BUILDING LOD-1

Supervisor: Vincent Chabannes

Introduction

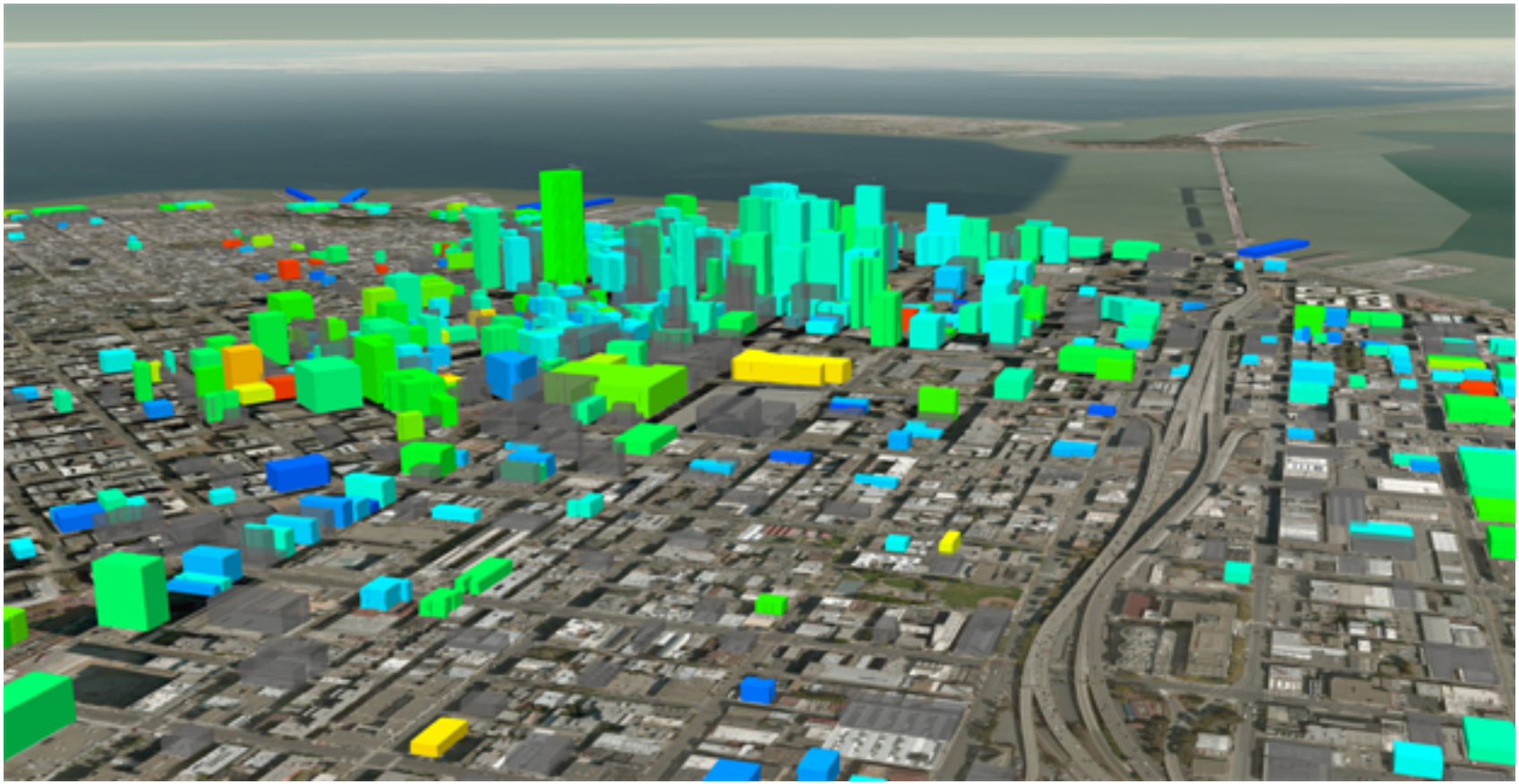
UFR de mathématique et d'informatique
Université de Strasbourg

