

# Paperfold

## About

Paperfold is another flattener for triangle mesh files, heavily based on <https://github.com/felixfeliz/paperfoldmodels> by Felix Scholz aka felixfeliz.

Possible input files are STL, Wavefront OBJ, PLY and OFF. (possible import file types with OpenMesh in principle: [https://www.graphics.rwth-aachen.de/media/openmesh\\_static/Documentations/OpenMesh-8.0-Documentation/a04096.html](https://www.graphics.rwth-aachen.de/media/openmesh_static/Documentations/OpenMesh-8.0-Documentation/a04096.html))

## Prepare to install openmesh

To run this extension you need to install OpenMesh bindings with python pip. When installing Paperfold by installing MigHyScape, this will already be done for you on Linux systems. Else:

### Ubuntu

```
sudo apt install cmake
sudo apt install g++
sudo apt install python3-dev
pip3 install openmesh
```

### Fedora

```
sudo dnf install cmake
sudo dnf install g++
sudo dnf install python3-devel
pip3 install cmake
pip3 install pybind11
pip3 install openmesh
```

### Windows

1. Windows **requires** [CMake executable](#) (but version 3, not 4!)
2. Install MS Visual Studio Community Edition → If you receive an error like "NMake Makefiles does not support platform specification, but platform x64 was specified." In case the installation fails you can also try to install precompiled Windows binaries from <https://gitlab.vci.rwth-aachen.de:9000/OpenMesh/OpenMesh>
3. Install openmesh bindings

```
pip3 install openmesh
```

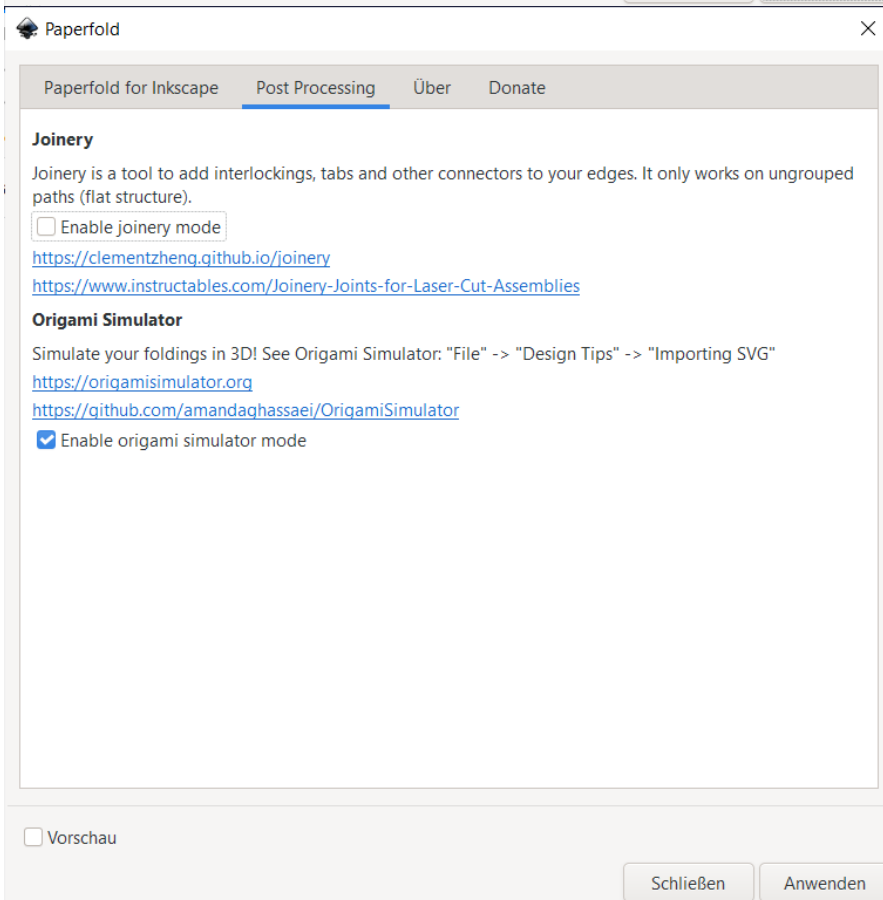
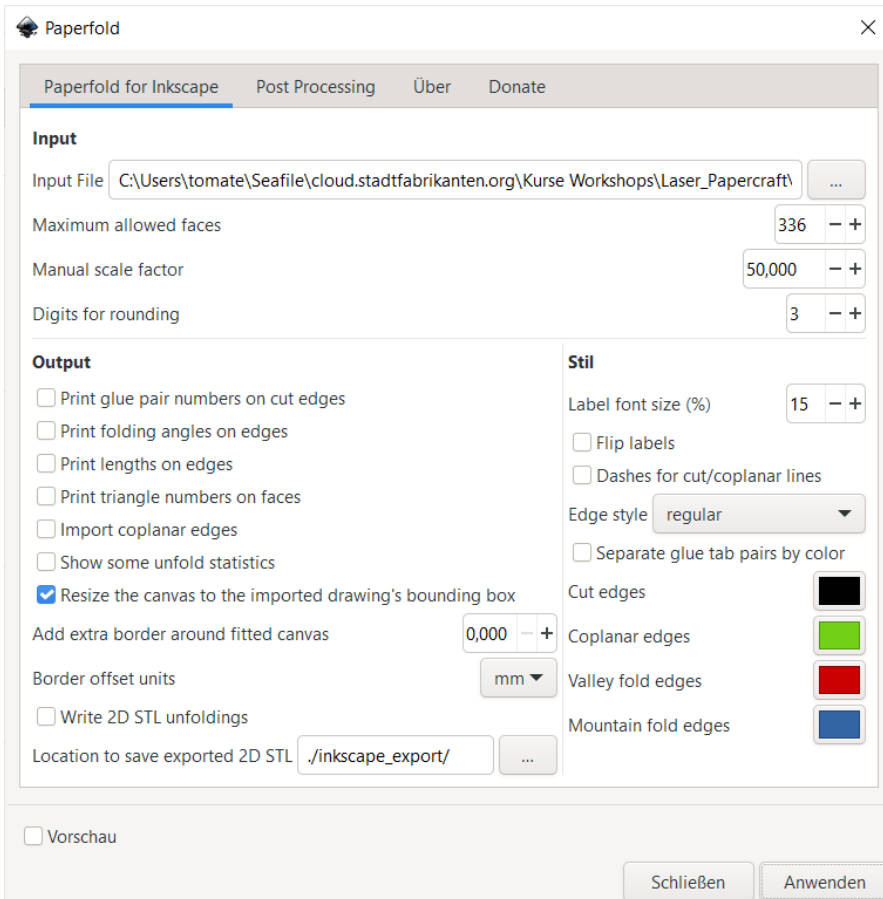
## How it works

