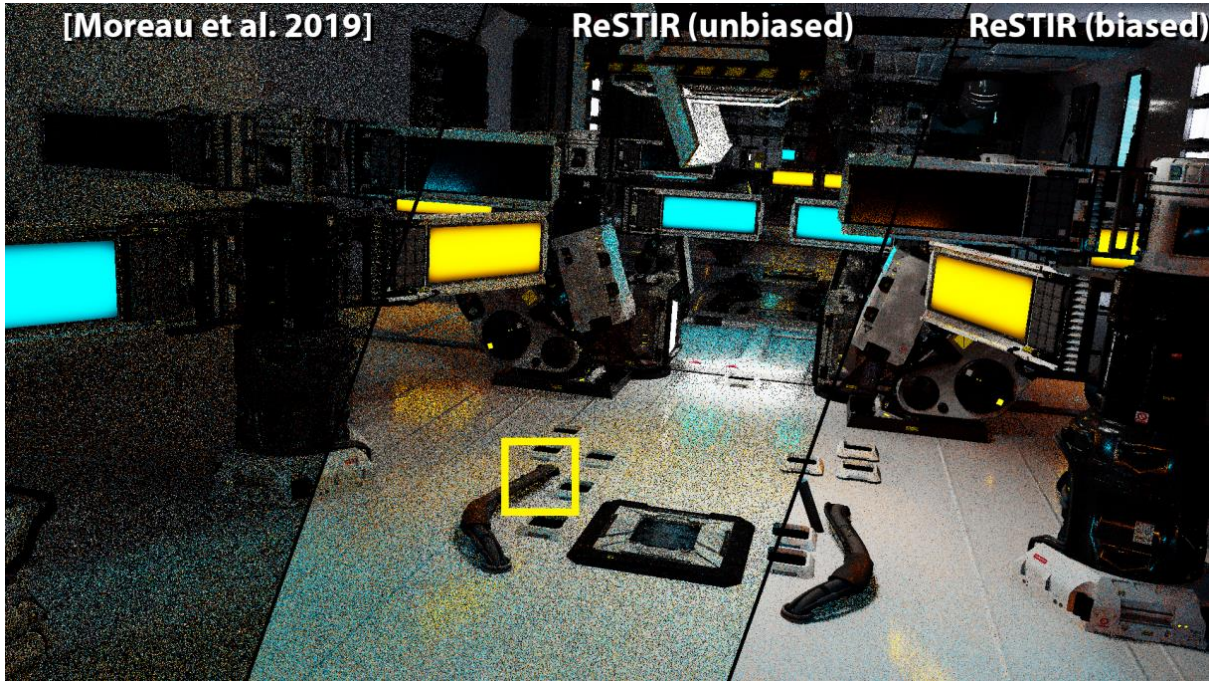
Real-Time Fluid Simulation (A, 2)

- **Simulate smoke by a more accurate advection solver**
  - Implement the stable fluids solver
  - Modify the advection by the reflection solver
  - Implement them on GPU using CUDA
  - GPU-based volume renderer of velocity field
- **Reference**
  - https://dl.acm.org/doi/pdf/10.1145/311535.311548
  - https://jzehnder.me/publications/advectionReflection/

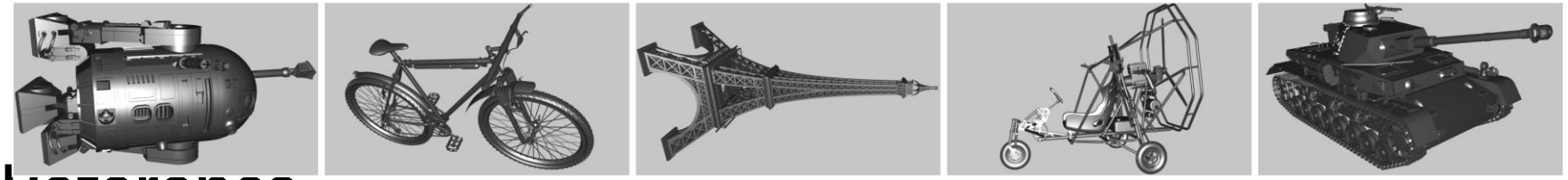
Reservoir Spatio-Temporal Importance Resampling (A, 2)

- **Rendering surface with ReSTIR efficiently**
  - Efficient sample many lights.
  - Reuse samples from other pixels.
  - Probably cross-sample instead of temporal.
- **Reference**
  - [2020] Spatiotemporal reservoir resampling for real-time ray tracing
  - [2021] ReSTIR GI Path Resampling for Real-Time Path Tracing



Ray Tracing NURBS Surface (A, 2)

