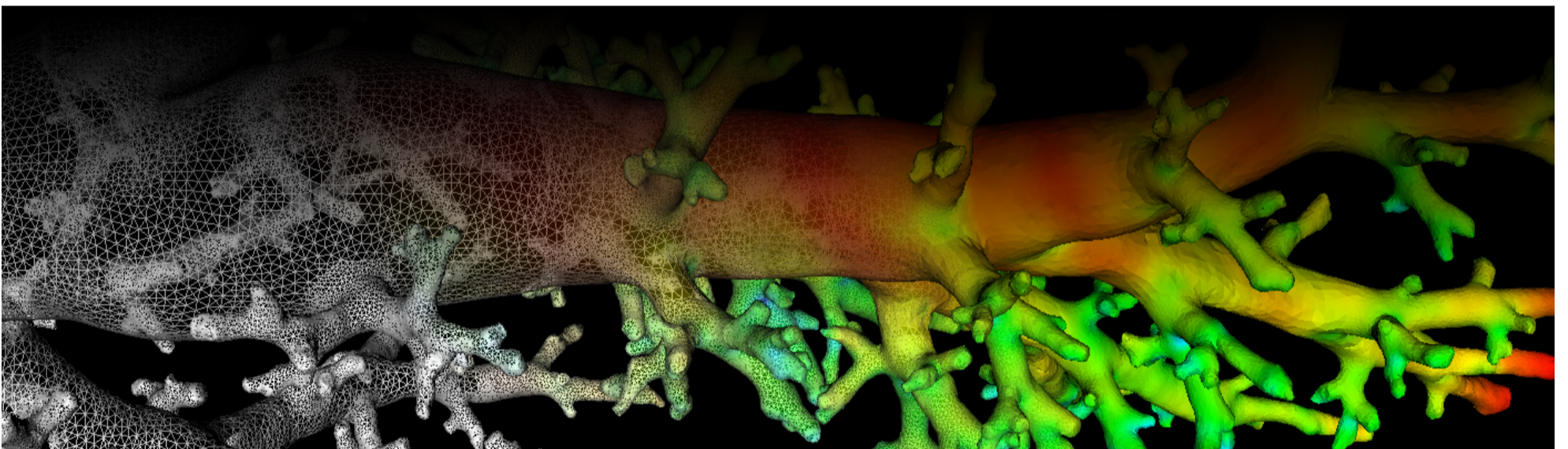


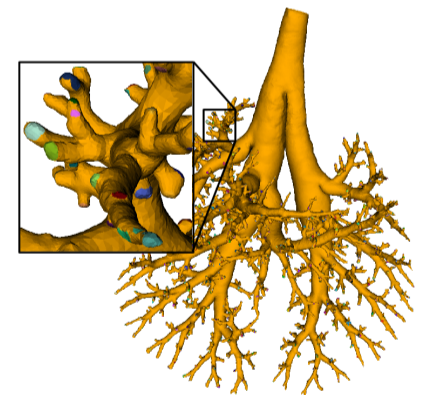
# Lung Anatomy + Particle Deposition (lapd) Mouse Archive for Modeling and Computational Toxicology



Mesh \*\_AirwayOutlets.vtk

Airway mesh with labeled outlet regions.

The airway tree and their walls are visible in the image data [\\*\\_Autofluorescent\\*.mha](#). Areas on the airway mesh where the bronchi transition into alveolar ducts do not have visible walls and form outlet regions ([Fig.1](#)). We segment and label these outlet regions ([Fig.2](#)). The resulting labelings are stored as label values assigned to the mesh's vertices.



