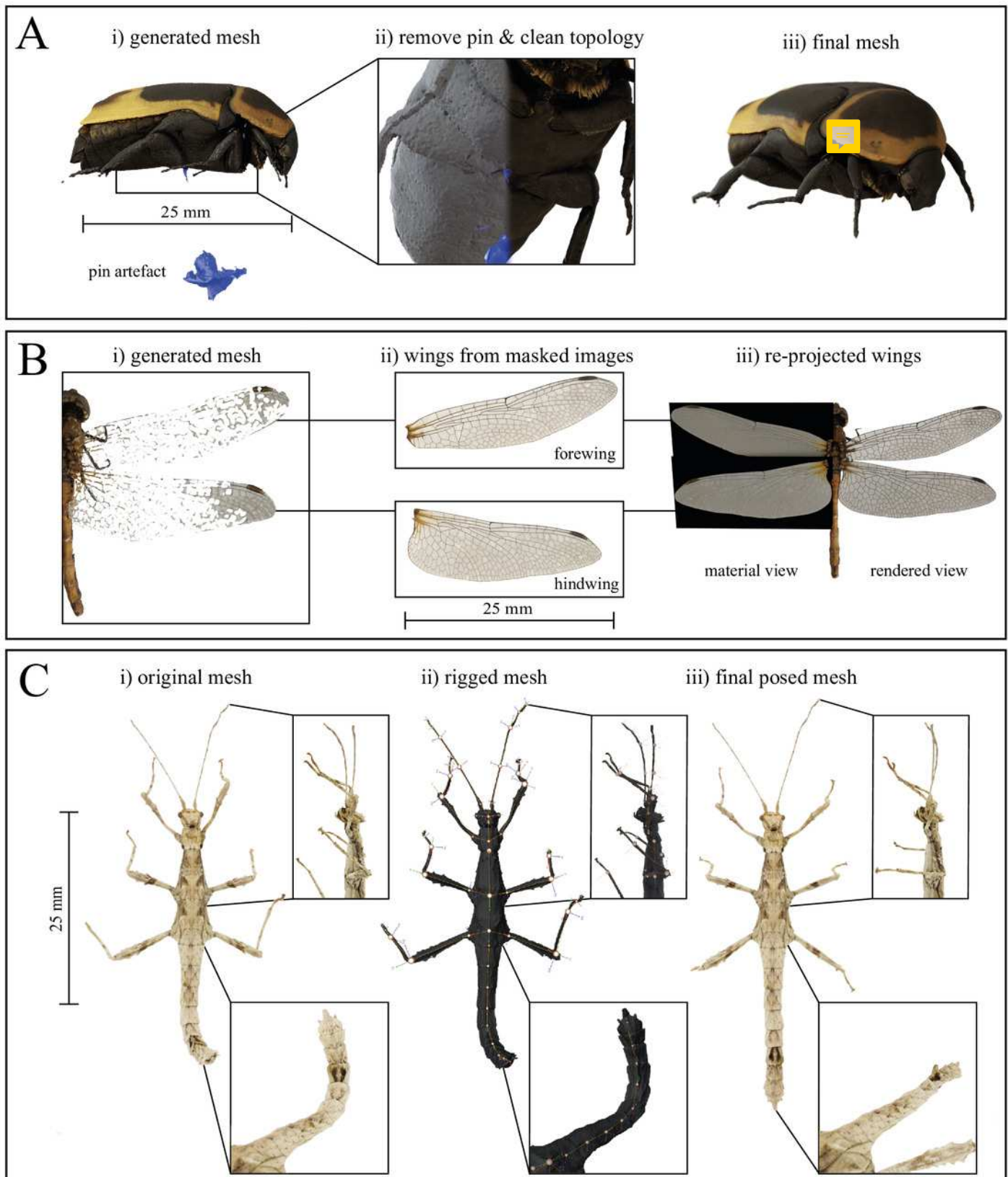


Figure 6

Examples of fully textured 3D models created with *scAnt*.

The models shown have been reconstructed with either Meshroom 2020 or 3DF Zephyr Lite (**Tab. 4**), post-processed and rendered in blender v2.8, using cycles. Boxes on the right illustrate various quality criteria for model reconstruction **(a)** Ventral, sagittal, anterior, and posterior view of *Porcellio scaber*, demonstrating the achievable model quality for the smallest scanned specimen. **(b)** Front right tarsus and claws of a *Camponotus gigas* specimen, illustrating the mesh density, as well as the texture quality at small scales. **(c)** Rigged, untextured mesh after the pin has been removed and mesh smoothing has been applied to the model. **(d)** & **(e)** Close-ups of the colour detail retained for the elytra and thorax of the reflective *Metallyticus splendidus* specimen. Topology and texture detail of a *Sungaya inexpectata* model reconstructed with Meshroom **(f)** and 3DF Zephyr Lite **(g)**. Zephyr captures fine surface texture better than Meshroom. All models, as well as additional details on SfM reconstruction and post-processing steps for each model, can be found on Sketchfab-.