The algorithm of paperfoldmodels consists of three steps:

1. Find a minimum spanning tree of the dual graph of the mesh.
2. Unfold the dual graph.
3. Remove self-intersections by adding additional cuts along edges.

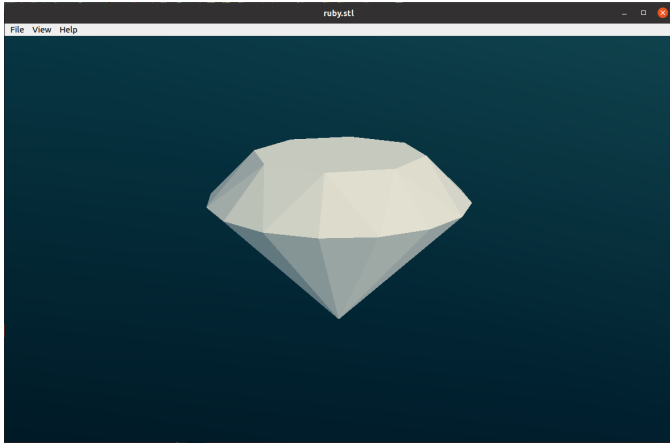
Reference: The code is mostly based on the algorithm presented in a by Straub and Prautzsch ([https://geom.ivd.kit.edu/downloads/proj-paper-models\\_cut\\_out\\_sheets.pdf](https://geom.ivd.kit.edu/downloads/proj-paper-models_cut_out_sheets.pdf)).

Please also have a look at [DXF 2 Papercraft](#) and [Papercraft Unfold](#).

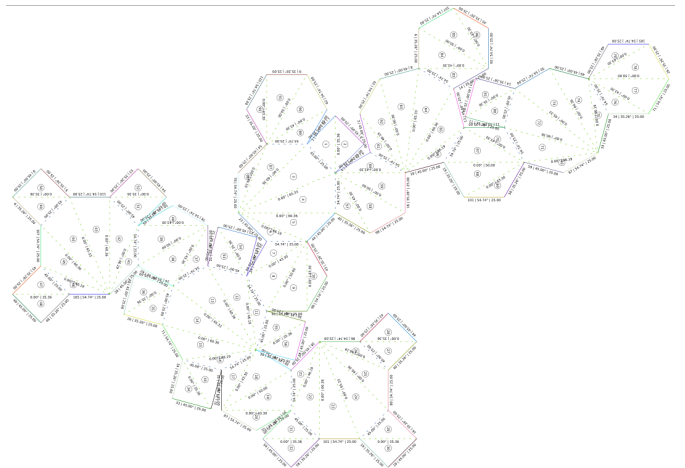
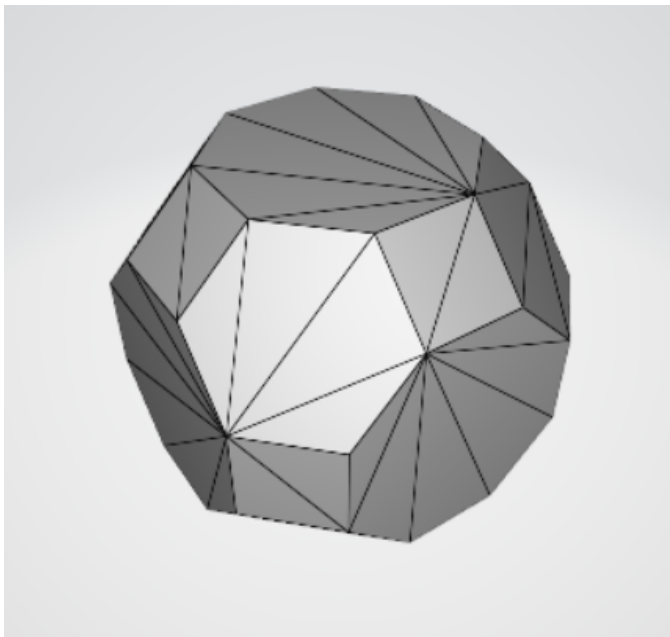
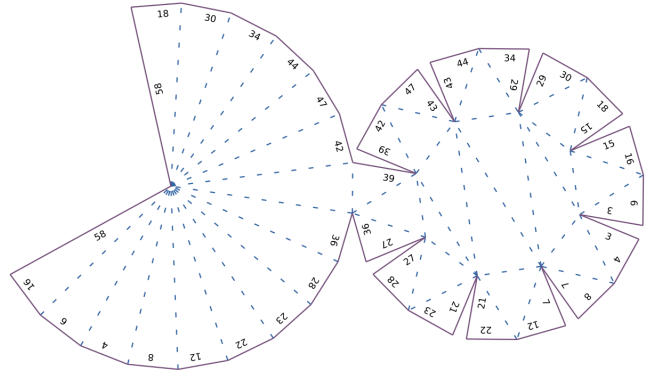


Example with a Ruby model file

**STL preview**

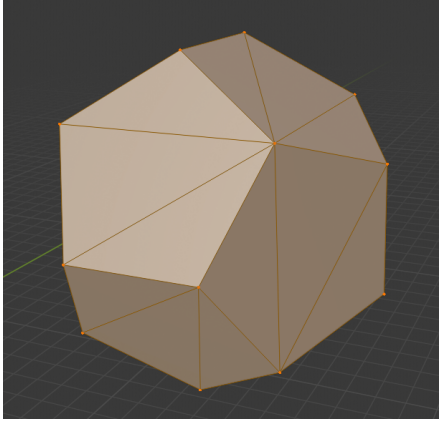
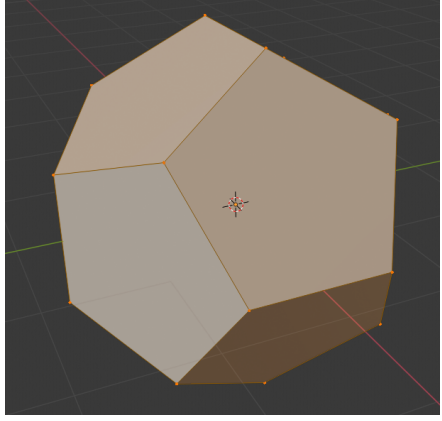
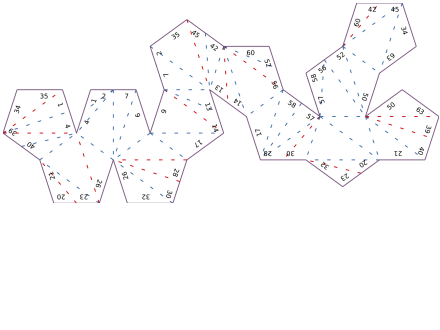
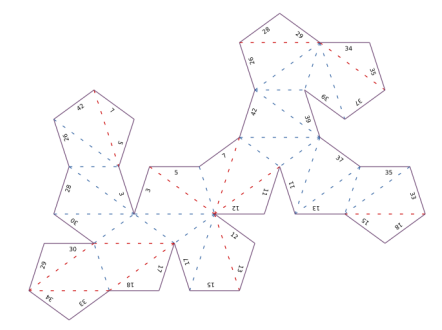


**Unwrapped by Paperfold**



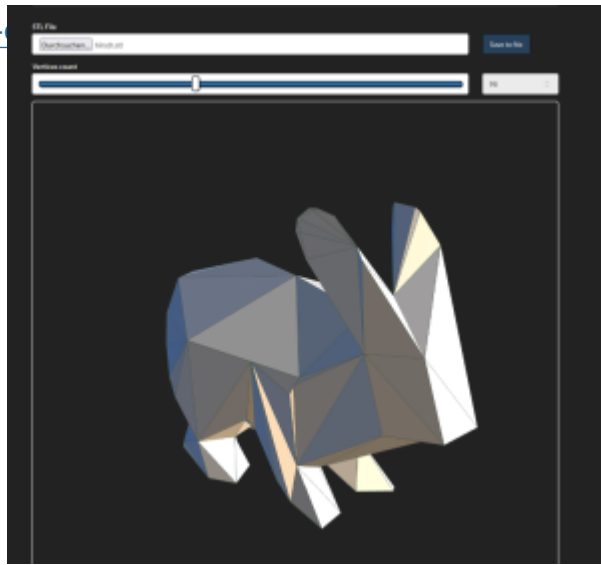
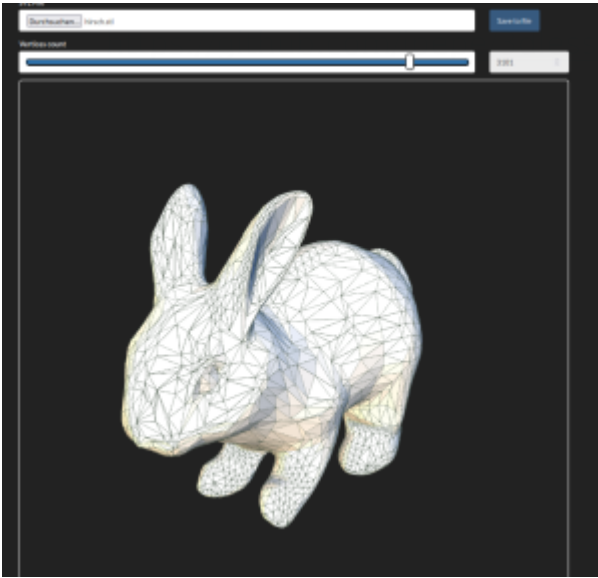
# Differences between poly facets and triangles

There is a great difference which comes from the input file. If you use polygonal input file, the unfolded mesh looks better because it has less useless folds and glue tab numbers. There is no algorithm to remove fold lines for coplanar adjacent triangles which form a polygon facet.

	Triangle mesh file (model saved stl)	Polygonal facets (model saved as .obj)
3D mesh preview (Blender)		
Unfolding		

## Hot tips

- Add some joints to your paperfold using Joinery
  - <https://clementzheng.github.io/joinery>
  - <https://www.instructables.com/Joinery-Joints-for-Laser-Cut-Assemblies>
- ...or add tabs using [Tab Generator](#)
- Simulate using Origami Simulator
  - <https://origamisimulator.org>
  - <https://github.com/amandaghassaei/OrigamiSimulator>
- generate cool model files from more complex 3d models using a vertice reducing tool:



---

Version #6

Erstellt: 2025-05-24 14:29:55 CEST von Mario Voigt

Zuletzt aktualisiert: 2025-08-27 10:44:45 CEST von Mario Voigt