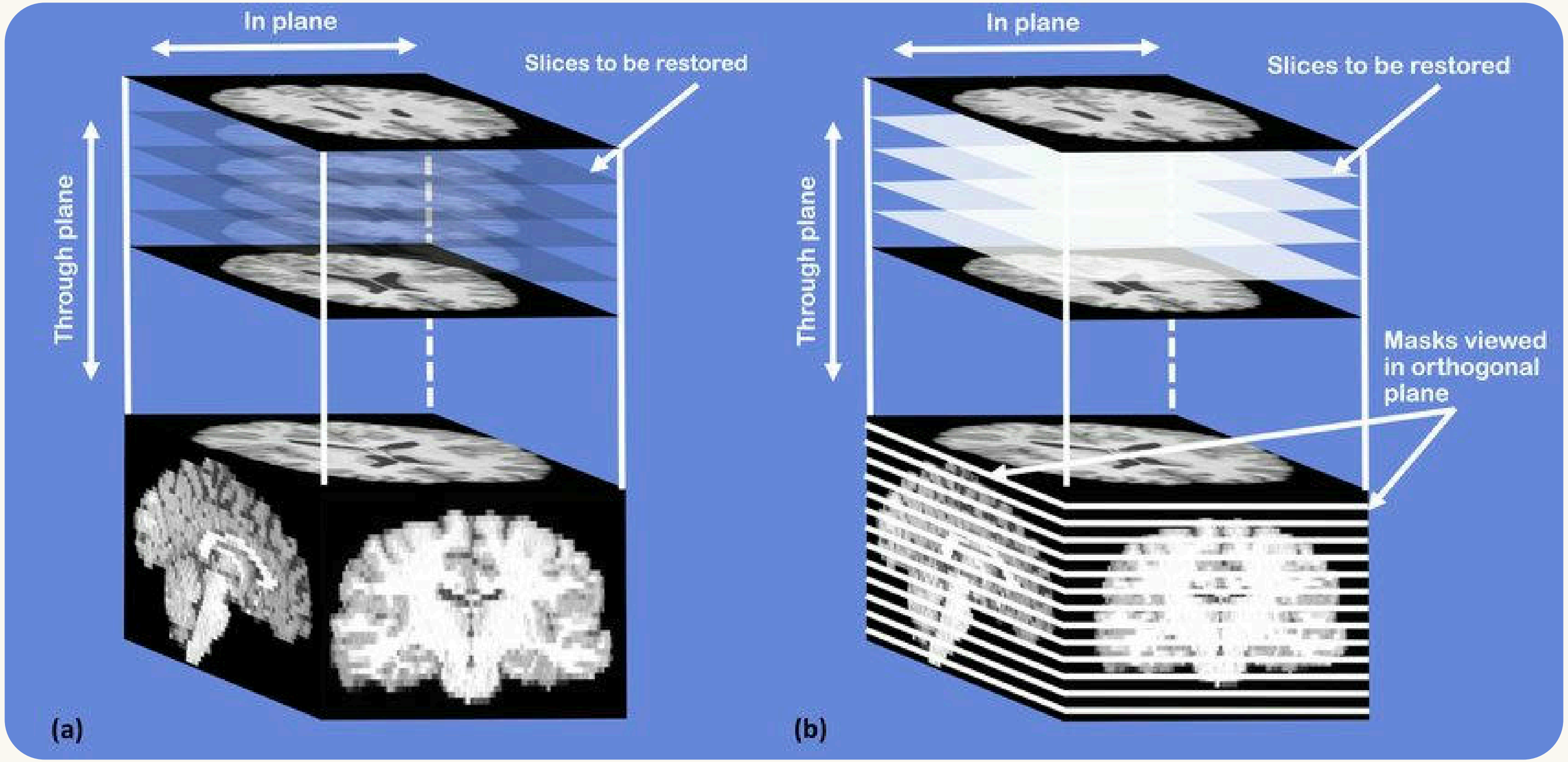
Marching Cubes Algorithm

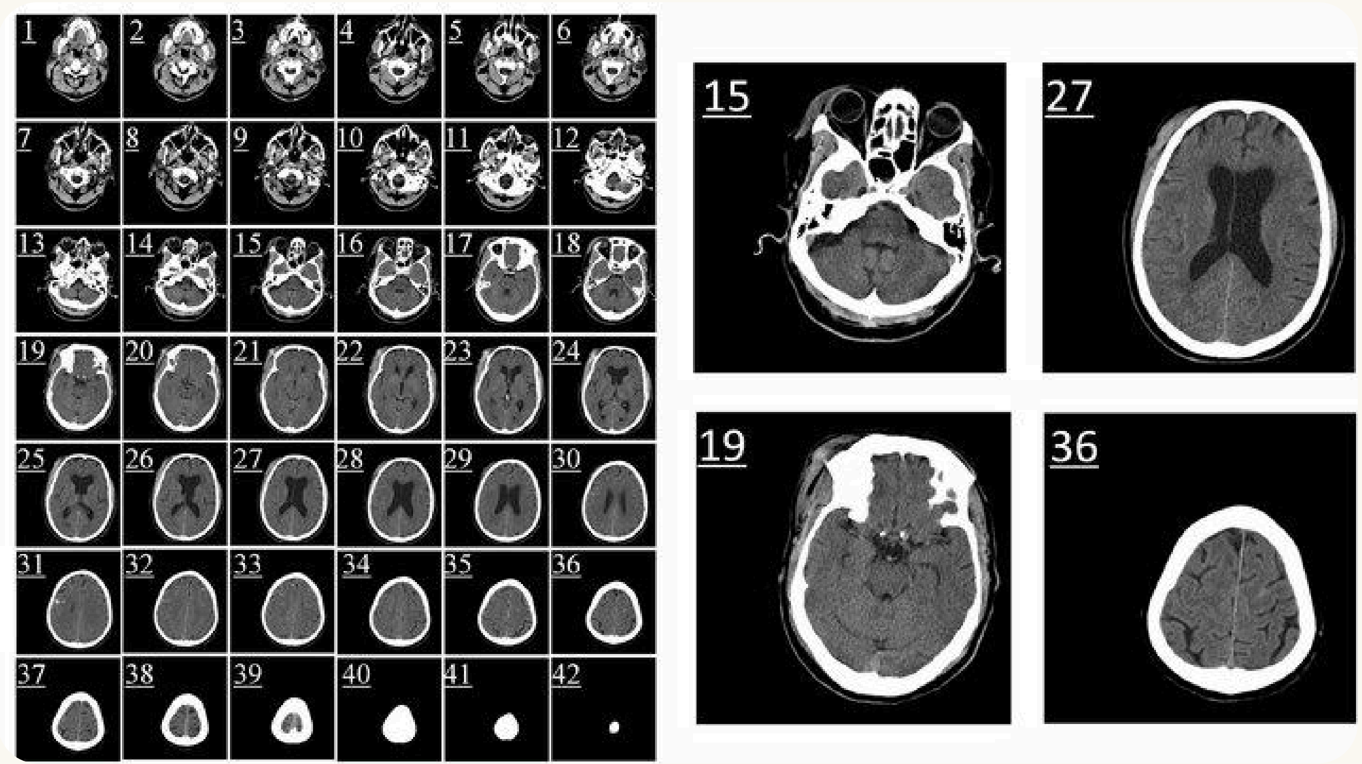
Apratim Mishra
2022BCSE061

1 Problem

- Medical imaging techniques like MRI and CT scan only produce 2d slices of images
- Using these 2D image stacks, we want to visualize them in 3D