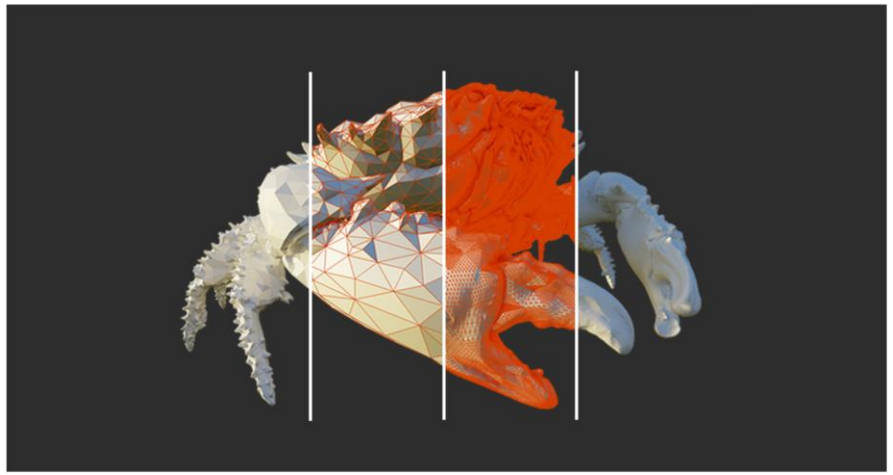
- **Rendering continues NURBS surface directly**
  - No meshing is required for rendering
  - Implement ray-surface intersection algorithm
  - Integrate into the ray-tracing framework with global illumination



- **Reference**
  - https://www.mattkeeter.com/projects/mrep/

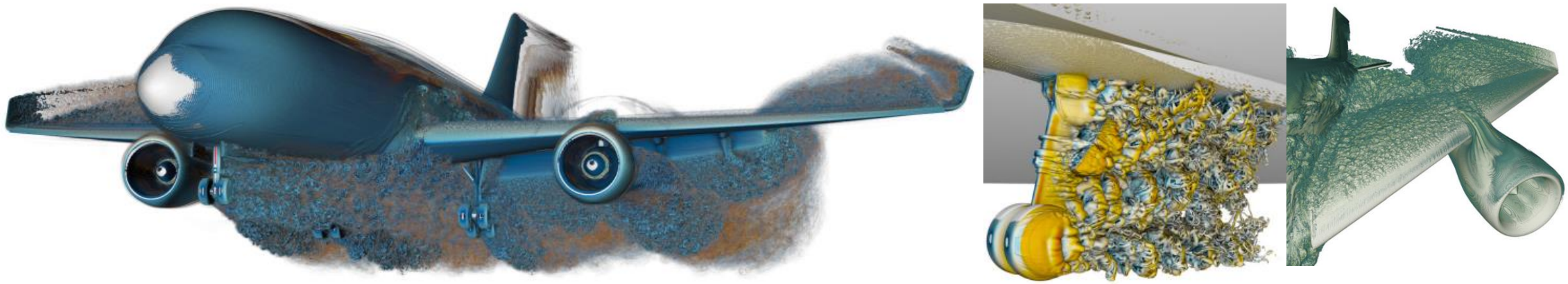
Render Intricate Geometries (A, 2)

- **Displacement BVH**
  - Implement building algorithms for displacement BVH
  - Implement traversal algorithms for displacement BVH
- **Reference**
  - [2018] A High-Resolution Compression Scheme for Ray Tracing Subdivision Surfaces with Displacement
  - [2021] Tessellation-Free Displacement Mapping for Ray Tracing



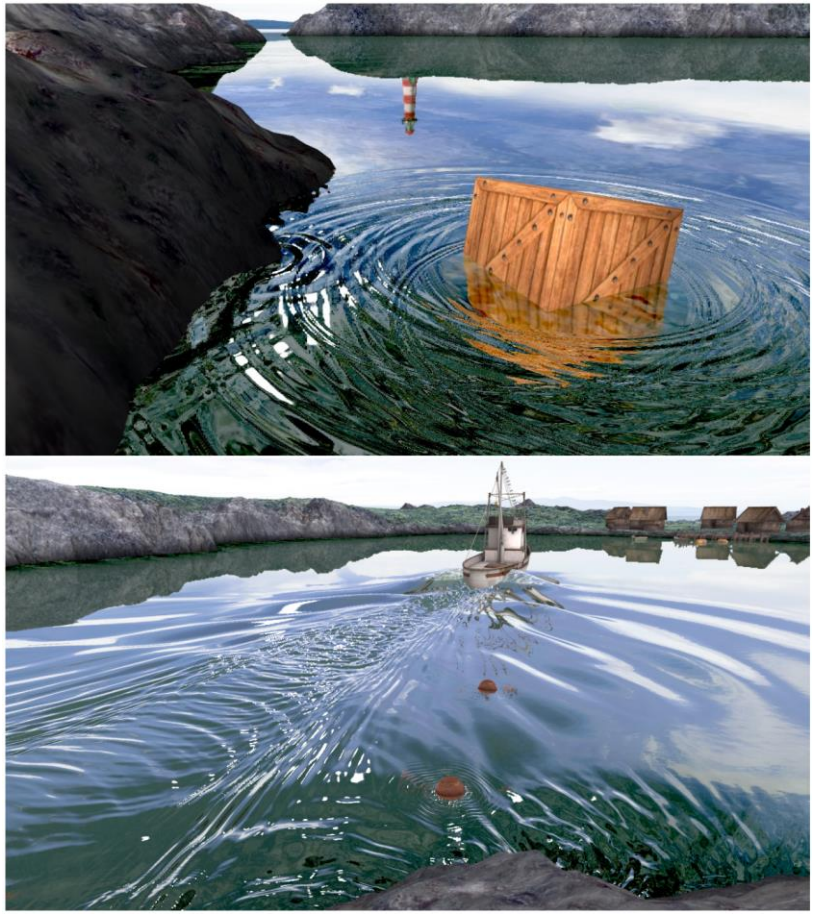
Multi-Resolution Isosurface Rendering (A, 2)