IRMA

cemosis
Centre de modélisation et de simulation de Strasbourg

NumPEx

Context

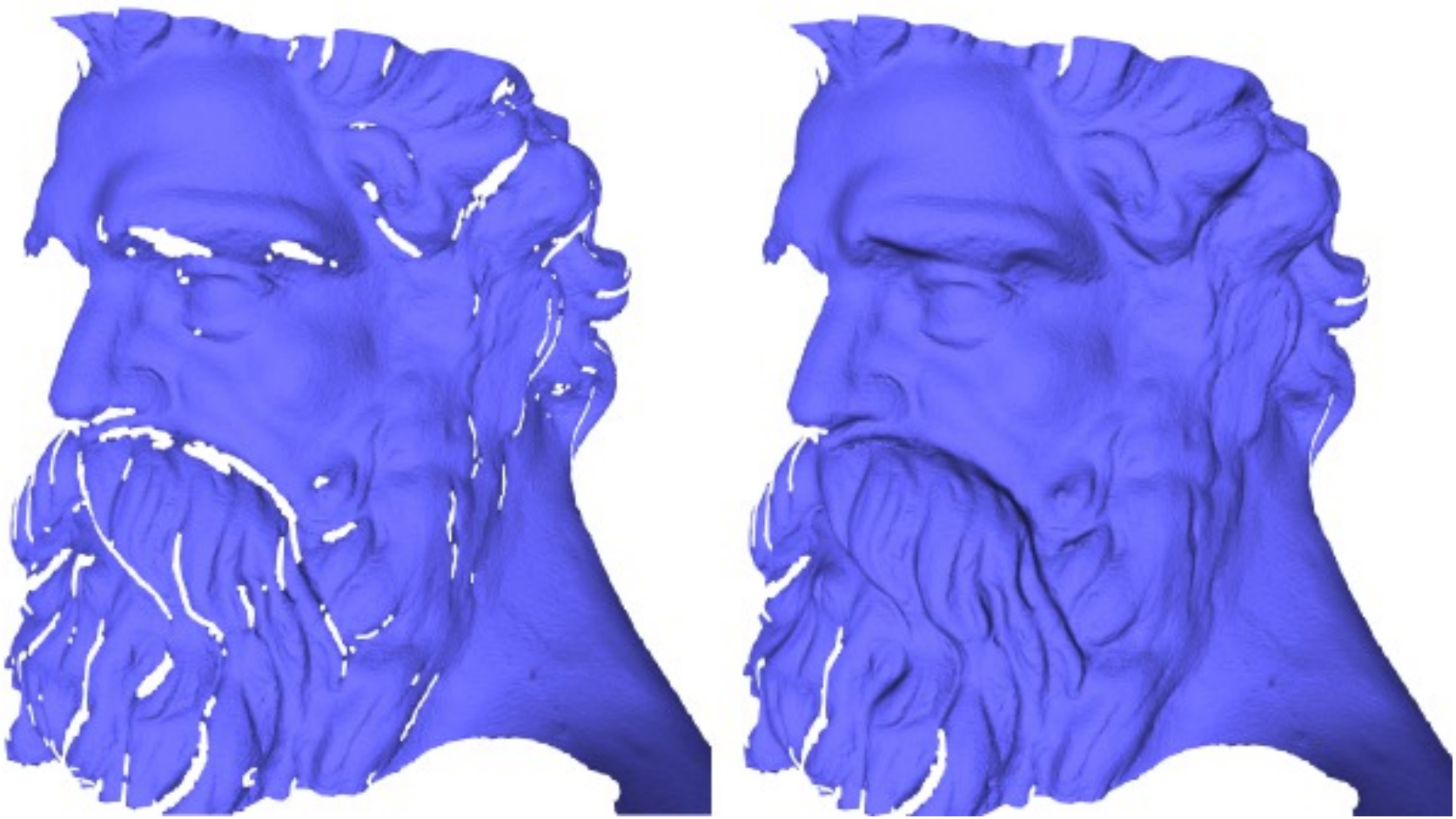


Energetic simulation on City

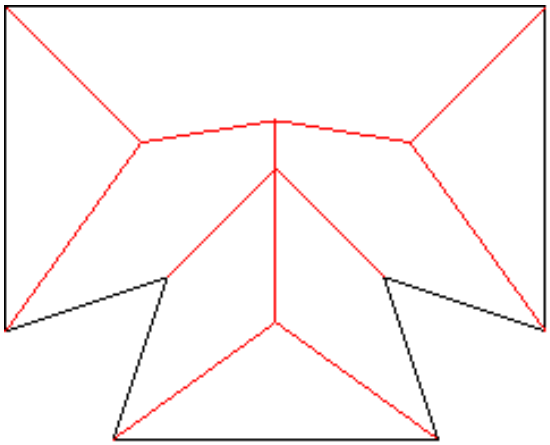
Objectives

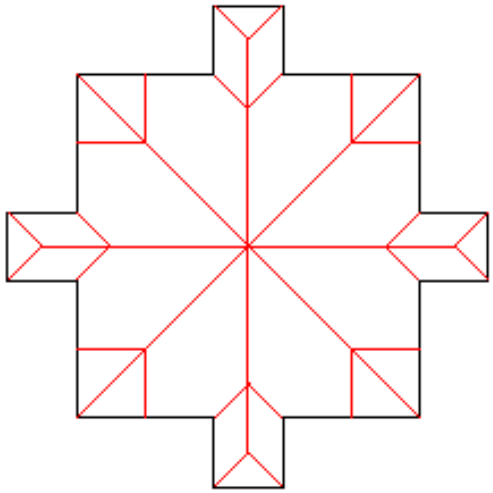
- Generation of a surface mesh representing the building's external envelope.
- Addition of internal walls delimiting exterior walls and any slabs for each floor.
- Roof modeling.
- Creation of a 3D building volume mesh (structure and/or indoor air).
- Mesh adaptation: quality control of the produced meshes.
- Complexifying building shape.

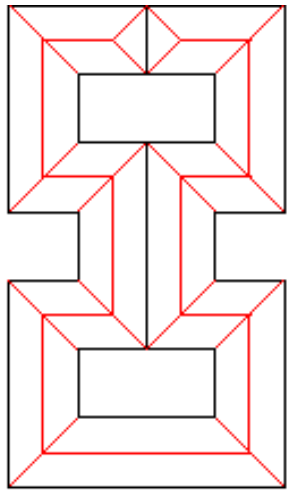
Polygon Mesh Processing



Straight Skeleton

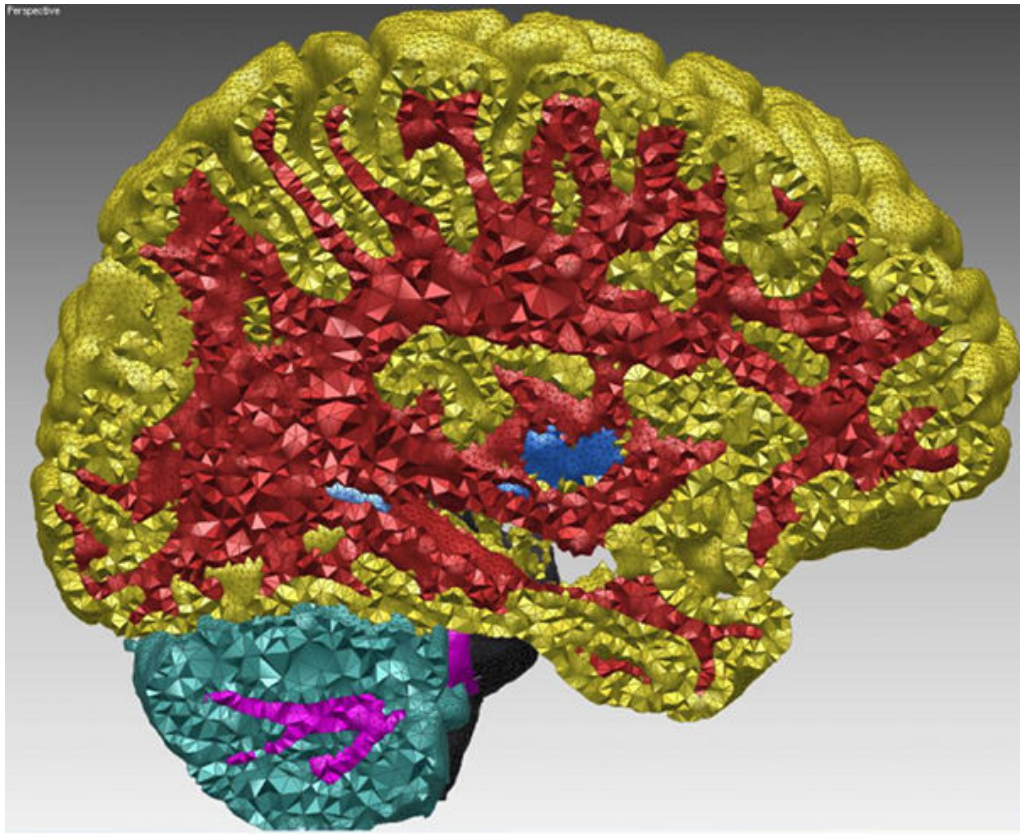






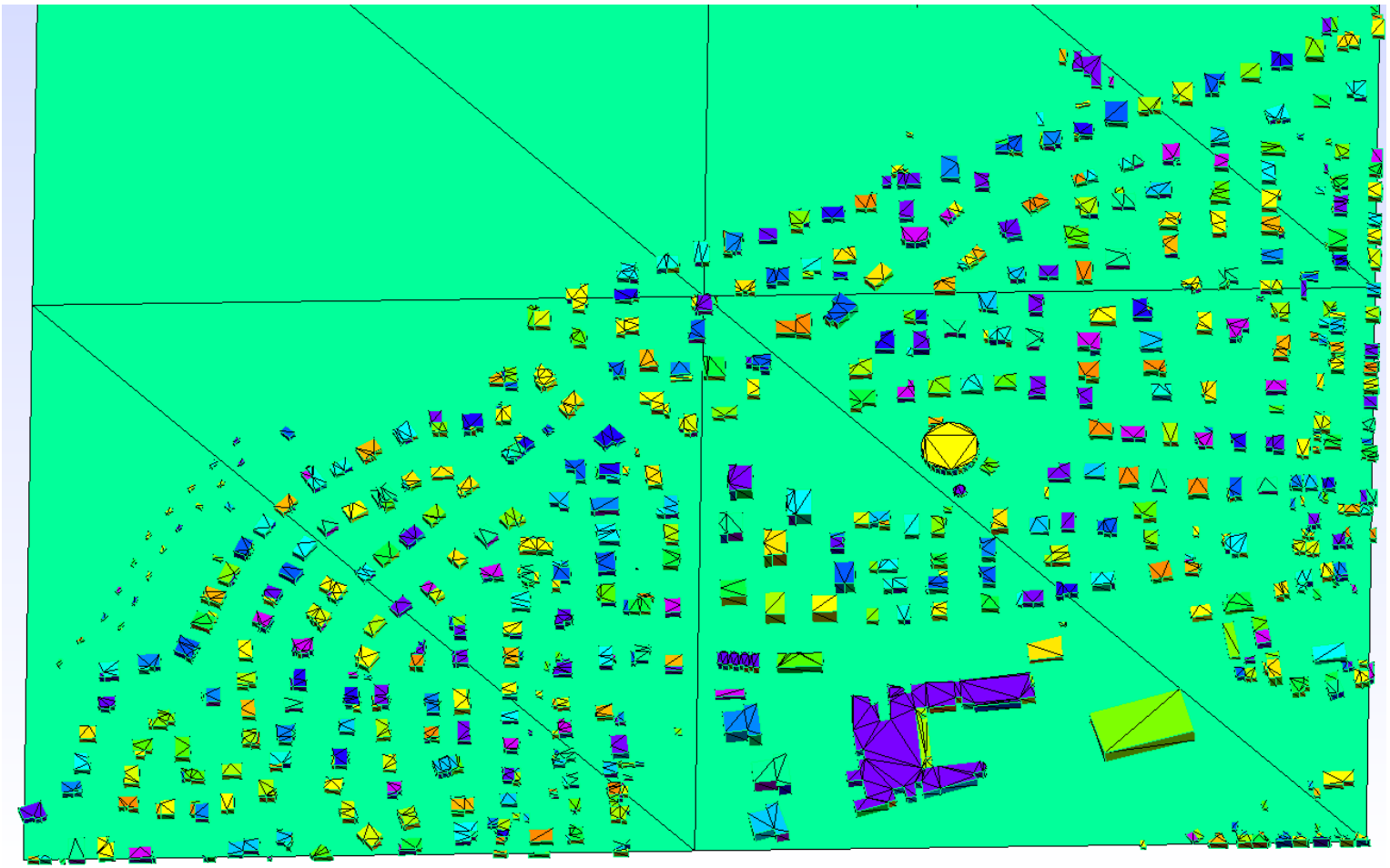
simple skeleton

3D Mesh Generation



Multilabel Mesh

Data



schiltigheim district

Workspace Methodology

Collaborative Development Environment:

- **Version Control:** We use GitHub as a shared repository.
- **Team Structure:** 4 interns collaborate on the same repository with a different goal.
- **Issue Tracking:** Work is organized by creating issues, each associated with a specific branch.
- **Test:** To avoid creation of bug we implement test.
- **Pull Requests:**
  - Development is done by submitting pull requests for new features and bug fixes.
  - Code reviews ensure quality and catch potential conflicts.