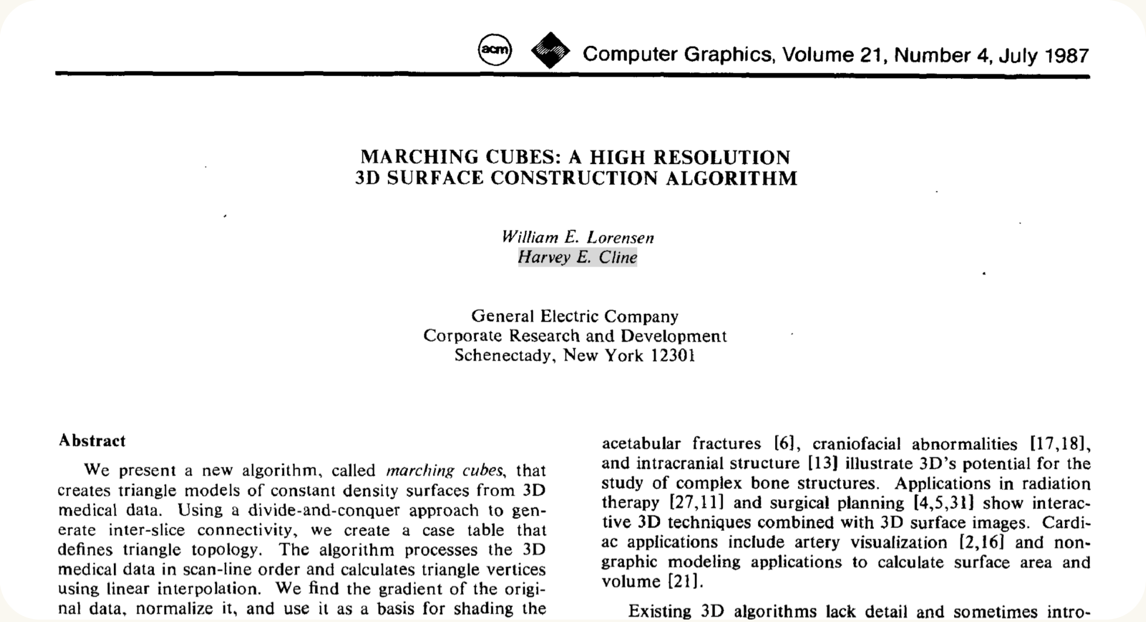
2D image slices received from MRI scan



2D image slices received from CT scan

2 Solution

Algorithm developed by *William E. Lorensen* and *Harvey E. Cline* published in 1987 SIGGRAPH proceedings



Computer Graphics, Volume 21, Number 4, July 1987

MARCHING CUBES: A HIGH RESOLUTION 3D SURFACE CONSTRUCTION ALGORITHM

William E. Lorensen
Harvey E. Cline

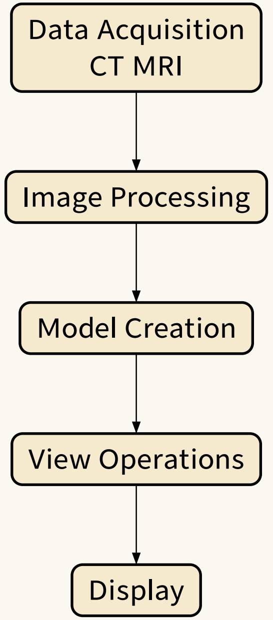
General Electric Company
Corporate Research and Development
Schenectady, New York 12301

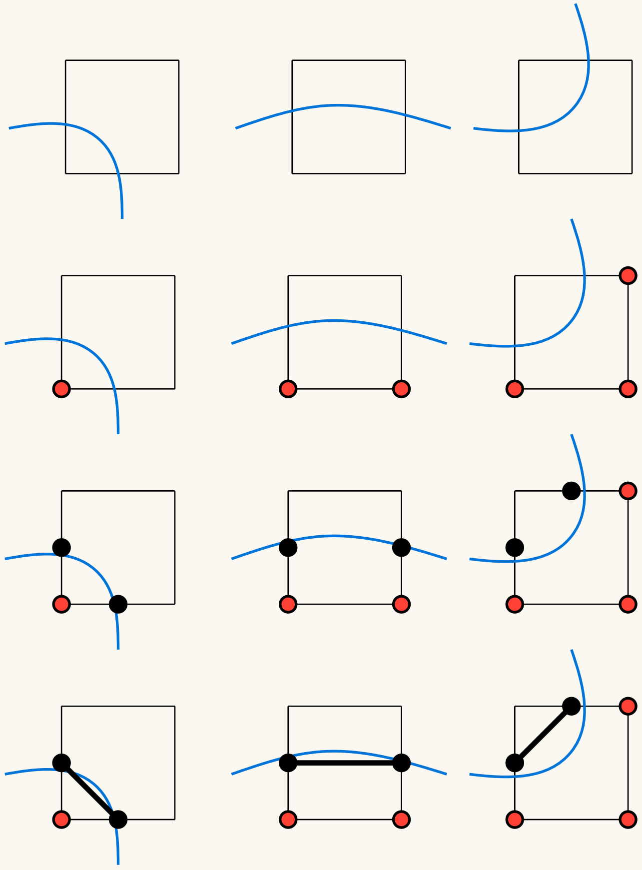
Abstract

We present a new algorithm, called *marching cubes*, that creates triangle models of constant density surfaces from 3D medical data. Using a divide-and-conquer approach to generate inter-slice connectivity, we create a case table that defines triangle topology. The algorithm processes the 3D medical data in scan-line order and calculates triangle vertices using linear interpolation. We find the gradient of the original data, normalize it, and use it as a basis for shading the

