ImagePipes: Instructions

https://github.com/EarthSciTech/ImagePipes

Amirsaman Rezaeyan

ImagePipes: MicroCT Image Processing Pipelines

ImagePipes is a modular and extensible Python-based toolkit designed for processing, analyzing, and segmenting high-resolution 3D micro-computed tomography (µCT) images. Developed for porous media and geoscientific applications, ImagePipes enables reproducible workflows for transforming raw µCT image data into quantitative insights. The toolkit was developed and validated through real-world research projects at ETH Zürich, Eawag, and Empa.

1. Workflow Overview

The ImagePipes workflow transforms reconstructed µCT images into segmented, quantitative datasets through a structured series of steps, as visualized in the attached diagram (Figure 1). Each module is independent yet designed to integrate seamlessly for batch processing and reproducibility.

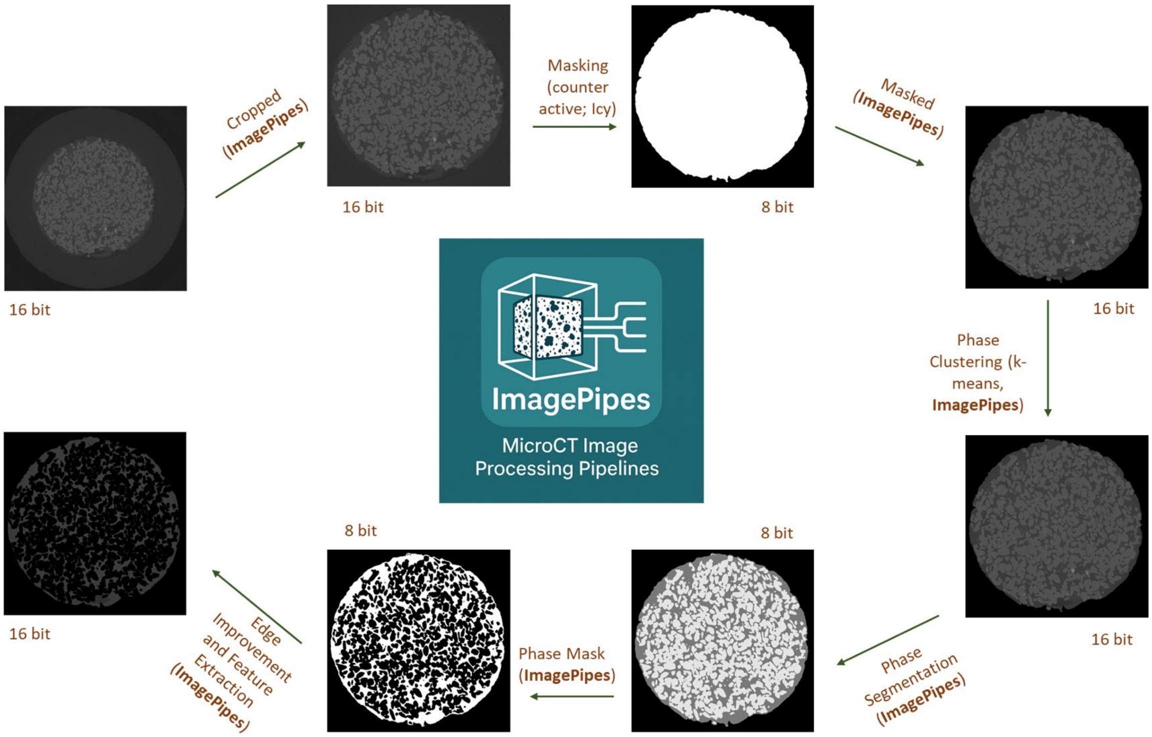

Figure 1. ImagePipes workflow.

Step 1: Image Cropping

The workflow begins with cropping the reconstructed μCT images to remove background and non-sample regions. Cropping is performed using ImagePipes' 'Crop' module, which maintains bit-depth (typically 16-bit).

Step 2: Masking

A binary mask is applied to isolate the region of interest (ROI). Masking is implemented using either ImagePipes or complementary software like ImageJ/Icy. The output is a binary 8-bit image defining the sample boundaries.

Step 3: Phase Clustering

K-Means clustering (scikit-learn) is used to classify image intensities into user-defined clusters. This step converts grayscale images (usually 16-bit) into labeled classes that represent different phases such as solid, liquid, and void.

Step 4: Phase Segmentation

Clustered images are post-processed to assign specific phase labels. The segmentation module includes hole-filling and interface correction using neighborhood filters to ensure topological continuity across the 3D stack.

Step 5: Edge Improvement
Segmentation boundaries are refined using 3D binary erosion and dilation techniques. This step minimizes noise near interfaces and ensures smoother binary phase boundaries while preserving geometric fidelity.

Step 6: Feature Extraction
Once phases are separated, quantitative metrics are extracted. These may include porosity, specific surface area, connectivity, and phase volume fractions. The extracted data can be directly used for transport modeling.

Step 7: Advanced Analyses (Optional)
Modules for further analyses—such as solute transport simulation, diffusion coefficient estimation, and tortuosity computation—extend ImagePipes capabilities into physical modeling workflows.

2. Core Functionalities

- Image Registration: Align misaligned or distorted image stacks using multi-resolution affine registration (Elastix).

- Phase Cross-Correlation Registration: Correct translation-only misalignments with phase cross-correlation.

- Noise Reduction: Apply 2D/3D Non-Local Means filters for denoising without losing structural information.

- Clustering & Segmentation: Perform k-means clustering and assign phase labels for pore, solid, and fluid regions.

- Edge Refinement: Apply 3D erosion-based morphological operations to refine phase boundaries.

- Conversion Tools: Map CT grayscale values to calibrated physical quantities (e.g., concentration, density).

- Mesh Generation: Create watertight 3D surface meshes (PLY format) using marching cubes.

- Resampling & Masking: Rescale and extract ROI from large tomogram stacks while preserving bit depth.

3. Use Cases

ImagePipes is optimized for μCT-based research and supports various applications, including:

- Reactive transport modeling and visualization
- Pore-scale flow and solute transport
- Multiphase CO_2 trapping and leakage studies
- Fracture-matrix interface characterization
- Sediment structure and grain morphology assessment

4. Software Stack

ImagePipes is built entirely in Python and leverages the following open-source libraries:

- NumPy, SciPy: Core numerical computation and filtering
- scikit-image: Image processing and morphology tools
- scikit-learn: Machine learning and clustering
- SimpleITK: Registration and format conversion
- OpenCV: Image input/output and visualization
- trimesh: Mesh generation
- tifffile: High-performance TIFF I/O
- rasterio: Geospatial image handling

5. Conclusion

ImagePipes provides a complete framework for transforming raw μCT images into segmented datasets and quantitative parameters relevant to porous media research. Its modular architecture supports reproducibility, scalability, and cross-disciplinary adaptation—from geosciences to materials science and beyond.

Author: Amirsaman Rezaeyan