- **Rendering isosurface with ray tracing based techniques**
  - Build k-d trees for isosurfaces encompassed in discrete volume data
  - Design transfer functions for isosurface rendering
  - Pre-integrated transfer function
- **Reference**
  - [2021] Ray Tracing Structured AMR Data Using ExaBricks



Advanced Water Wave (A, 1)

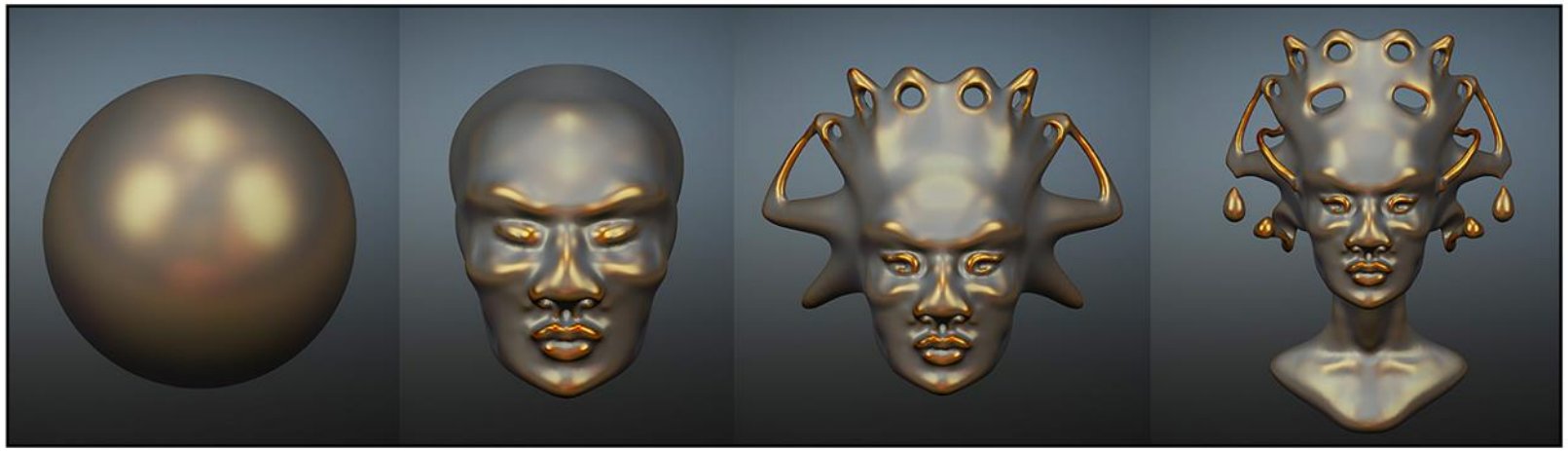
- **Advanced water wave animation**
  - Implement advanced wave particles or procedural methods to solve the wave equation
  - Render the water surface for example with OpenGL and texture
  - Real-time required
- **Reference**
  - [2007] Wave Particles
  - [2017] Water Wave Packets
  - [2018] Water Surface Wavelets



Interactive Sculpting (A, 1)

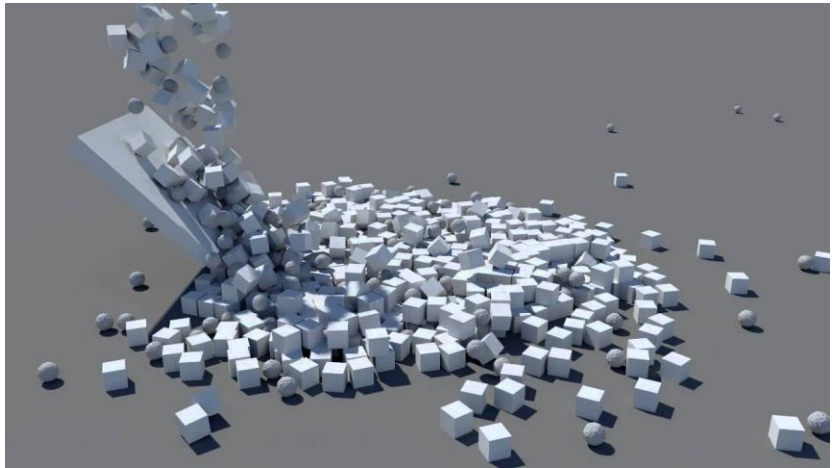
- Interactive
- sculpture visual tool
- Dynamic
- changes of mesh structure

- Reference:
  - Stanculescu L, Chaine R, Cani M P. Freestyle: Sculpting
  - meshes with self-adaptive topology[J].



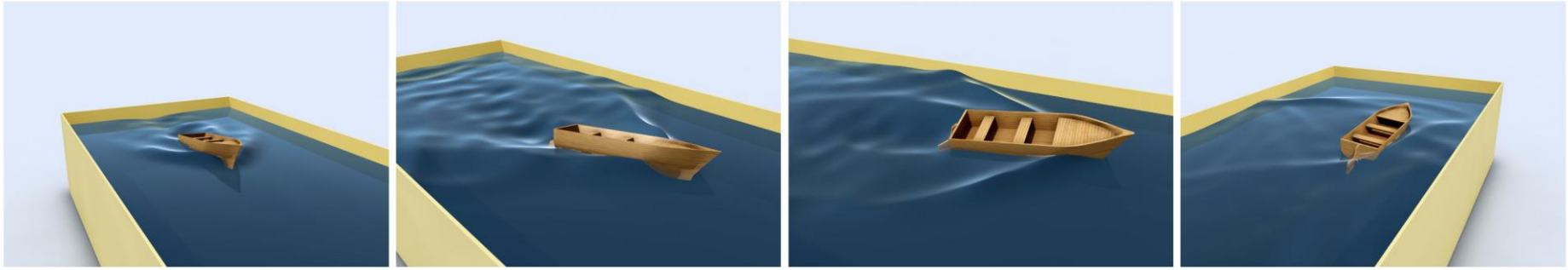
Massive Rigid-Body Simulation (B, 2)

- **Implement the massive rigid body simulation**
  - Basic transformation: translation + rotation over time
  - Collision detection
  - Handling colliding and resting contacts
- **Reference**:
  - Iterative Dynamics with Temporal Coherence
  - Game Physics in One Weekend



Water Wave (B, 2)

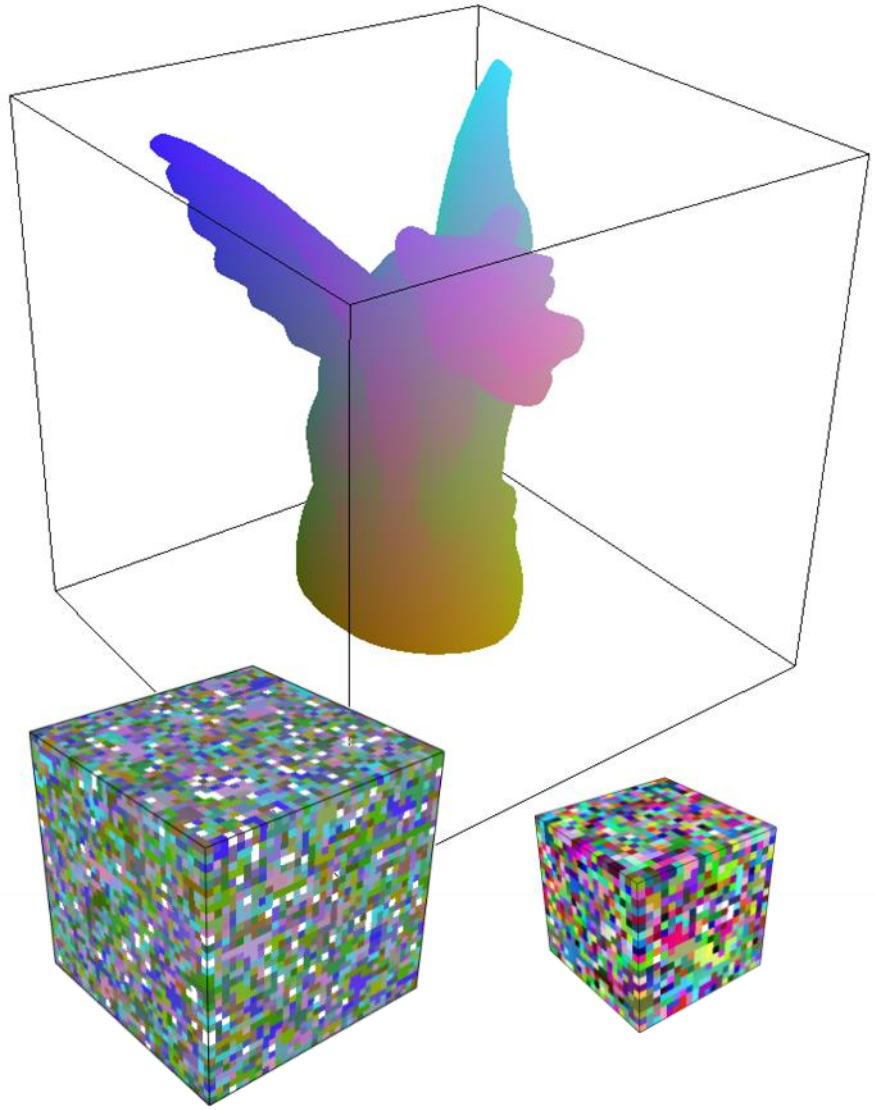
- **Simple water wave animation**
  - Implement wave particles or convolution methods to solve the wave equation
  - Render the water surface for example with OpenGL and texture