acetabular fractures [6], craniofacial abnormalities [17,18], and intracranial structure [13] illustrate 3D's potential for the study of complex bone structures. Applications in radiation therapy [27,11] and surgical planning [4,5,31] show interactive 3D techniques combined with 3D surface images. Cardiac applications include artery visualization [2,16] and nongraphic modeling applications to calculate surface area and volume [21].

Existing 3D algorithms lack detail and sometimes intro-



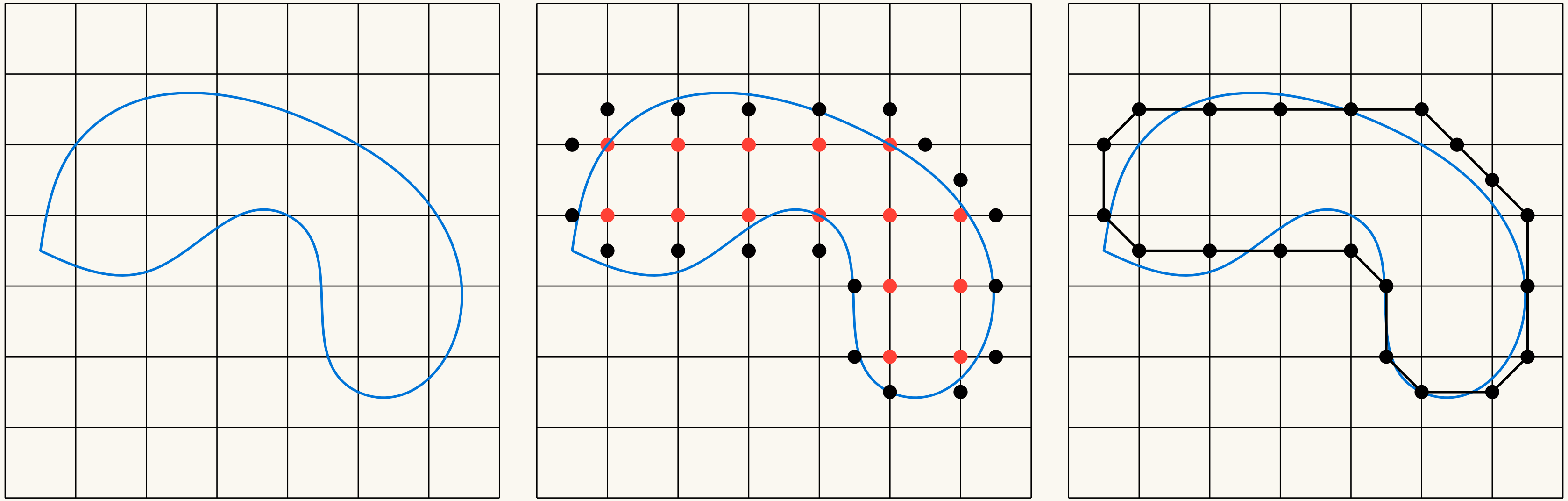


Object drawn on 2d grid

Points inside the object marked in red

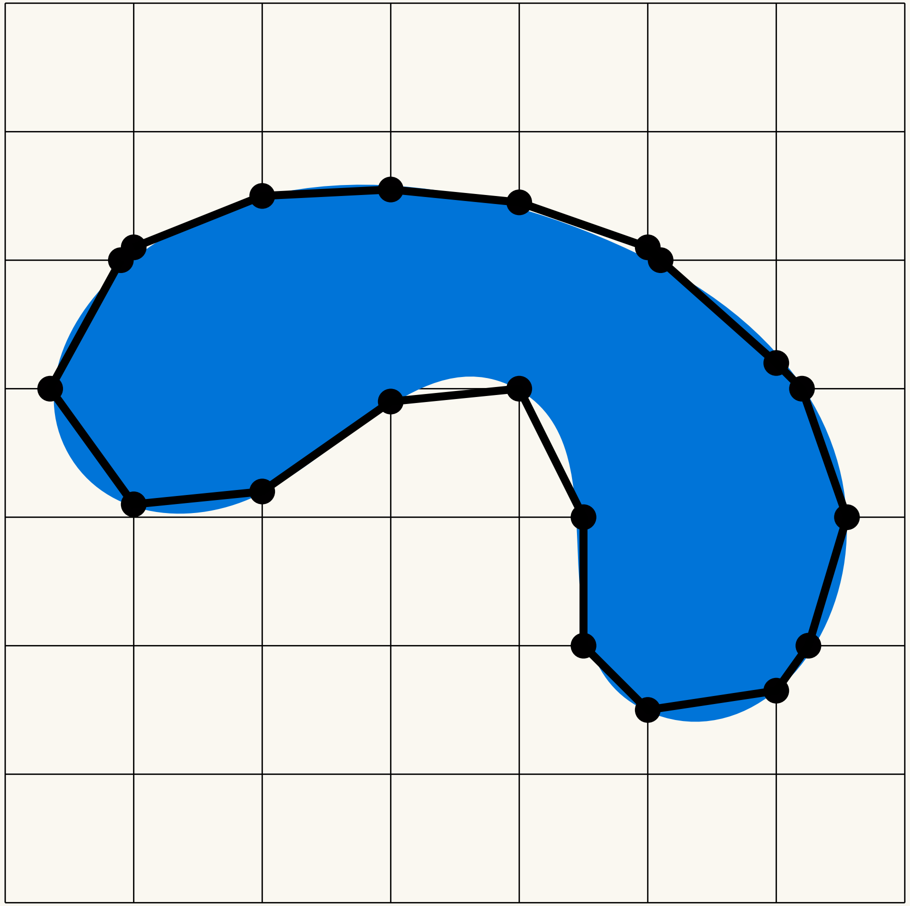
Middle points activated due to red points marked in **black**

Join the activated points



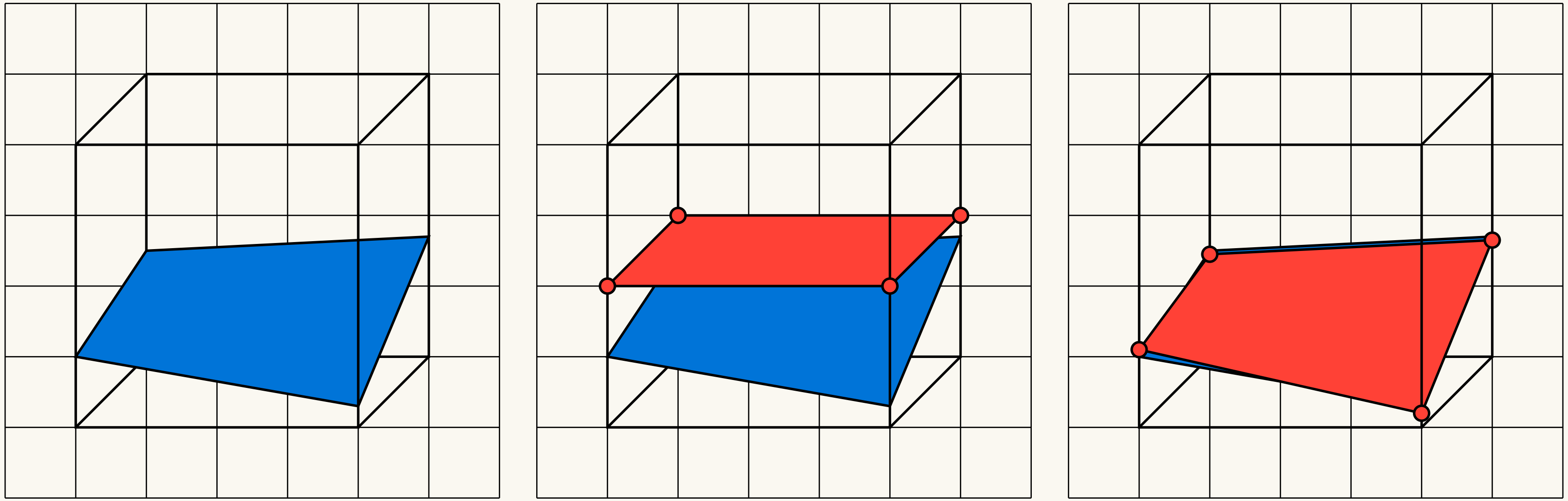
Marching Squares in 2D

1: Object traced on squares in blue
2: Points inside the object in red, points on boundary in **black**
3: Water tight traced mesh



Optimisation

After the last step, move the points closer to object boundary, by moving it along its edge axis without going out of the edge boundary.

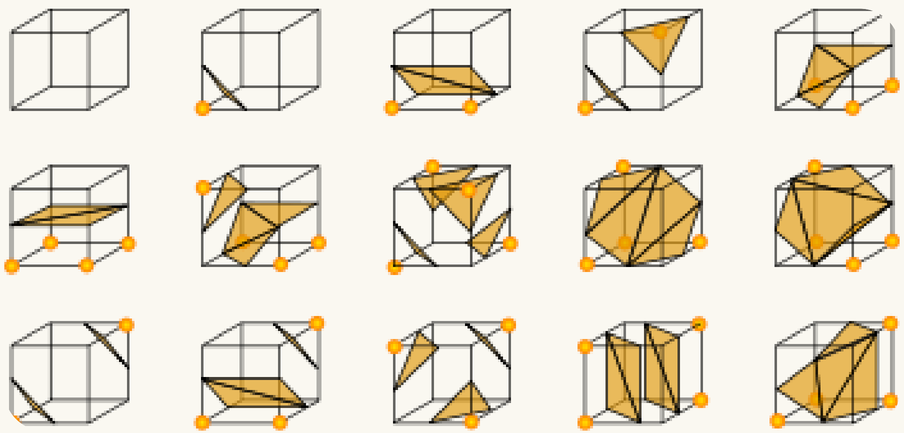


Marching Cube in 3D

1: Object traced in cube

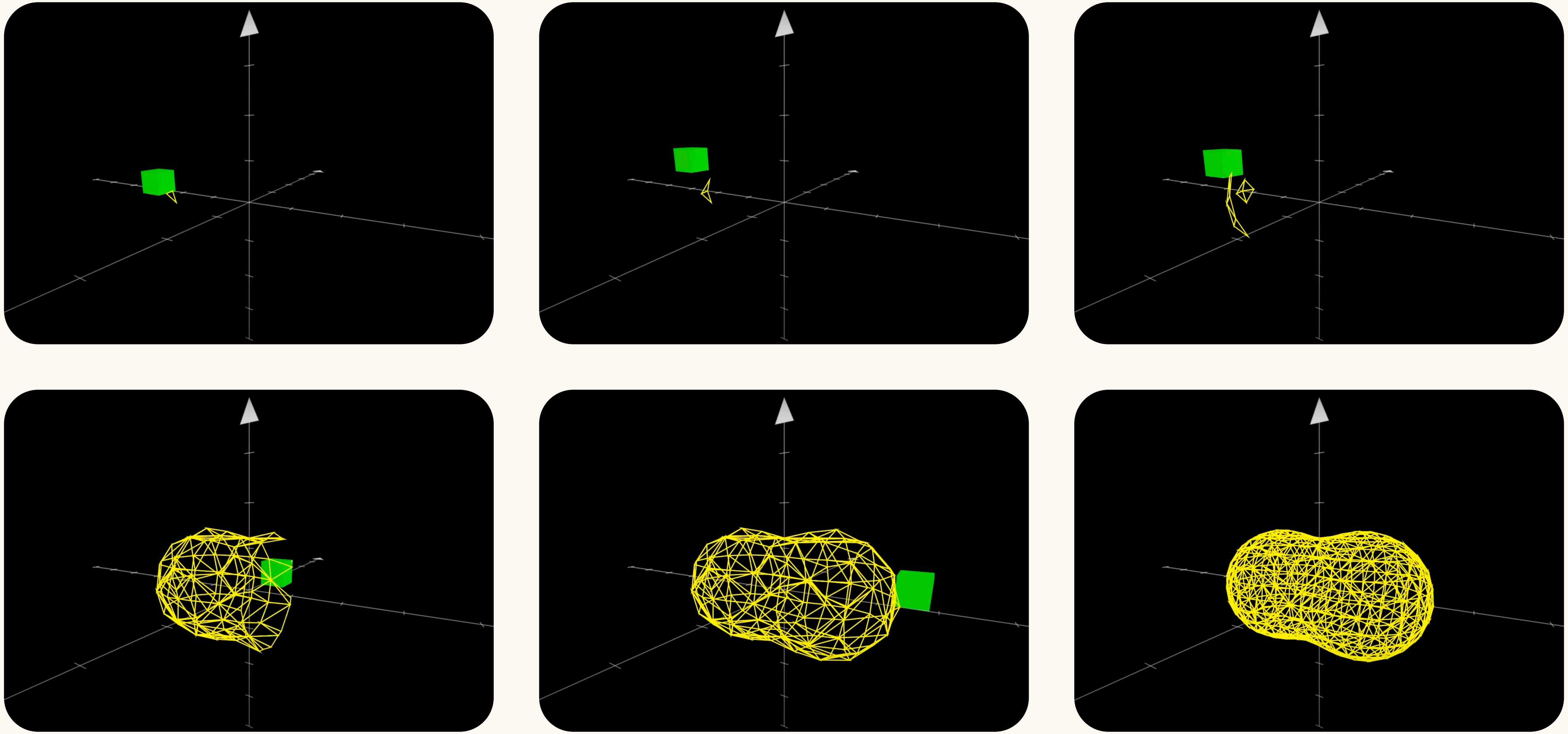
2: Mark mid points to make shape around the object, shown in red

3: Move the points along the respective edge axis for optimisation



All 15 possible cases

- Since each vertex can either be outside or inside, there are technically $2^8 = 256$ possible configurations, but many of these are equivalent to one another.
- There are only 15 unique cases, shown here.
- This allows for easy triangle generation using lookup table for each case

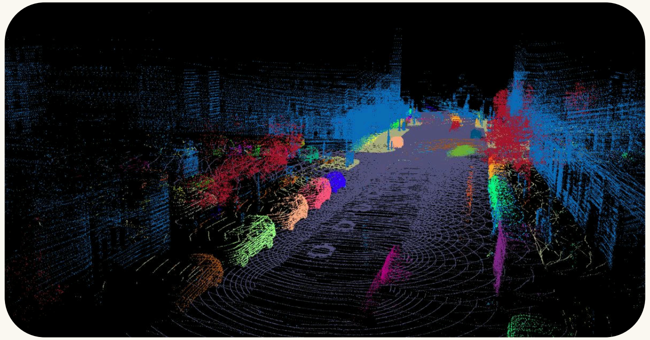


3 Implementation Details

- **Data Structures:** Efficient storage of vertex and edge information is crucial.
- **Optimization:** Techniques like edge and vertex caching can improve performance.
- **Parallelization:** The algorithm is well-suited for parallel processing due to the independence of cube evaluations.

4 Applications

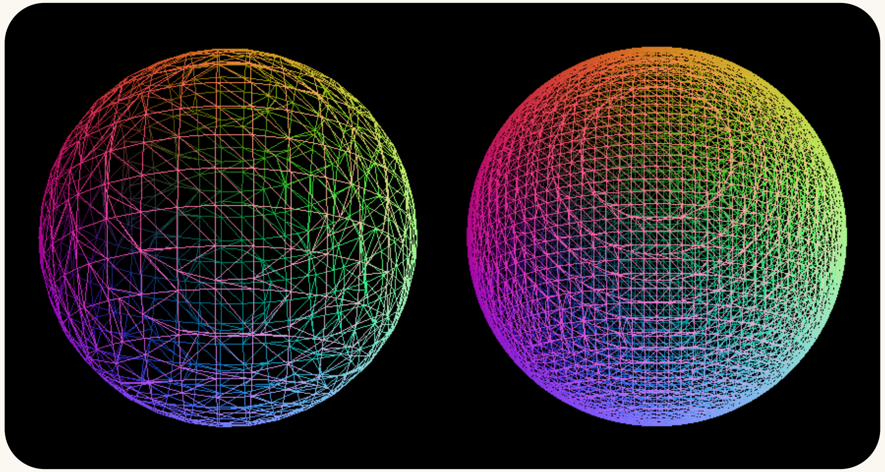
- **Medical Imaging:** Visualization of anatomical structures from CT and MRI scans.
- **Scientific Visualization:** Representation of scalar fields in physics and engineering.
- **Computer Graphics:** Modeling complex surfaces and terrains.

