

3D Representation

Van-Thinh Vo

June 2025

Mục lục

1	Giới thiệu	2
2	Các phương pháp biểu diễn 3D	2
2.1	Depth map	2
2.2	Voxel Grid	3
2.3	Point Cloud	3
2.4	Mesh	4
2.5	Signed Distance Function (SDF)	4
2.6	Neural Radiance Field (NeRF)	5
2.7	3D Gaussian Splatting	5

Nếu có bất kỳ sai sót nào, mọi người có thể phản hồi qua email thinh.vovan@hcmut.edu.vn để hỗ trợ mình với nha :v

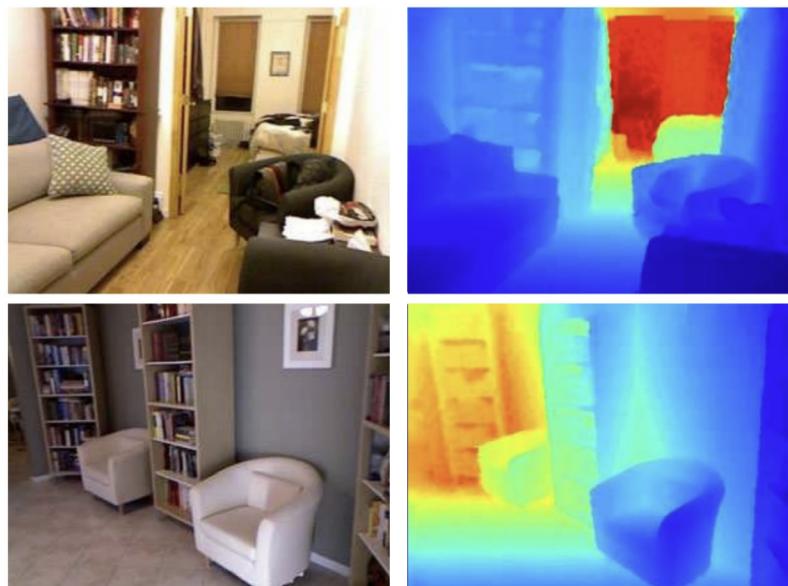
1 Giới thiệu

Biểu diễn 3D (3D Representation) là một lĩnh vực nghiên cứu về cách biểu diễn và trực quan hóa vật thể ba chiều giống như thế giới thật. Nó có rất nhiều ứng dụng trong thực tế như đồ họa máy tính: game 3D, vẽ 3D...; trong phương tiện tự lái (autonomous driving): nhận diện được môi trường và chướng ngại vật, khoảng cách dựa trên ảnh 2D từ camera; hay trong y tế, việc xây dựng các cơ quan và hệ thống trong cơ thể thành 3D giúp hình dung dễ dàng hơn và hỗ trợ cho các kế hoạch phẫu thuật; lĩnh vực robotic cũng rất cần những biểu diễn 3D cho robot có thể quan sát thế giới một cách rõ ràng nhất. Qua đó ta thấy đây là một lĩnh vực rất có ý nghĩa trong thực tiễn và vì thế, đã có rất nhiều nghiên cứu và phát triển cho đến hiện tại. Ở đây mình xin trình bày một số phương pháp chủ yếu được tham khảo theo khóa học **CS231n** của đại học Stanford và paper.

2 Các phương pháp biểu diễn 3D

2.1 Depth map

Depth map là phương pháp dựa vào một bản đồ độ sâu cho biết khoảng cách từ camera đến vật thể trong ảnh. Nó có thể được xây dựng từ một số cảm biến hiện đại bằng tia hồng ngoại. (Ở đây ta sẽ nghiên cứu cách về cách dùng mô hình Deep Learning để sinh ra bản đồ độ sâu).