- **Reference**
  - [2007] Wave Particles
  - [2016] Dispersion Kernels for Water Wave Simulation

Sparse Volume Visualization (B, 2)

- **Rendering a sparse volume with advanced**
- **data structure**
  - Implement hash table, BVH, K-d Trees or
  - other data structures or other accelerating techs.
  - The faster the better.
  - Real-time may get A
- **Reference**
  - [2009] Real-Time Parallel Hashing on the
  - GPU
  - [2022] Sparse Volume Rendering using
  - Hardware Ray Tracing and Block Walking
  - [2011] Coherent Parallel Hashing
  - [2006] Perfect Spatial Hashing

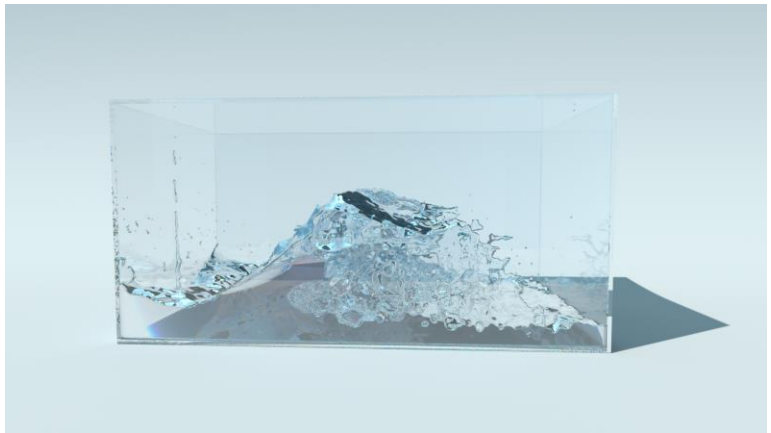


Photon Mapping (B, 1)

- **Create an offline renderer using photon mapping techniques**
  - Capable of handling caustics from light refraction through transparent substances.
  - Accelerate photon-finding with k-d tree.
  - Solid derivation on the math behind.
- **Reference**
  - [2009] Stochastic progressive photon mapping

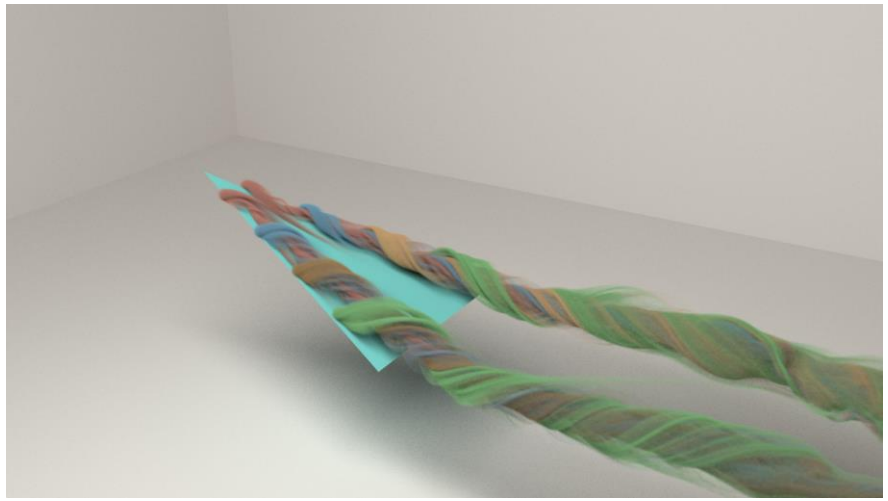
Water Surface Rendering (B, 1)

- **A offline renderer with water surface&volume rendering capability**
  - Should handle intricate water geometry.
  - Handle physically-based homogeneous volume inside the water surface.
  - Can reproduce the caustics effect efficiently (select your approach).
- **Reference (not that paper-oriented)**
  - https://rgl.s3.eu-central-1.amazonaws.com/media/papers/Jakob2016Path.pdf
  - https://cseweb.ucsd.edu/~ravir/dilution.pdf



Realistic Smoke Rendering (B, 1)

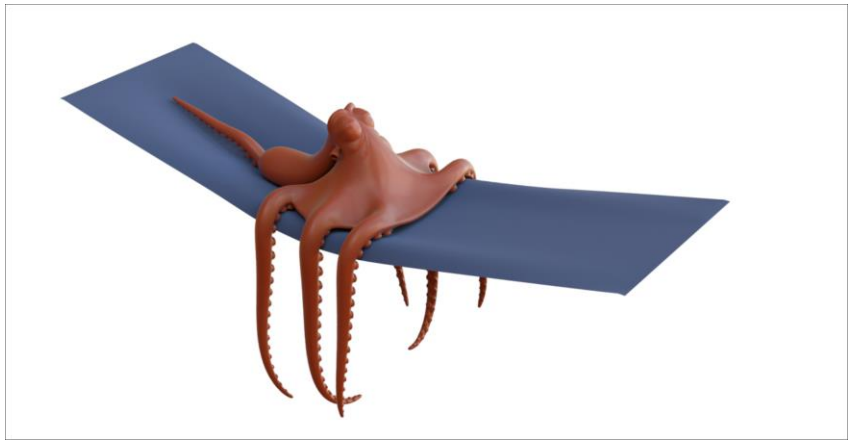
- **Rendering smoke by volumetric techniques**
  - Volume data saved in VDB files
  - Multiple scattering
  - Shadow effects
  - Combined with surface rendering
- **Reference**
  - Paper: [2017] Spectral and Decomposition Tracking for Rendering Heterogeneous
  - Paper: [2018] Monte Carlo methods for volumetric light transport simulation
  - Library: OpenVDB, NanoVDB.



Deformable Solids Simulation (B, 1)

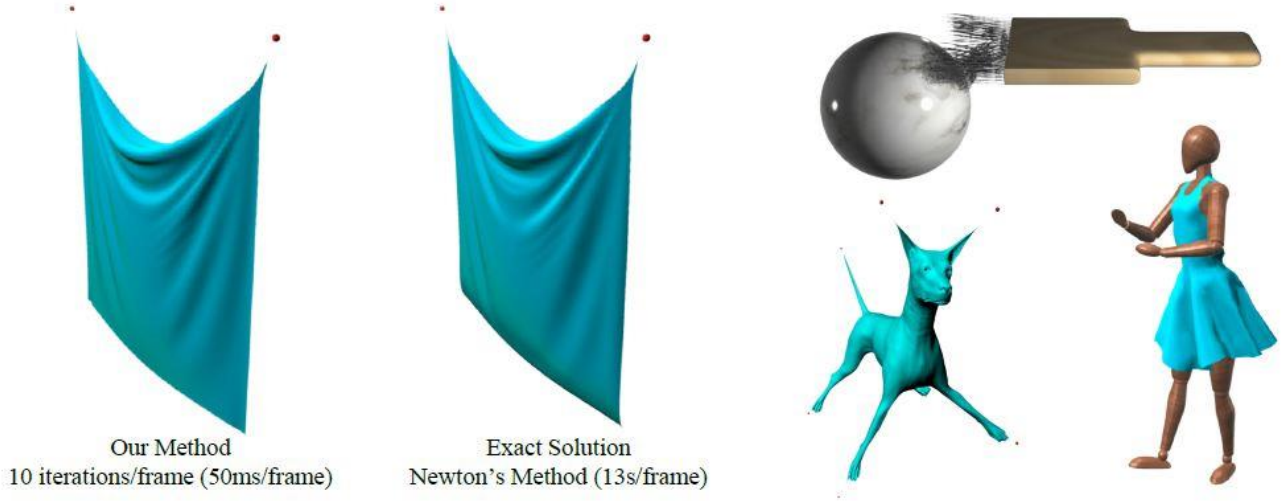
- **Deformable body simulation**
  - Implement a deformable body simulation without collision.
  - Multiple energy models and their comparisons.
  - (Preferably) Use implicit euler for time integration.

