

ampscan: A lightweight Python package for shape analysis of prosthetics and orthotics

Joshua W Steer¹, Oliver Stocks¹, Jack Parsons¹, Peter R Worsley², and Alexander S Dickinson¹

¹ Bioengineering Sciences Research Group, School of Engineering, Faculty of Engineering and Physical Sciences, University of Southampton ² Clinical Academic Facility, School of Health Sciences, Faculty of Environment and Life Sciences, University of Southampton

DOI: [10.21105/joss.02060](https://doi.org/10.21105/joss.02060)

Software

- [Review](#) ↗
- [Repository](#) ↗
- [Archive](#) ↗

Editor: [Kevin M. Moerman](#) ↗

Reviewers:

- [@danasolav](#)
- [@Kevin-Mattheus-Moerman](#)

Submitted: 08 November 2019

Published: 13 April 2020

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC-BY](#)).

Summary

The increasing accessibility of surface scanners is giving users the ability to accurately digitise the 3D surface geometry of real world objects which may then be manufactured by CNC machining or 3D printing. In addition to hobbyist applications, these devices are being increasingly used within the medical field, for example in prosthetics and orthotics clinics to capture the patients' individual limb geometry. These scans are imported into computer-aided design packages to generate patient-specific medical devices, such as prosthetic sockets, spinal braces or ankle-foot orthoses. This increasing digitisation of patient data provides great potential for analysis in order to inform and improve clinical practice. While this has been an area of academic interest for several decades, clinical use is limited to date. One of the reasons for this is the lack of tools available for clinicians to analyse the geometry of their patient datasets.

Building on foundational work published by Zachariah, Sorenson and Sanders (Zachariah, Sorenson, & Sanders, 2005), in a previously published paper (Dickinson, Steer, Woods, & Worsley, 2016) we detailed a method for comparing pairs of surface scans using alignment, registration and visualisation. This enabled extraction of key measurements for evaluation of the accuracy and reliability of clinically-used scanners. Further studies have also demonstrated using this package for comparing the consistency of casting techniques (Dickinson A, 2020), statistical shape modeling across the population (Steer et al., 2019) and quantifying rectifications between the residual limb and the prosthetic socket (Steer et al., 2017).

This method was written in MATLAB, although this was considered restrictive for other researchers to access as they may not possess a relevant license. In order to maximise access to the developed techniques and improve performance, especially for 3D visualisation, the methods were rewritten from scratch within Python, including updated algorithms for each stage of the process.

The ampscan package has been designed with a range of user groups in mind, from practicing clinicians to biomedical engineering researchers, with an appreciation that they may not have an extensive background in coding. To this aim, the software has been developed in Python and leverages the commonly used libraries of NumPy, SciPy, matplotlib, vtk and pyqt. As such, full functionality of the software can be accessed without requiring additional installs. The core analysis of the package can be carried out either within a script-based environment, or with the supplied lightweight graphical user interface (J. Steer et al., 2019).



Figure 1: A graphical summary of a typical ampscan process. a) Importing a pair of scans, b) Automatically snap centres, c) Align using ICP algorithm, d) Register and visualise shape deviation and e) Automatically analyse the registered scan

The typical ampscan process is demonstrated above (Figure 1) and the core functionality summarised below, with a more detailed description available in the online [documentation](#):

- **AmpObject:** this is the core object of the package and holds the key data and methods. The key data held within the AmpObject is the mesh data including arrays of the vertices, faces, normals and field values. The vtk actor for visualisation is also stored. The **core** methods of the AmpObject include imports for .stl files, saving .stl files, rotation and translation. Further methods on the core AmpObject include **smoothing**, **trimming** and **visualisation**.
- **Alignment:** This takes two AmpObjects, one fixed and one moving, and applies a rigid transformation to the moving AmpObject in order to minimise the spatial error between the two AmpObjects. This is performed through an Iterative Closest Point (ICP) algorithm.
- **Registration:** This takes two AmpObjects, one baseline and one target, and applies a non-rigid transformation to morph the baseline vertices onto the surface of the target. This is performed by a point-to-plane method. The registered shape ends up with the same number of vertices and connectivity as the baseline, thereby enabling shape comparison.
- **Analysis:** This is used to extract key geometrical information about the shape, including volume and serial slice cross section areas and perimeters.
- **Graphical User Interface:** This enables visualisation of multiple AmpObjects within a single window, giving access to the automated and manual alignment tools as well as registration. This facilitates the core analysis of the scan data for people who are not experienced Python users. The Python version of the GUI is available in the **gui** folder from the git repo.

Other packages perform the specific tasks of ampscan more comprehensively such as mayavi for scientific data visualisation (Ramachandran & Varoquaux, 2011), open3d for alignment and

registration techniques (Zhou, Park, & Koltun, 2018), and GIBBON for integration with finite element modelling (Moerman, 2018). At this time, the package only contains limited mesh fixing tools that may be required for erroneous scans. Until such methods are implemented, the authors recommend meshlab which also contains methods for alignment and visualisation (Cignoni et al., 2008).

An ampscan [webapp](#) is currently in development to further increase accessibility of these tools for clinicians and researchers working across the growing field of bespoke digital medical device design, and enable them to perform standardised, accessible and reliable analysis, and enhance patient experience.

Acknowledgments

While the ampscan package was not funded directly, the authors would like to thank the following for their financial support:

- JWS: the University of Southampton's EPSRC Doctoral Training Program (ref EP/M508147/1) and EUROSTARS project (ref 9396).
- PRW: the EPSRC-NIHR "Medical Device and Vulnerable Skin Network" (ref EP/M000303/1).
- ASD: the Royal Academy of Engineering, UK, (ref RF/130) and the EPSRC-NIHR "Global Challenges Research Fund" (ref EP/R014213/1).

Finally, the authors would like to thank the Research Software Group at the University of Southampton for all their support during the development of this software, as well as Omar Animashaun and Tim Dunn for their contributions to ampscan.

References

- Cignoni, P., Callieri, M., Corsini, M., Dellepiane, M., Ganovelli, F., & Ranzuglia, G. (2008). MeshLab: an Open-Source Mesh Processing Tool. In V. Scarano, R. D. Chiara, & U. Erra (Eds.), *Eurographics italian chapter conference*. The Eurographics Association. doi:[10.2312/LocalChapterEvents/ItalChap/ItalianChapConf2008/129-136](#)
- Dickinson, A. S., Steer, J. W., Woods, C. J., & Worsley, P. R. (2016). Registering a methodology for imaging and analysis of residual-limb shape after transtibial amputation. *Journal of rehabilitation research and development*, 53(2), 207–218. doi:[10.1682/jrrd.2014.10.0272](#)
- Dickinson A, K. S., Donovan-Hall M. (2020). Selecting appropriate 3D scanning technologies for prosthetic socket design and transtibial residual limb shape characterisation. *engrXiv*. doi:[10.31224/osf.io/s4kbn](#)
- Moerman, K. M. (2018). GIBBON: The geometry and image-based bioengineering add-on. *The Journal of Open Source Software*. doi:[10.21105/joss.00506](#)
- Ramachandran, P., & Varoquaux, G. (2011). Mayavi: 3D visualization of scientific data. *Computing in Science & Engineering*, 13(2), 40–51. doi:[10.1109/MCSE.2011.35](#)
- Steer, J., Blab, F., Worsley, P., Browne, M., Ganter, H., & Dickinson, A. (2017). Comparing residuum-socket interface pressure and soft tissue deformations for three transtibial trial socket designs. In *AOPA world congress*.
- Steer, J. W., Browne, M., Worsley, P. R., & Dickinson, A. S. (2019). Predictive prosthetic socket design. Part 1: Population-based evaluation of transtibial prosthetic sockets by

- fea-driven surrogate modelling. *Biomechanics and Modeling in Mechanobiology*. doi:[10.1007/s10237-019-01195-5](https://doi.org/10.1007/s10237-019-01195-5)
- Steer, J., Worsley, P., Stocks, O., Animashaun, O., & Dickinson, A. (2019). AmpScan: Open-source 3D scan shape analysis for prosthetics and orthotics. *Prosthetics and Orthotics International*, 43(1_suppl), 94. doi:[10.1177/0309364619883197](https://doi.org/10.1177/0309364619883197)
- Zachariah, S. G., Sorenson, E., & Sanders, J. E. (2005). A method for aligning trans-tibial residual limb shapes so as to identify regions of shape change. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 13(4), 551–557. doi:[10.1109/tnsre.2005.858459](https://doi.org/10.1109/tnsre.2005.858459)
- Zhou, Q.-Y., Park, J., & Koltun, V. (2018). Open3D: A modern library for 3D data processing. *arXiv:1801.09847*.