Files

main Go to file

Im2mesh_GUI app
Im2mesh_Matlab
Im2mesh_Octave
demo_html
COPYRIGHT NOTICE.txt
Im2mesh functions and param...
Im2mesh functions and param...
Im2mesh_GUI Tutorial.pdf
LICENSE
README.md

Im2mesh (2D image to finite element mesh)

Im2mesh is a MATLAB/Octave package for generating finite element mesh based on 2D segmented multi-phase image. It provides a robust workflow capable of processing various input images, such as microstructure images of engineering materials. Due to its generalized framework, Im2mesh can handle segmented image with more than 10 phases. Im2mesh was originally released on MathWorks File Exchange in 2019. Im2mesh can also be used as a mesh generation interface for MATLAB multi-part geometry.

News:

- **Version 2.2.1 is able to edit polygonal boundary before mesh generation. Check demo14-16.**
- Version 2.2.0 supports using Gmsh as mesh generator (unstructured quadrilateral mesh).
- Version 2.1.6 updates the DOI. Im2mesh is now citable.

Features:

- Accurately preserve the contact details between different phases.
- Incorporates polyline smoothing and simplification
- Able to avoid sharp corners when simplifying polylines.
- Support phase selection before meshing.
- 3 mesh generators are available for selection: MESH2D, generateMesh, and Gmsh.
- Generated mesh can be exported as `inp` file (Abaqus) and `bdf` file (Nastran bulk data, compatible with COMSOL). Mesh can be exported as many formats via Gmsh, such as STL.
- Graphical user interface (GUI) version is available as a MATLAB app.

Dependencies

- When using Im2mesh package in MATLAB, you need to install MATLAB Image Processing Toolbox and Mapping Toolbox.
- When using Im2mesh package in GNU Octave, you are not required to install these toolboxes.

Version compatibility

- Im2mesh_GUI: MATLAB R2017b or later; version higher than R2018b is preferred.
- Im2mesh package: MATLAB R2017b or later. GNU Octave 9.3.0 or later.

How to start

After downloading Im2mesh package (releases), I suggest you start with Im2mesh_GUI app in the folder, which will help you understand the workflow and parameters of Im2mesh. A detailed tutorial is provided in Im2mesh_GUI Tutorial.pdf.

Then, you can learn to use Im2mesh package in the folder "Im2mesh_Matlab" or "Im2mesh_Octave". 16 examples are provided.

- If you're using MATLAB, examples are live script `mlx` files (`demo1.mlx` ~ `demo16.mlx`). Note that `demo02.mlx` requires MATLAB Partial Differential Equation (PDE) Toolbox. If you don't have PDE Toolbox, you can skip `demo02.mlx` .
- If you're using Octave, examples are `m` files (`demo1.m` ~ `demo10.m`).
- Examples are also available as `html` files in the folder "demo_html".

Examples:

- demo01 - Demonstrate function `im2mesh` , which use MESH2D as mesh generator.
- demo02 - Demonstrate function `im2meshBuiltIn` , which use MATLAB built-in function `generateMesh` as mesh generator.
- demo03 - How to export mesh as `inp` , `bdf` , and `node` / `ele` file
- demo04 - What is inside function `im2mesh` .
- demo05 - Parameter `tf_avoid_sharp_corner`
- demo06 - Thresholds in polyline smoothing
- demo07 - Parameter `grad_limit` for mesh generation

- demo08 - Parameter `hmax` for mesh generation
- demo09 - How to select phases for meshing
- demo10 - Different polyline smoothing techniques
- demo11 - How to find node sets at the interface and boundary
- demo12 - Demonstrate function `pixelMesh` (pixel-based quadrilateral mesh)
- demo13 - How to use Gmsh as mesh generator
- demo14 - How to use polyshape object to define multi-part geometry for mesh generation
- demo15 - How to edit polygonal boundaries before meshing
- demo16 - How to add points or nodes to polygonal boundaries before meshing

Cite as

If you use Im2mesh, please cite it as follows.

Ma, J., & Li, Y. (2025). Im2mesh: A MATLAB/Octave package for generating finite element mesh based on 2D multi-phase image (2.1.5). Zenodo. https://doi.org/10.5281/zenodo.14847059

Once my paper is published, I will update a new DOI here.

Acknowledgments

I sincerely thank Dr. Yang Lu for providing valuable advice. I also appreciate Dr. Darren Engwirda for the open-source mesh generator.

Other related projects

- voxelMesh (voxel-based mesh)