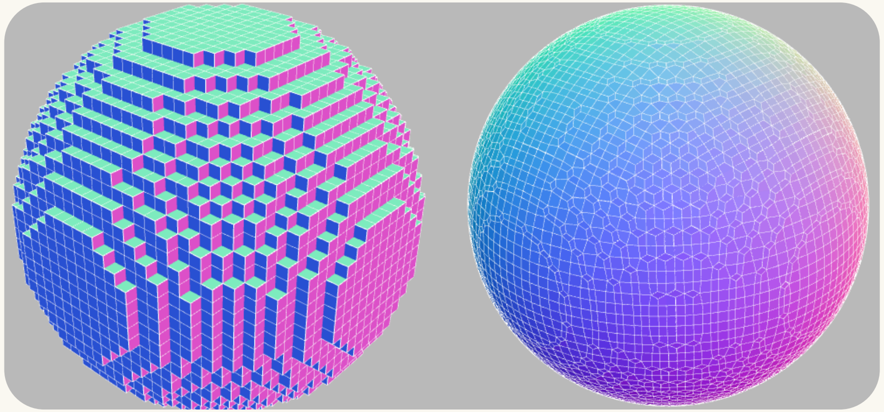


Lidar Point Cloud

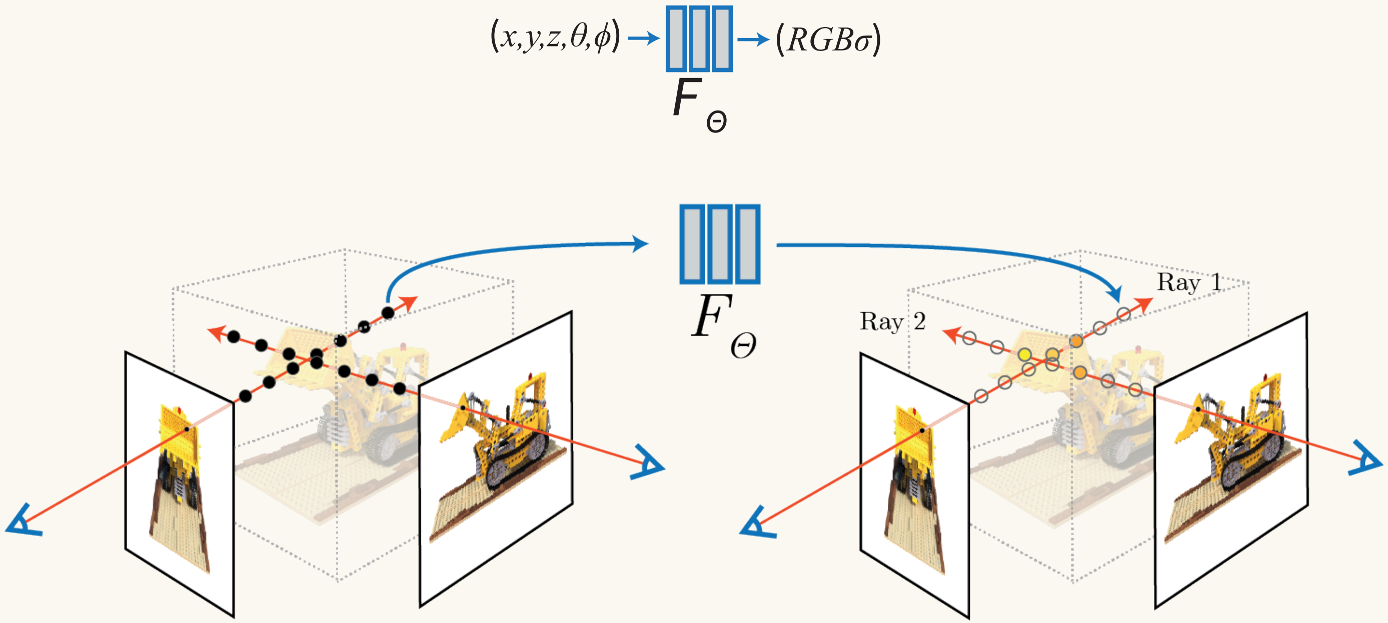
5 Advantages

- **High Resolution:** Produces detailed and accurate 3D surfaces.
- **Efficiency:** Capable of processing large datasets effectively.
- **Versatility:** Applicable to various fields requiring 3D visualization.





6 Future Retrospective



NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis

THANK YOU