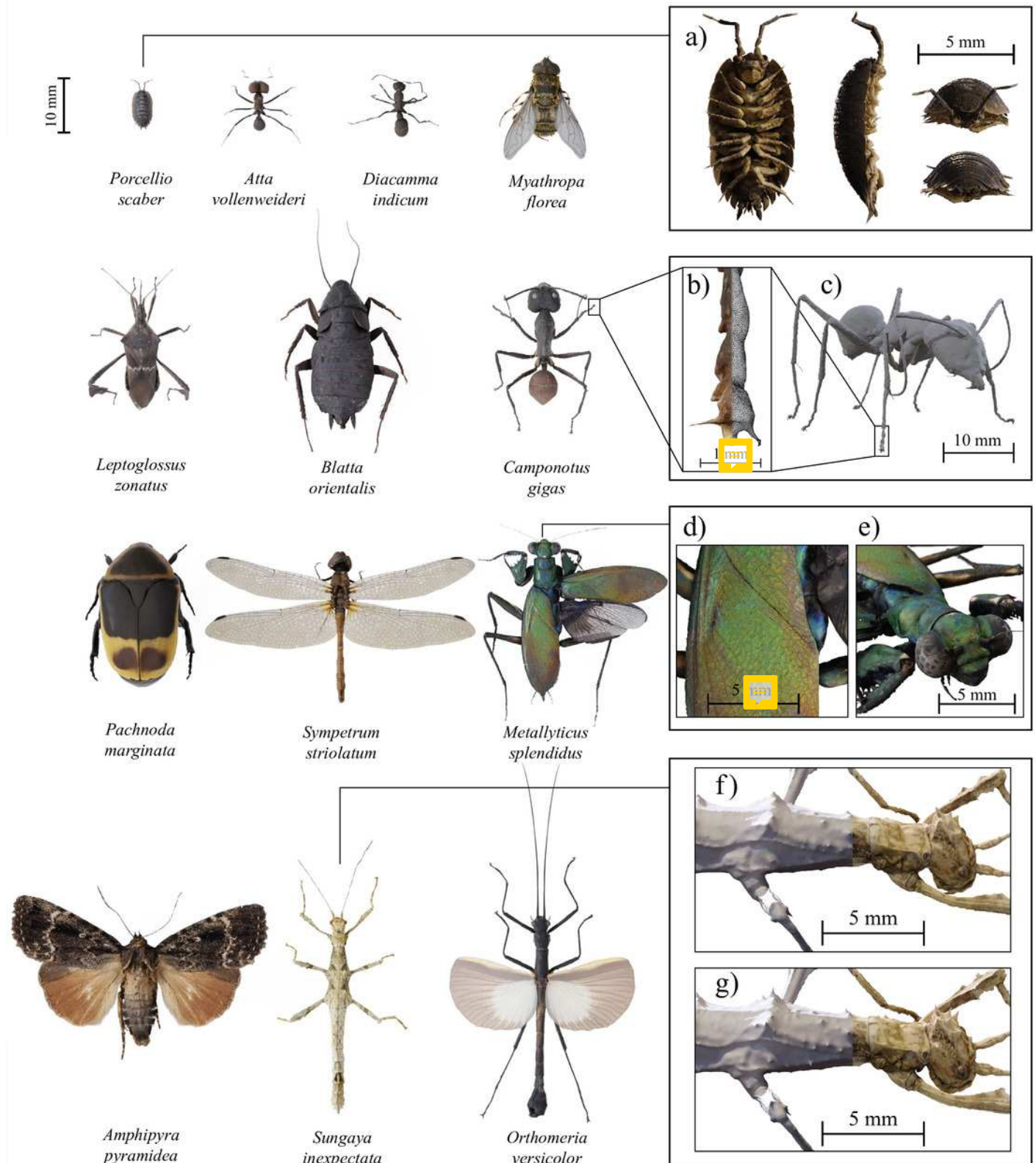


Table 1 (on next page)

List of modifiers required for the image stack alignment process of Hugin.

1

modifier	effect
-m	Use the field of view of the most distal captured focal plane as a reference, and align and rescale all successive images accordingly (required as the magnification of successive images increases as the camera moves towards the object; more sophisticated methods are available, provided that the exact position of the camera relative to the scanned object is known, see Ströbel <i>et al.</i> , 2018).
-x -y	Align each successive camera view in both x- and y-axes.
-c 100	Set the number of extracted control points to 100. The number of control points affects the accuracy of the correction to image pitch and yaw, as well as of the lens distortion. While more control points generally increase the accuracy of the alignment process, the processing time increases exponentially. In some cases, excessive numbers of extracted control points can even decrease alignment quality due to inaccurate matches, amplified by shallow focus overlap between images.
--gpu	Force the use of the GPU for remapping (optional)

2

Table 2(on next page)

List of modifiers required for the image stack merging process of Enfuse.

