Development and Evaluation of Point Cloud Compression for the Point Cloud Library

Julius Kammerl
Institute for Media Technology, TUM, Germany

May 12, 2011
Goals/Motivation:

▶ Efficient for real-time processing
▶ General compression approach for unstructured point clouds (varying size, resolution, density, point ordering)
▶ Exploit spatial sparseness of point clouds
▶ Exploit temporal redundancies in point cloud streams
▶ Keep introduced coding distortion below sensor noise
Hierarchical tree data structures can efficiently describe sparse 3D information

Focus on real-time compression favors octree-based point cloud compression approach

Octree structures enable fast spatial decomposition
Octree-based Encoding

- Root node describes a cubic bounding box which encapsulates all points
- Child nodes recursively subdivide point space
- Nodes have up to eight children ⇒ Byte encoding
- Point encoding by serializing high-resolution octree structures!
Temporally adjacent point clouds often strongly correlate:

 Serialized Octree A:
00000100 01000001 00011000 00100000
00000100
01000001
00011000 00100000

Differentially encode octree structures using XOR:

 XOR Encoded Octree B:
00000000 00000011 00000000 00000010

Gain: reduced entropy of the serialized binary data!
Experimental results of octree-based point cloud compression
Data rate comparison between regular octree compression (gray) and differential octree compression (black) for 1 mm$^3$ resolution
Real-time spatial change detection based on XOR comparison of octree structure
Challenge: With increased octree resolution, complexity grows exponentially

Solution: Limit octree resolution and encode point detail coefficients

Enables trade-off between complexity and compression performance

Also applicable to point components (color, normals, etc.)
Compression Pipeline

Encoding Pipeline:

Point Cloud → Octree Structure → Point Component Encoding → Entropy Encoding → Compressed PC

Point Detail Encoding → Position Detail Coefficients

Decoding Pipeline:

Compressed PC → Entropy Decoding → Octree Structure → Point Cloud

Point Detail Decoding → Point Component Decoding
Experimental results for point detail encoding at octree resolution of $9mm^3$.

Enables fast real-time encoding with high point precision.
Point cloud compression demo
Contributions to PCL

Contributed algorithms using octrees for spatial decomposition:

- Point cloud compression
- Fast search operations (neighbor search, radius search, voxel search)
- Voxel-grid filter / downsampling
- Voxel-centroid filter / downsampling
- Point density analysis
- Spatial 3D change detection
- Occupancy checks/maps
Further PCL contributions:

- Fast range coder implementation (fixed-point version of arithmetic entropy coder) with static and adaptive frequency tables.
- Fast neighbor search techniques for organized point clouds (with Nico)
- Octree viewer implementation (with Nico)
- Helping out at the PCL front
Thank you!