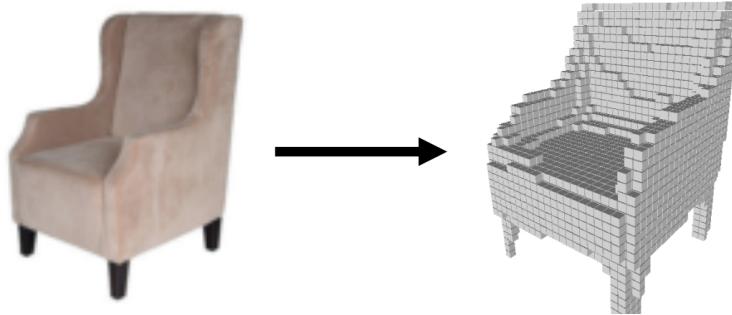
Tuy nhiên, việc biểu diễn khoảng cách tuyệt đối là một điều rất khó. Ta không thể biết khoảng cách tuyệt đối đến một điểm mà chỉ có thể biết khoảng cách tương đối giữa các điểm trong ảnh. Ví dụ như hai vật thể có kích thước gấp đôi nhau, nhưng vật thể có kích thước to ở xa gấp đôi vật thể ở gần thì trên mặt phẳng ảnh, chúng có kích thước bằng nhau dẫn đến vấn đề về bất biến cân bằng (scale invariant). Để giải quyết vấn đề này, ta sử dụng metric *L2 Distance* kết hợp với *scale invariant* (y là predict, y^* là ground-truth), hàm sẽ phạt khoảng cách tương

đối giữa các điểm thay vì khoảng cách tuyệt đối:

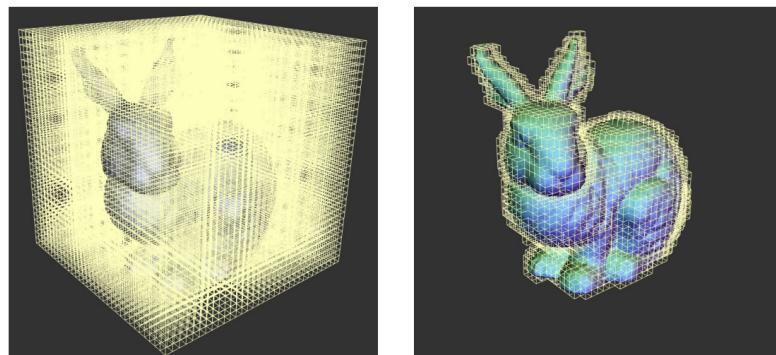
$$\begin{aligned}
 D(y, y^*) &= \frac{1}{2n^2} \sum_{i,j} ((\log y_i - \log y_j) - (\log y_i^* - \log y_j^*))^2 \\
 &= \frac{1}{n} \sum_i d_i^2 - \frac{1}{n^2} \sum_{i,j} d_i d_j \\
 &= \frac{1}{n} \sum_i d_i^2 - \frac{1}{n^2} \left(\sum_i d_i \right)^2
 \end{aligned}$$

2.2 Voxel Grid

Là phương pháp biểu diễn bằng các khối hình hộp gọi là *Voxel* (Volumetric Elements). Đây là một cách biểu diễn đơn giản (lấy ý tưởng từ pixel trong ảnh 2D). Mỗi *Voxel* có thể chứa thông tin về màu sắc, mật độ và hay các đặc trưng khác của vật. Kỹ thuật này đơn giản hóa việc xây dựng cái vật thể có cấu tạo phức tạp, tuy nhiên nó có độ phân giải (resolution) không cao và cần rất nhiều bộ nhớ.

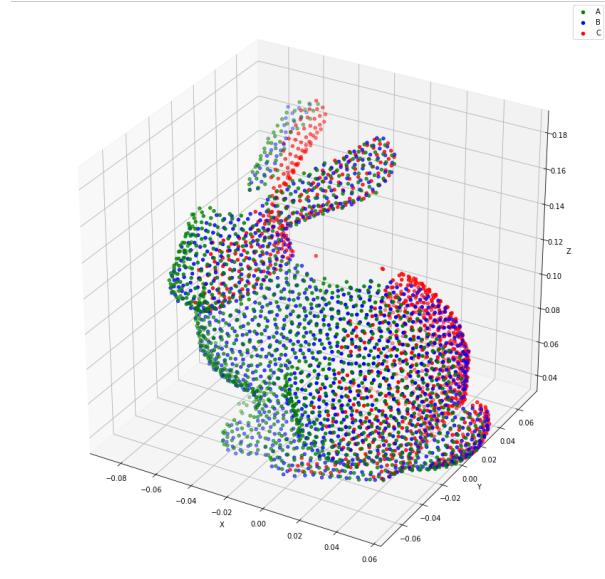


Oct-Tree là một phương pháp giúp tối ưu bộ nhớ cho biểu diễn bằng *Voxel*. Bằng cách dùng cây bát phân chia vùng không gian 3D ra thành các phần nhỏ, mỗi node của cây sẽ là một phần của không gian. Vì thế ta có thể lược bