

# OpenSCAD Batch Exporter: Automating Parametric Design Workflows for Research and Engineering

Cameron K. Brooks<sup>1</sup> and Joshua M. Pearce<sup>1,2</sup>

<sup>1</sup> Department of Electrical & Computer Engineering, Western University, London, ON N6A 5B9, Canada

<sup>2</sup> Ivey Business School, Western University, London, ON N6A 5B9, Canada

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

## Software

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

Editor: [Open Journals](#) ↗

## Reviewers:

- [@openjournals](#)

Submitted: 01 January 1970

Published: unpublished

## License

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

## Summary

The OpenSCAD Batch Exporter is an open-source tool designed to automate the generation of STL models using parametric designs in OpenSCAD. By leveraging parameter files in CSV or JSON format, the tool simplifies the creation of multiple design variations, making it invaluable for research, prototyping, and engineering. Users can rapidly generate tailored design sets with ease, allowing accelerating and enabling novel workflows for applications such as experimental analysis, mass customization, and combinatorial design testing.

This tool addresses critical pain points in research and engineering workflows, such as the need to manually create or export designs from code-based CAD systems. By offering batch exporting capabilities, it supports efficient testing and analysis, promotes the use of parametric models in low-resource environments, and reduces bandwidth requirements by allowing compact script distribution instead of large 3D model files.

Inspired by community contributions and existing batch exporting tools, this project provides a polished and user-friendly solution, bridging gaps in usability and scalability within the OpenSCAD ecosystem.

## Statement of Need

Parametric design tools like OpenSCAD enable researchers and engineers to create highly customizable computational 3D models. However, exporting multiple designs based on varied parameter sets can be time-consuming and error-prone. The OpenSCAD Batch Exporter automates this process, enabling rapid generation of design variants, which is critical for:

### 1. Research Applications:

- Supports iterative experimental setups, such as those used in agrivoltaics ([Pearce, 2021](#)), chemical synthesis ([Hou et al., 2021](#)), electronic components ([Brooks et al., 2023](#)), scientific hardware ([Hietanen et al., 2018](#); [LeSuer et al., 2018](#)), medical devices ([Gallup et al., 2020](#); [S. Oberloier et al., 2022](#); [Petsiuk et al., 2020](#)), manufacturing ([Beeker et al., 2018](#)) and hardware prototyping ([Shane Oberloier & Pearce, 2018](#)).
- Allows researchers to optimize designs by testing numerous parameter variations without manual intervention.

### 2. Mass Customization and Low-Resource Settings:

- Facilitates the distribution of compact scripts that can generate large design libraries, reducing storage and bandwidth requirements. This is especially beneficial for researchers in low-resource environments.
- Promotes distributed manufacturing and parametric design practices ([Machado et al., 2019](#)).

### 3. Ease of Use:

- Provides a straightforward interface for non-experts, allowing users to interact with pre-made parameter sets without coding knowledge.
- Bridges the gap between OpenSCAD's code-based environment and accessible design workflows for scientific and engineering applications.

This tool significantly extends the usability of OpenSCAD, making it easier for researchers, engineers, educators, and makers to integrate data-driven design into their workflows.

## Functionality

The OpenSCAD Batch Exporter allows users to:

- Batch Export:** Supports exporting STL files in ASCII or binary format using parameter files in CSV or JSON format.
- Selection Options:** Allows users to target specific parameter sets for export using a flexible selection syntax.
- Format Conversion:** Includes utilities for converting between CSV and JSON parameter files.
- Graphical User Interface (GUI):** Simplifies usage for non-technical users, while retaining a command-line interface for advanced workflows.
- Cross-Platform Compatibility:** Works on systems where OpenSCAD is installed, leveraging its command-line capabilities.

A command-line interface and an intuitive graphical user interface (GUI) make the tool adaptable for users of varying expertise levels. As illustrated in Figure 1, the OpenSCAD Batch Exporter GUI provides an intuitive interface for configuring batch exports. Users can select the OpenSCAD file, parameter file, and output directory, and specify export settings.

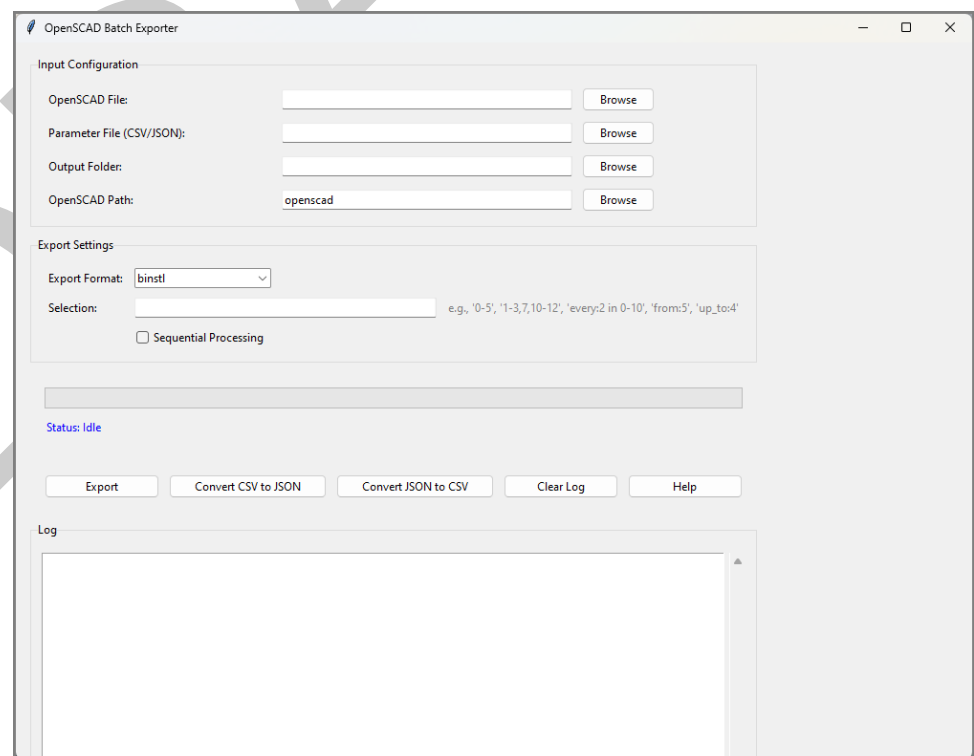


Figure 1: The OpenSCAD Batch Exporter GUI in idle state.

64 During the export process, users can monitor progress and view logs, as shown in Figure  
65 Figure 2.

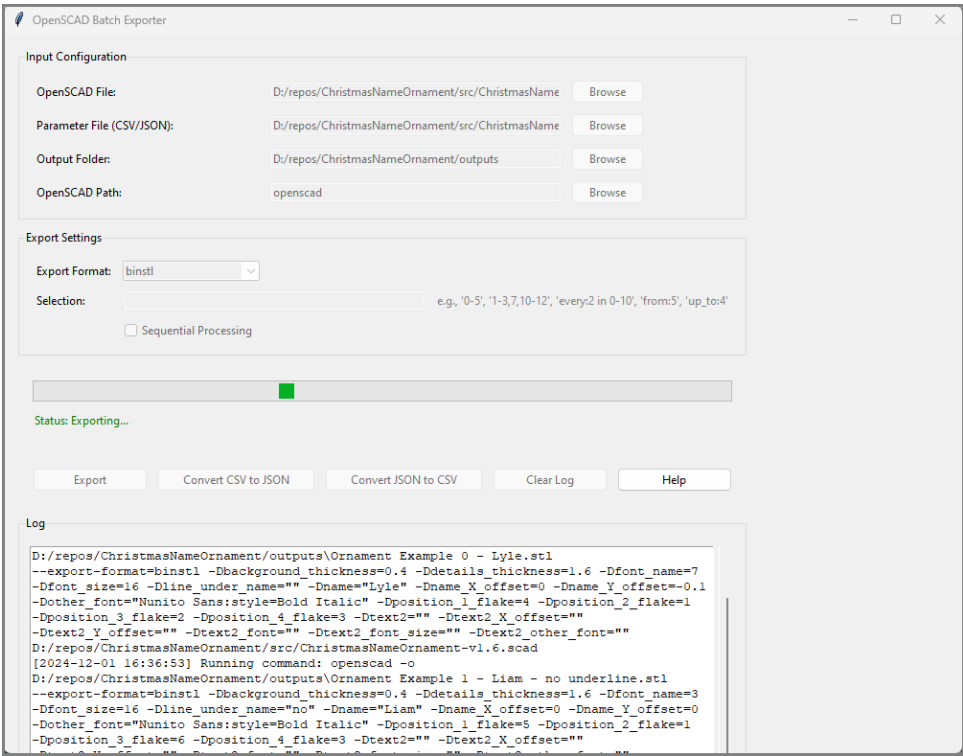


Figure 2: Configuring batch export parameters and monitoring progress in the GUI.

66 After the batch export completes, the generated STL files can be accessed, exemplified in  
67 Figure Figure 3.

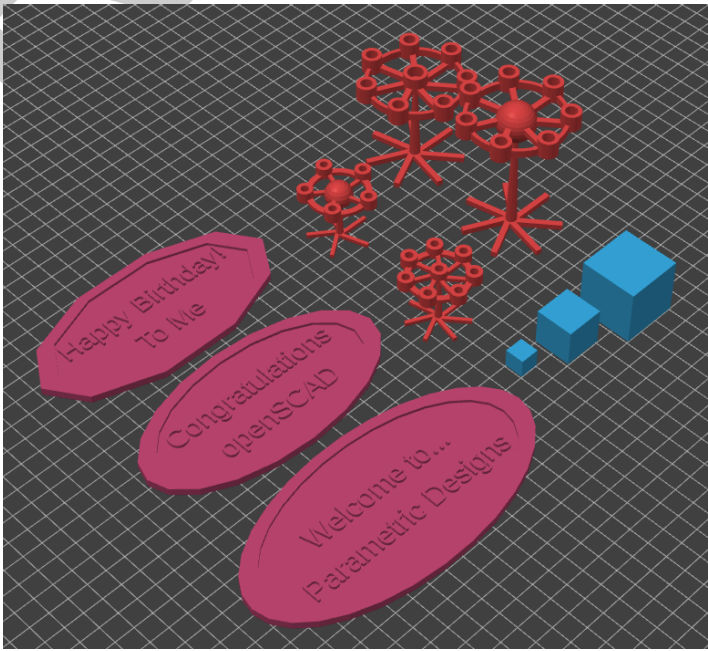
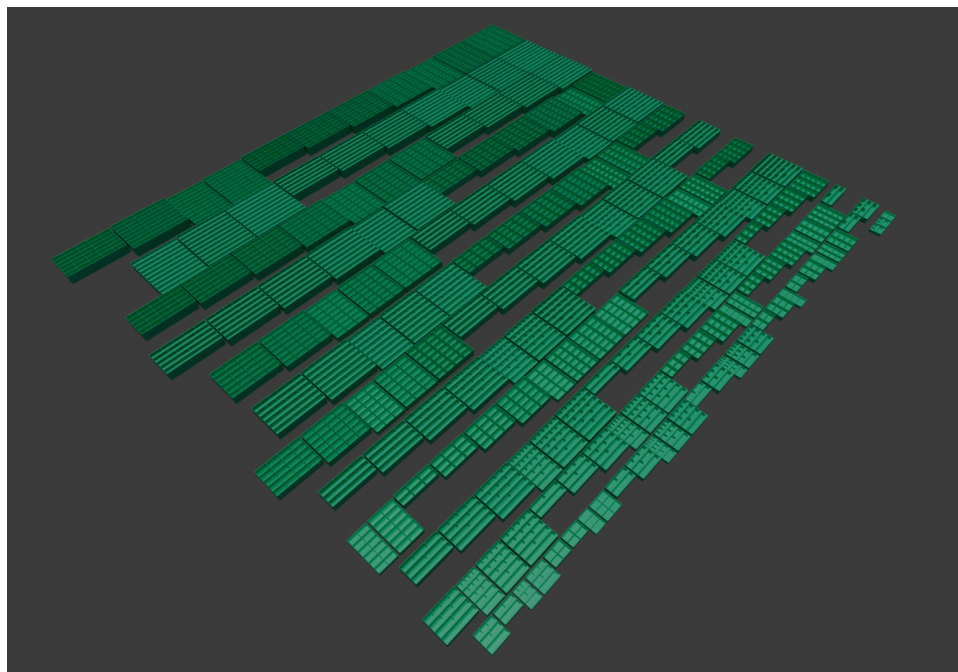


Figure 3: Examples of generated STL outputs from batch exporting.

68 The OpenSCAD Batch Exporter was validated for large-scale operations using the Gridfinity  
69 storage system rebuilt in OpenSCAD ([Kenneth, 2024](#)), generating 200 unique models to test  
70 performance and reliability under high-demand workloads. The resulting STL models were  
71 visualized and analyzed in Blender, as shown in [Figure 4](#).



**Figure 4:** Visualization of 200 Gridfinity models generated using the OpenSCAD Batch Exporter.

72 This entire development and testing process was conducted using the latest stable release,  
73 OpenSCAD 2021.01 ([OpenSCAD, 2021](#)).

## 74 Related Work

75 This tool builds on efforts by the OpenSCAD community to streamline batch exporting ("[Multi  
76 File Export](#)," 2020; "[Using Batch Files and OpenSCAD to Generate STL's](#)," 2018). It integrates  
77 and improves upon existing approaches, notably:

- 78     ▪ [OpenSCAD Batch Export STL](#)
- 79     ▪ [OpenSCAD Bulk Export](#)

80 Unlike prior methods and tools, the presented OpenSCAD Batch Exporter offers a complete  
81 and user-friendly package, with added functionality for parametric design workflows, broader  
82 compatibility, and robust documentation. It is currently being used for an ongoing research  
83 project called SnapTessSCAD ([Brooks, 2024](#)), a complex OpenSCAD-based design program  
84 for computational design of 3D-printable interlocking connectors, validating the robustness  
85 and utility of the OpenSCAD Batch Exporter program.

## 86 Acknowledgments

87 This project was supported by the Thompson Endowment and The Natural Sciences and  
88 Engineering Research Council of Canada. We thank the OpenSCAD community for their  
89 foundational contributions, which inspired this project.

## References

- Beeker, L. Y., Pringle, A. M., & Pearce, J. M. (2018). Open-source parametric 3-D printed slot die system for thin film semiconductor processing. *Additive Manufacturing*, 20, 90–100. <https://doi.org/10.1016/j.addma.2017.12.004>
- Brooks, C. K. (2024). Uwo-fast/SnapTessSCAD. In *GitHub*. <https://github.com/uwo-fast/SnapTessSCAD>
- Brooks, C. K., Peplinski, J. E., & Pearce, J. M. (2023). Overcoming Chip Shortages: Low-Cost Open-Source Parametric 3-D Printable Solderless SOIC to DIP Breakout Adapters. *Inventions*, 8(2), 61. <https://doi.org/10.3390/inventions8020061>
- Gallup, N., Pringle, A. M., Oberloier, S., Tanikella, N. G., & Pearce, J. M. (2020). Parametric nasopharyngeal swab for sampling COVID-19 and other respiratory viruses: Open source design, SLA 3-D printing and UV curing system. *HardwareX*, 8. <https://doi.org/10.1016/j.ohx.2020.e00135>
- Hietanen, I., Heikkinen, I. T. S., Savin, H., & Pearce, J. M. (2018). Approaches to open source 3-D printable probe positioners and micromanipulators for probe stations. *HardwareX*, 4. <https://doi.org/10.1016/j.ohx.2018.e00042>
- Hou, W., Bubliauskas, A., Kitson, P. J., Francoia, J.-P., Powell-Davies, H., Gutierrez, J. M. P., Frei, P., Manzano, J. S., & Cronin, L. (2021). Automatic Generation of 3D-Printed Reactionware for Chemical Synthesis Digitization using ChemSCAD. *ACS Central Science*, 7(2), 212–218. <https://doi.org/10.1021/acscentsci.0c01354>
- Kenneth. (2024). *Kennetek/gridfinity-rebuilt-openscad*. <https://github.com/kennetek/gridfinity-rebuilt-openscad>
- LeSuer, R. J., Osgood, K. L., Stelnicki, K. E., & Mendez, J. L. (2018). OMIS: The Open Millifluidic Inquiry System for small scale chemical synthesis and analysis. *HardwareX*, 4. <https://doi.org/10.1016/j.ohx.2018.e00038>
- Machado, F., Malpica, N., & Borromeo, S. (2019). Parametric CAD modeling for open source scientific hardware: Comparing OpenSCAD and FreeCAD Python scripts. *PLOS ONE*, 14(12), e0225795. <https://doi.org/10.1371/journal.pone.0225795>
- Multi File export. (2020). In *GitHub*. <https://github.com/openscad/openscad/wiki/Multi-File-export>
- Oberloier, S., Gallup, N., & Pearce, J. M. (2022). Overcoming supply disruptions during pandemics by utilizing found hardware for open source gentle ventilation. *HardwareX*, 11. <https://doi.org/10.1016/j.ohx.2021.e00255>
- Oberloier, Shane, & Pearce, J. M. (2018). General Design Procedure for Free and Open-Source Hardware for Scientific Equipment. *Designs*, 2(1), 2. <https://doi.org/10.3390/designs2010002>
- OpenSCAD. (2021). *Openscad/openscad*. openscad. <https://github.com/openscad/openscad>
- Pearce, J. M. (2021). Parametric Open Source Cold-Frame Agrivoltaic Systems. *Inventions*, 6(4), 71. <https://doi.org/10.3390/inventions6040071>
- Petsiuk, A., Tanikella, N. G., Dertinger, S., Pringle, A., Oberloier, S., & Pearce, J. M. (2020). Partially RepRapable automated open source bag valve mask-based ventilator. *HardwareX*, 8. <https://doi.org/10.1016/j.ohx.2020.e00131>
- Using batch files and OpenSCAD to generate STL's. (2018). In *3D Printing Forum - Pinshape*. <https://forums.pinshape.com/t/using-batch-files-and-openscad-to-generate-stls/2157>