- **References**
  - https://graphics.pixar.com/library/DynamicDeformablesSiggraph2020/paper.pdf
  - https://viterbi-web.usc.edu/~jbarbic/femdefo/barbic-courseNotes-modelReduction.pdf



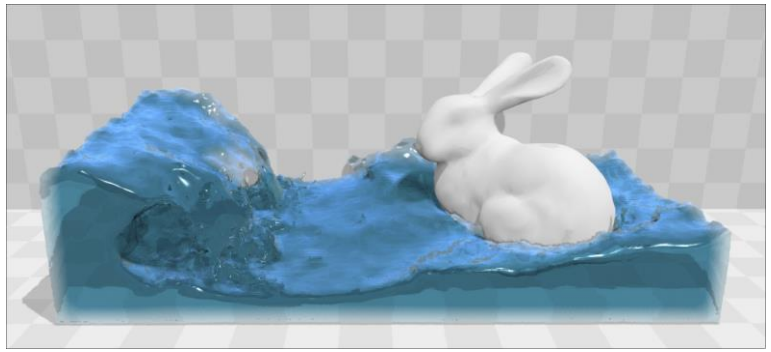
Cloth Simulation with Mass-Spring Model (B, 1)

- **Fast Simulation of Mass-Spring System**
  - Implementation of the fast non-linear cloth simulation solver.
  - Possibly GPU parallel implementation with CUDA.
- **Reference**
  - Fast Simulation of Mass-Spring Systems (utah.edu)



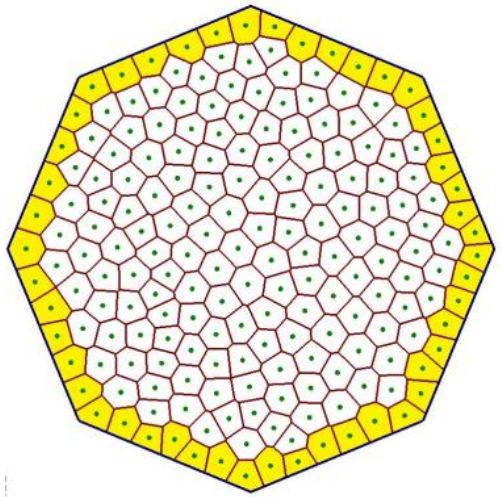
Liquid Simulation by SPH Method (B, 1)

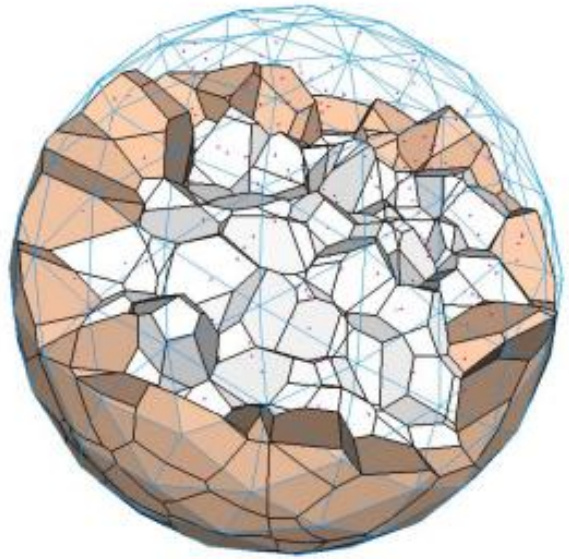
- **Implement the (preferably 3D) Position Based Fluids for particle-based liquid simulation**
  - A fixed-domain neighbor search engine.
  - The parallized Position Based Fluids solver.
  - Surface extraction for rendering or render particles.
- **Reference**
  - https://mmacklin.com/pbf_sig_preprint.pdf



Voronoi Diagram (C, 1)

- **Compute 2D/3D clipped Voronoi Diagram**
  - Voronoi Diagram in a limited area is called clipped Voronoi Diagram.
  - Implement basic or advanced algorithm to construct Voronoi Diagram on 2D/3D erea.
  - Implement 2D will get C
  - Implement a fast 3D version may get B
  - Implement an extremely fast 3D version may get A
- **Reference**
  - [1986] A sweepline algorithm for Voronoi diagrams
  - [2016] Efficient Computation of 3D Clipped Voronoi Diagram
  - [2020] Parallel computation of 3D clipped Voronoi diagrams

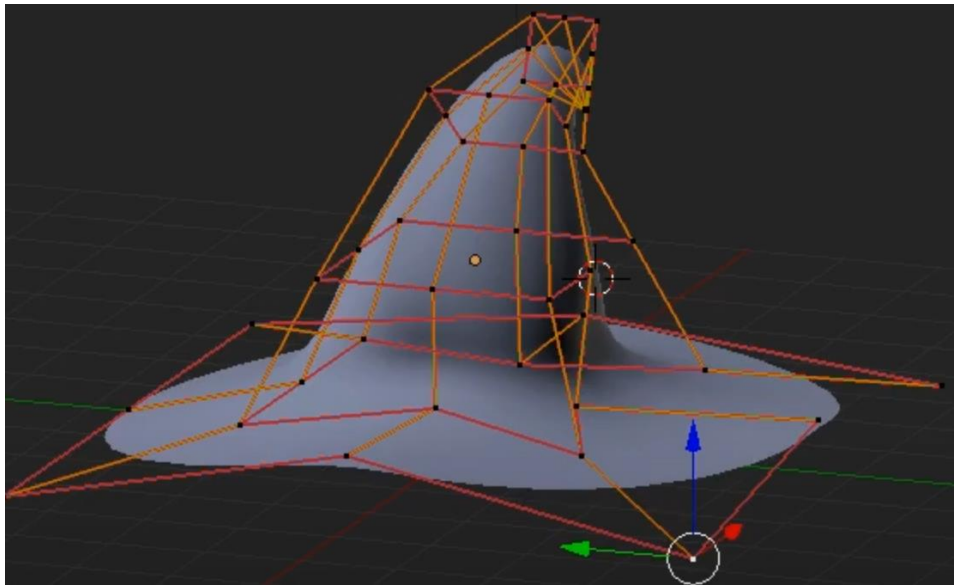




NURBS Surface Editing (C, 1)

- **A small interactive NURBS modeling editor**
  - Implement NURB evaluator
  - Create an interactive UI system for surface editing

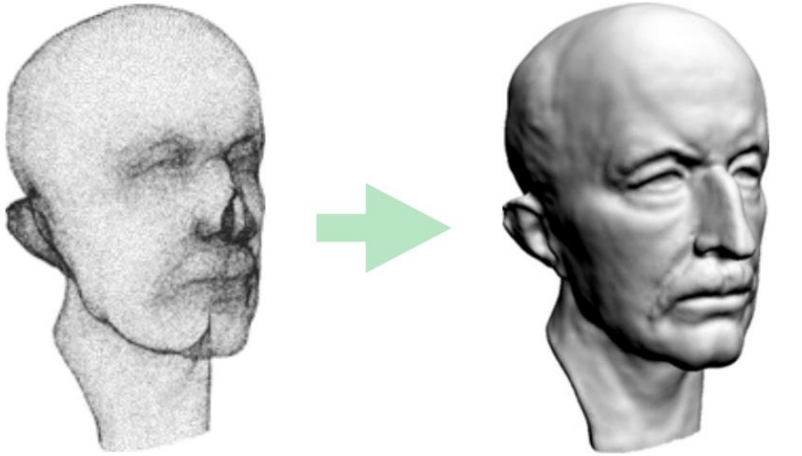
- **Reference**:
  - Piegl L, Tiller W. The NURBS book[M].



Surface Reconstruction (C, 1)

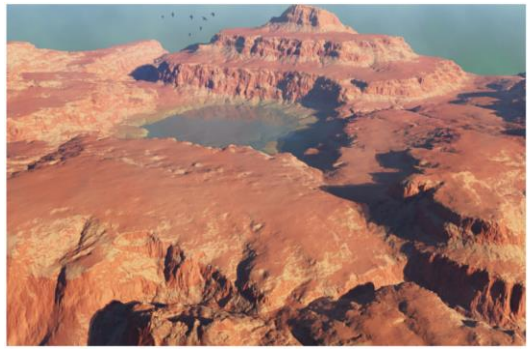
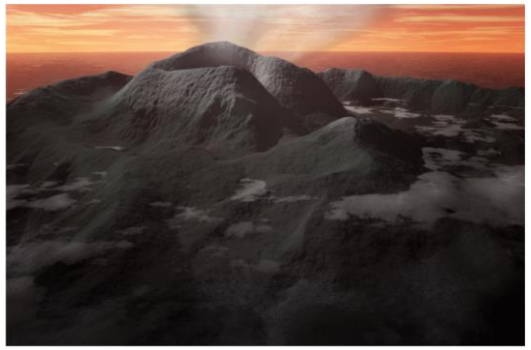
- **Poisson surface reconstruct from point clouds**
  - Estimate SDF of point clouds by Poisson equation
  - Extract meshes via marching cubes

- **Reference**:
  - Michael Kazhdan, et al.
  - Poisson surface reconstruction



Terrain Synthesis (C, 1)

- **Synthesize terrain for use in computer games**
  - Synthesize terrain with stochastic heightmap or other methods
  - Free-view navigation in the generated terrain (first-person perspective).


- **Reference**
  - [2004] Realtime Procedural Terrain Generation
  - [2007] Terrain Synthesis from Digital Elevation Models
  - [2015] Parallel, Realistic and Controllable Terrain Synthesis