Methodology

Building Creation Steps:

- **Surface Mesh:** Creation of the surface mesh of every building
- **Building Merging:** merge every building in one object
- **Volume Meshing:** Create a volume mesh from a surface mesh
- **Qualtiy check:** Check the proprietes of the final object

MultipleBuilding Class

MultipleBuilding main method;

- **Data loading method:** The 'loadFromJsonGis' and 'loadFromJsonDat' method load the data needed for the building creation.3
- **Building creation method:**
  - **extrudeWalls**
  - **addFloors**
  - **generateRoof**
  - **mergeWallRoof**
  - **mergeAllMeshes**
  - **meshing3D**

Parameters

MultipleBuilding Parameters:

- **Input File:** This option is to input gis or dat file for the building creation
- **Debug:** This option enable everythings that permit debug
- **Roof:** This option enable the creation of the roof of buildings
- **Union:** This option change the merge of floor and roof to a volume union
- **Volume:** This option enable the volume meshing of the resulted surface mesh

Volume Mesh

Functionality Implemented:

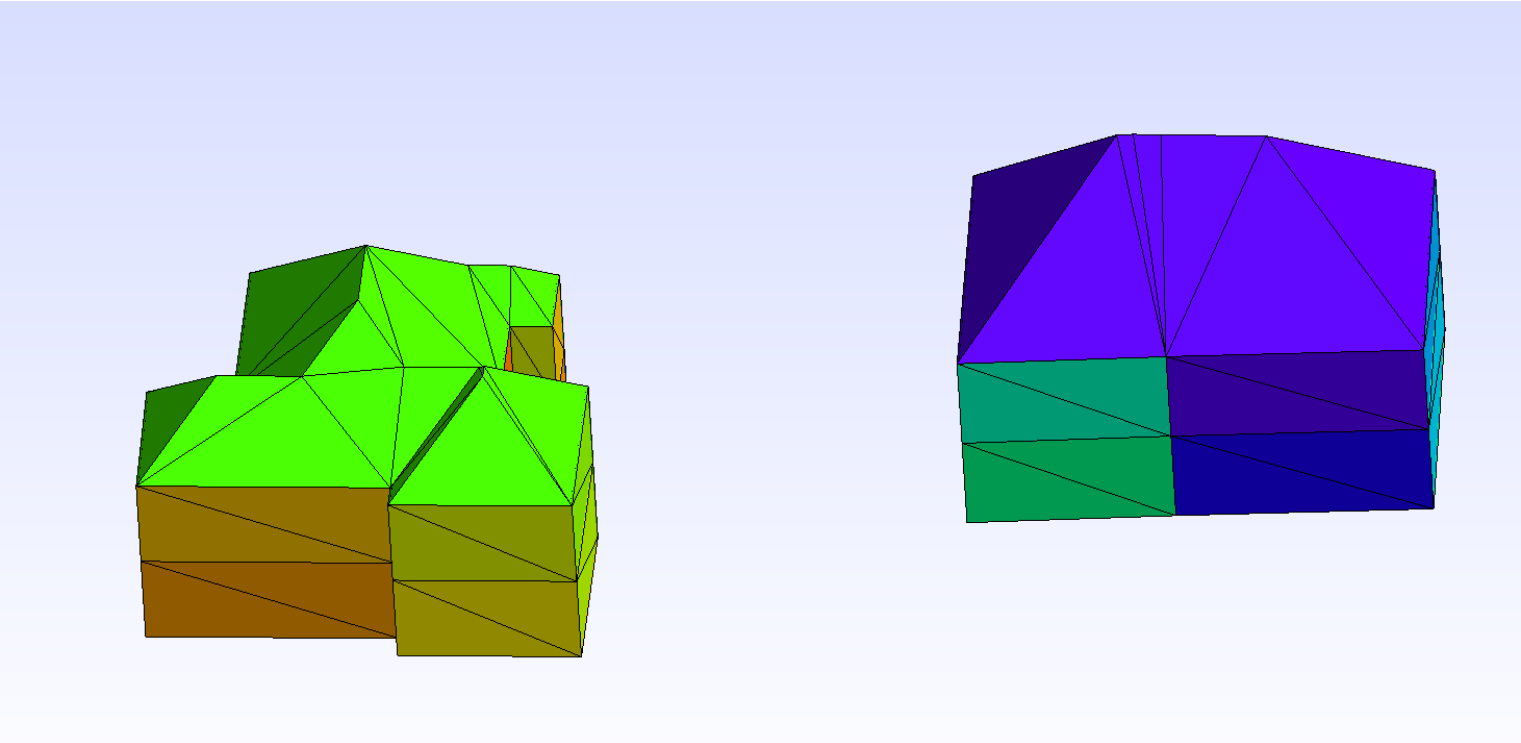
- **meshToPolyhedron:** This function convert ktirio mesh to a cgal polyhedron.
- **polyhedronTo3DMesh:** This function convert polyhedron to a volume mesh.
- **mesh_3d:** This function convert a surface mesh to a volume mesh.

Test

Test Implemented:

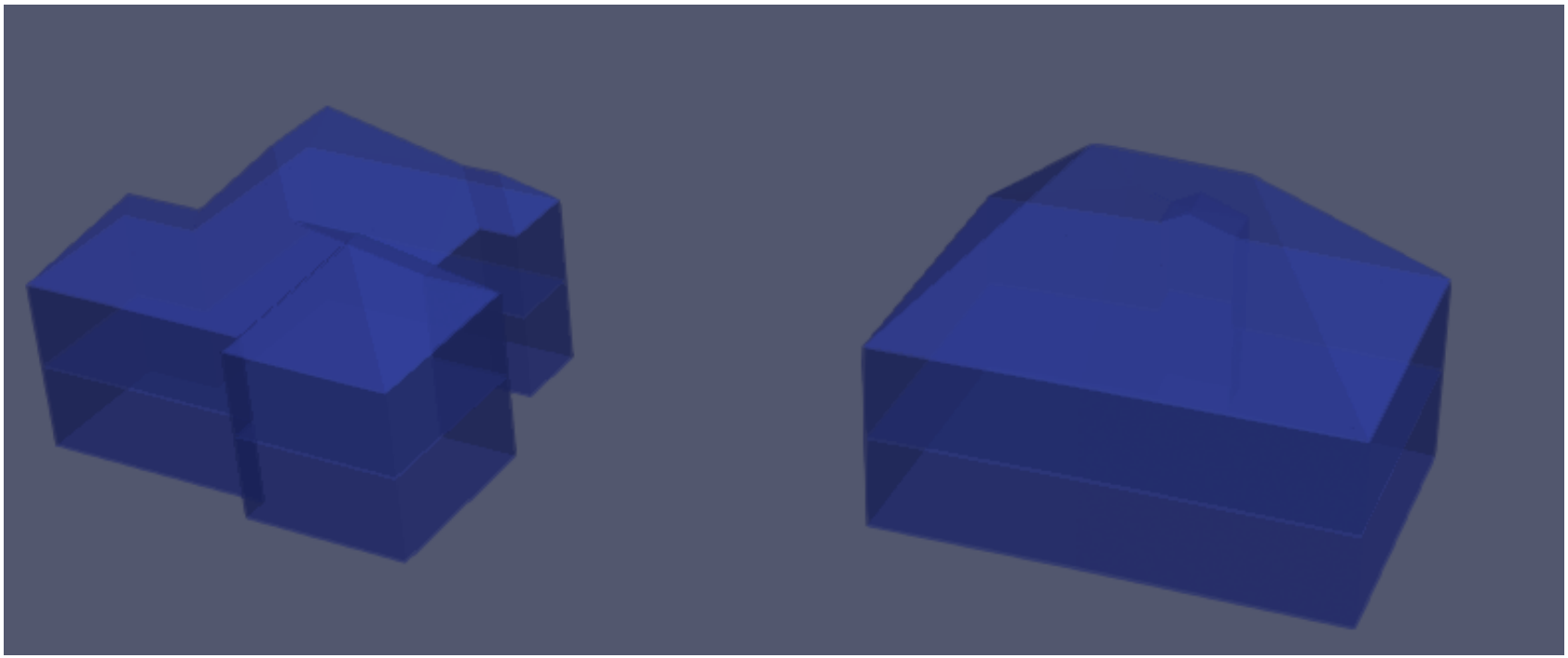
- **Marker:** This test verify the handling of markers within meshes.
- **Polyhedron:** This test verify the creation of a valid polyhedron from a mesh.

