Introducing Tools for Storing, Rendering and Annotating Triangle Meshes in ROS and RViz

Mesh Tools

Sebastian Pütz
Osnabrück University
spuetz@uos.de

Thomas Wiemann
Osnabrück University
twiemann@uos.de
Mesh Applications and Advantages

- 3D Mapping of outdoor terrain
- Obtaining surface information
- Multilayer surface information
- Multilevel environments
- Visualization of terrain analysis
- Robot Mesh navigation planning
- Visualization of textures
- Data reduction
- Intuitive annotation of objects
Mesh Tools – Packages

- mesh_msgs
- mesh_msgs_transform
- mesh_msgs_hdf5
- hdf5_map_io
- label_manager
- rviz_mesh_plugin
- rviz_map_plugin
Mesh Geometry

# MeshGeometry Message
geometry_msgs/Point[] vertices
geometry_msgs/Point[] vertex_normals
mesh_msgs/TriangleIndices[] faces

# TriangleIndices Message
uint32[3] vertex_indices

# MeshGeometryStamped Message
std_msgs/Header header
string uuid
mesh_msgs/MeshGeometry mesh_geometry

Vertex Buffer

\[
\begin{array}{ccccc}
V_1 & V_2 & V_3 & V_4 & V_5 \\
1 & 2 & 3 & 4 & 5 \\
\end{array}
\]

Index Buffer

\[
\begin{array}{ccccc}
1 & 2 & 3 & 2 & 5 \\
F_1 & F_2 & F_3 \\
\end{array}
\]
Mesh Vertex Costs and Colors

# MeshVertexCosts
float32[] costs

# MeshVertexCosts Stamped
std_msgs/Header header
string uuid # reference to the mesh
string type # costs type
mesh_msgs/MeshVertexCosts
  mesh_vertex_costs

MeshVertexColors and labeled objects on the Ford Campus Dataset
Mesh Navigation

- Reconstruct sensor data to a mesh
- Terrain analysis on the mesh to cost layers
- Combine cost layers to a navigation layer
- Perform path planning and motion control on the mesh using Move Base Flex

Mesh Face Cluster

# MeshFaceCluster
uint32[] face_indices
# label (optional)
string label

# MeshFaceClusterStamped Message
std_msgs/Header header
string uuid
MeshFaceCluster cluster
# overwrite existing labeled faces
bool override
Mesh Materials & Textures

```python
# MeshTexture
string uuid
uint32 texture_index
sensor_msgs/Image image

# MeshMaterial
uint32 texture_index
std_msgs/ColorRGBA color
bool has_texture

# MeshVertexTexCoords
float32 u
float32 v

# MeshMaterials
mesh_msgs/MeshFaceCluster[] clusters
mesh_msgs/MeshMaterial[] materials
uint32[] cluster_materials
mesh_msgs/MeshVertexTexCoords[] vertex_tex_coords

# MeshMaterialsStamped
std_msgs/Header header
string uuid
mesh_msgs/MeshMaterials mesh_materials
```
Hierarchical Data Format IO

- Representation of complex data structures and metadata.
- Portable file format
- No limits (number or size of data)
- C, C++, Python, Fortran 90, and Java interfaces.
- Storing pre-computed mesh layers, e.g., roughness, height differences, and semantic information
- Storing textures like, RGB images or even textures generated by a hyper spectral camera.

github.com/uos/mesh_tools
wiki.ros.org/mesh_tools