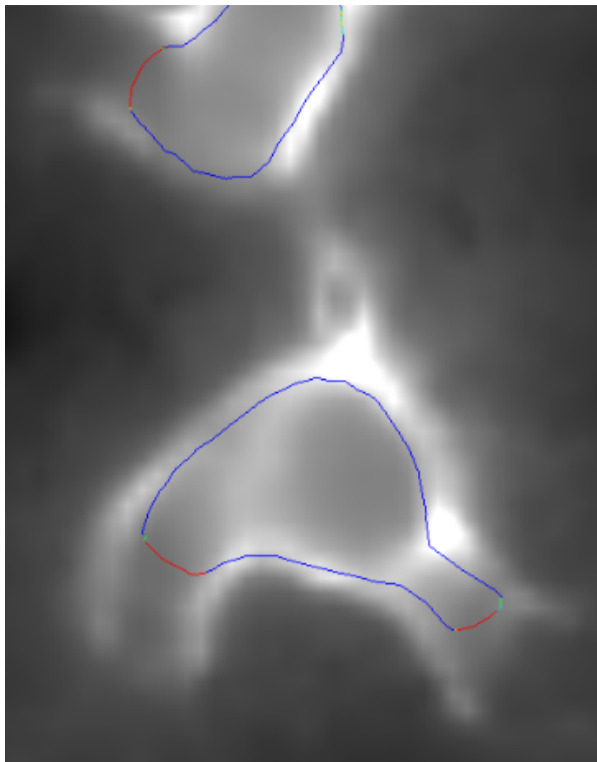


Fig. 1: Outlet regions in airway mesh.

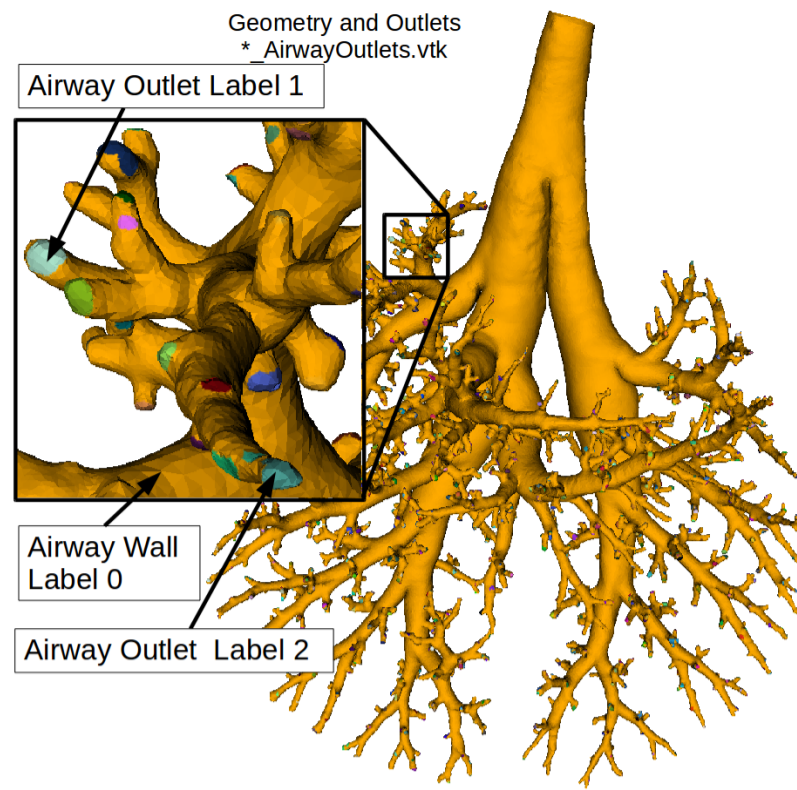


Fig. 2: Airway mesh with labeling of outlet regions.

Aerosol deposition measurements near the airway wall for each mesh vertex are available in [\\*\\_AirwayWallDeposition.vtk](#). Geometrically and topologically [\\*\\_AirwayOutlets.vtk](#) is identical to [\\*\\_AirwayWallDeposition.vtk](#) and [\\*\\_AirwaySegments.vtk](#).

## Code Example

This examples shows how to read and write a mesh such as [\\*\\_AirwayOutlets.vtk](#) using C++ and ITK.

[readWriteMesh.cpp](#) hosted with ❤ by [GitHub](#)

[view raw](#)

```

/*
Example how to read and write meshes used in lapdMouse project using ITK.

```bash
./readWriteMesh m01_AirwayOutlets.vtk out.vtk
```
*/

// ITK includes
#include <itkMesh.h>
#include <itkMeshFileReader.h>
#include <itkMeshFileWriter.h>

int main(int argc, char**argv)
{
    if (argc!=3)
    {
        std::cerr << "Usage: " << argv[0] << " input output" << std::endl;
        return -1;
    }

    // typedef for meshes used in lapdMouse project
    typedef itk::Mesh< float, 3 > MeshType;

```

## Related Data Structures

[\\*\\_AirwayWallDeposition.vtk](#) | [\\*\\_AirwaySegments.vtk](#)

## Related Code Examples

[readWriteMesh.cpp](#) | [mapOutlet2AirwaySegment.cpp](#)