Surface Mesh

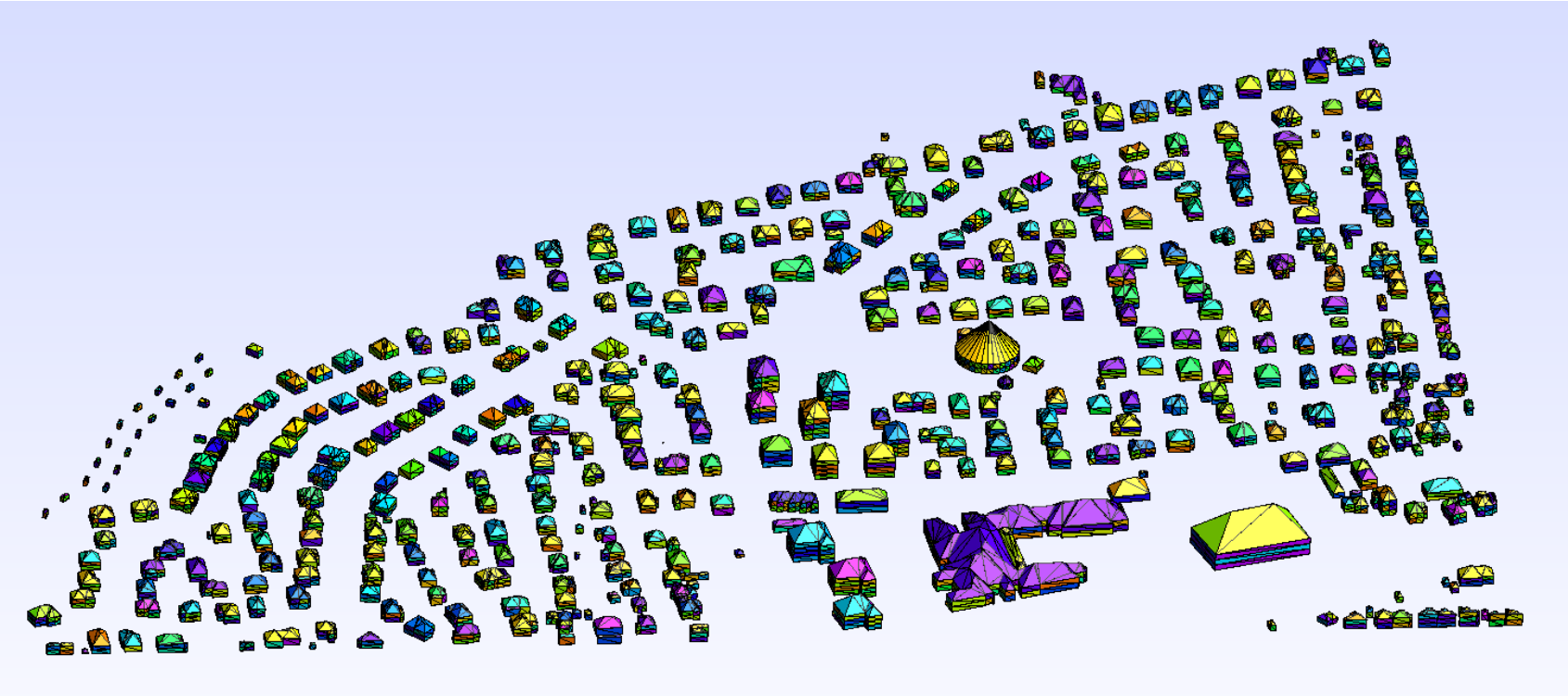


Surface Mesh of 2 buildings

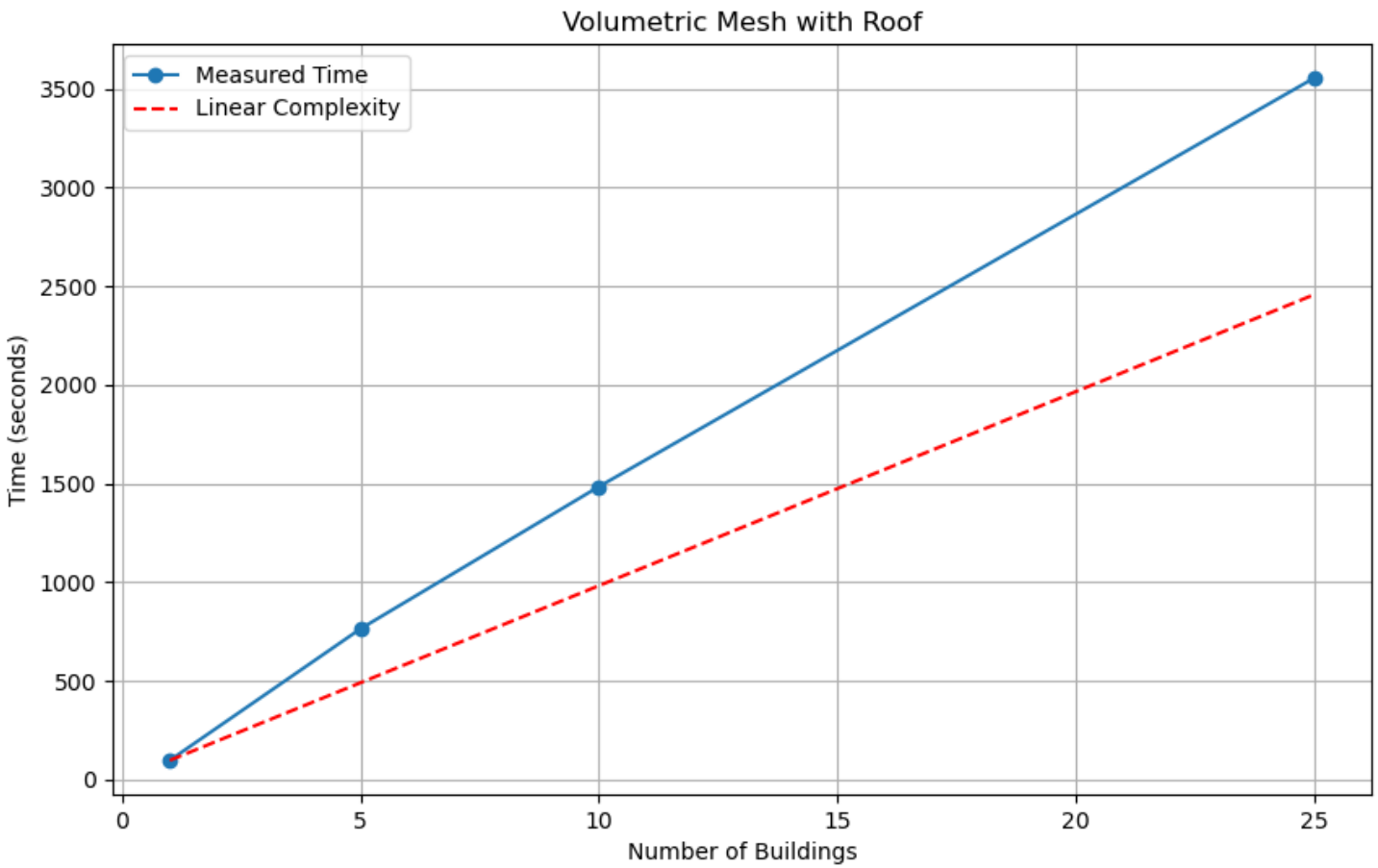
Volume Mesh



Surface Mesh of district



Complexity analysis



Perspectives

- **Enhanced Features:** Expand ‘MultipleBuilding‘ class capabilities to support complex structures and custom elements.
- **Optimization:** Improve mesh processing efficiency, especially for large models.
- **Volume Meshing Tests:** Develop tests to ensure accurate and efficient 3D mesh generation.
- **Improving Floor Creation:** Enhance flexibility in floor creation with custom heights and different floor shape.
- **Feel++ Integration:** Use ‘Feel++‘ for extensive mesh validation.

Conclusion

- **Project Goal:** Developed a geometric reconstruction tool for LOD1 buildings, aiding energy simulations and urban planning.
- **Key Contributions:** Implemented the ‘MultipleBuilding‘ class, converted meshes to CGAL polyhedrons, and generated 3D volume meshes.
- **Collaboration:** Utilized GitHub for structured workflow, focusing on issues, branches, and pull requests.
- **Testing:** Ensured robustness through rigorous testing and validation of new features.
- **Future Perspectives:** Expand features, optimize performance, and integrate advanced tools to further improve the Ktirio library.

reference I


The CGAL Project.
CGAL User and Reference Manual.
CGAL Editorial Board, 5.6.1 edition, 2024.


Mulin Yu and Florent Lafarge.
Finding Good Configurations of Planar Primitives in Unorganized Point Clouds.
In *CVPR 2022 - IEEE Conference on Computer Vision and Pattern Recognition*, La Nouvelle-Orléans, United States, June 2022.


Ktirio - construction et immobilier.


Feelpp. open-source c++ library.


Meshlab. open source system for processing and editing 3d triangular meshes.

reference II

- Irma. institute for advanced mathematical research.
- Hidalgo2. hpc and big data technologies for global challenges.
- Cemosis. innovation through modeling, simulation, optimisation and high performance computing.
- Numpex. french program dedicated to exascale.
- Paraview. open source post-processing visualization engine.
- Json , file format.
- Modeling energy efficiency block by block | energy technologies area.