1

modifier	effect
--exposure-weight=0	Determines the contribution of pixels close to the ideal luminance of the blended image. A value of zero implies that all pixels contribute equally.
--saturation-weight=0	Determines the contribution of highly-saturated pixels. A value of zero implies that all pixels contribute equally.
--contrast-weight=1	Determines the contribution of pixels with high local contrast, which result from sharp edges. A value of one amplifies the contribution of these pixels.
--hard-mask	The use of hard-masks increases the level of detail in the final image and reduces halos where the outlines of focal planes overlap, as it uses only information from the sharpest focal plane
--contrast-edge-scale=1	Determines the pre-processing function used prior to edge detection. A value of unity uses local-contrast-enhancement.
--gray-projector=1-star	Determines the relative weight of the colour channels for greyscale conversion. By default, the colour channels are averaged, but if halos are visible in the EDOF image, 1-star can be set as the grey- projector instead (resulting in an emphasis of small contrast variations in highlights and shadows). The 1-star conversion is disabled by default, as it is more computationally expensive, but it can be activated within the scanner setup section of the GUI (Fig. 2) under the option “stacking method”.

2

Table 3(on next page)

List of key metadata parameters, required for the photogrammetry process.

parameter	value example	effect
Make	FLIR	Required for correct assignment in the camera sensor database
Model	BFS-U3-200S6C-C	
CameraSerialNumber	XXXXXXXX	All EDOF images must share the same CameraSerialNumber, so that the same camera intrinsics, solved once for a single scene, can be applied to every scene
FocalLength	35.0	Required to compute the correct magnification and distortion of each image. The FocalLength parameter refers to the value provided on the lens.
FocalLengthIn35mmFormat	95.0	This parameter refers to the equivalent focal length on 35 mm film, and therefore depends on the sensor width of the camera.
SensorWidth	13.1	This parameter is required by some meshing software to correctly compute the camera intrinsics in addition to the FocalLengthIn35mm-parameter.

Table 4(on next page)

Overview of scanned specimens.

The length of the longest body axis is measured on the final mesh within ~~b~~lender v2.8. When both Meshroom (**M**) and 3DF Zephyr (**Z**) are listed in the right column, separate meshes were created from the same masked EDOF images for comparison. (*) indicates the mesh shown in **Fig. 6**. (side by side comparisons available on Sketchfab)

species	length of longest axis (excluding antennae and legs)	authority	reconstructed using Meshroom (M), 3DF Zephyr Lite (Z)
<i>Amphipyra pyramidea</i>	20.40 mm	Linnaeus 1758	M & Z*
<i>Atta vollenweideri</i>	9.58 mm	Forel 1893	Z*
<i>Blatta orientalis</i>	27.51 mm	Linnaeus 1758	Z*
<i>Camponotus gigas</i>	22.59 mm	Smith 1858	Z*
<i>Diacamma indicum</i>	9.32 mm	Santschi 1920	Z*
<i>Leptoglossus zonatus</i>	19.38 mm	Dallas 1852	Z*
<i>Metallyticus splendidus</i>	29.98 mm	Westwood 1835	M, Z*
<i>Myathropa florea</i>	14.82 mm	Linnaeus 1758	M*, Z
<i>Orthomeria versicolor</i>	37.53 mm	Redtenbacher 1906	M*
<i>Pachnoda marginata</i>	26.66 mm	Kolbe 1906	M*
<i>Porcellio scaber</i>	7.07 mm	Latreille 1804	M, Z*
<i>Sungaya inexpectata</i>	36.27 mm	Zompro 1996	M*, Z*
<i>Sympetrum striolatum</i>	28.33 mm	Charpentier 1840	Z*

1

2

Table 5(on next page)

Distance measurements on certified gauge cubes to produce step-sizes of known height.

The reported measured mean refers to the distance between the vertices of the top plane of the step-cube (1.50 mm, 1.10 mm, 1.05 mm respectively) and the reference cube (1.00 mm).

step-size (mm)	0.500	0.100	0.050
measured mean (mm)	0.51727	0.11267	0.06541
std (mm)	0.00565	0.00623	0.00509
relative error	3.45 %	12.67 %	30.83 %

1

Table 6 (on next page)

Distances measured on a *Sungaya inexpectata* specimen.

Measurements have been performed using a LEICA Z6 Microscope on ~~the specimen itself~~ (2D), or blender v2.8 on the generated model (3D) (n=7 different observers). Listed are the shortest (**min**) and longest (**max**) distances, the mean (μ), the standard deviation (σ) and the coefficient of variation (c_v). We tested for differences in the coefficient of variation using the asymptotic test for the equality of coefficients of variation from k populations after Feltz and Miller, 1996-, implemented in the R-package cvequality (Marwick and Krishnamoorthy, 2019)-, for which we provide the test-statistic (D_{AD}), and the associated p-value.

	head width (mm)		femur length (mm)		abdomen length (mm)	
n = 7	2D	3D	2D	3D	2D (curved)	3D (straight)
min	3.0456	3.1766	6.0805	6.4451	16.5849	19.4330
max	3.1782	3.2207	6.4206	6.7019	19.7830	20.1077
μ	3.1205	3.2035	6.2526	6.5727	18.1537	19.5979
σ	0.0402	0.0156	0.1033	0.0784	1.1670	0.2019
c_V	0.0129	0.0049	0.0165	0.0119	0.0643	0.0103
D_{AD}	4.85, p = 0.028		0.85, p = 0.36		12.50, p < 0.0001	

1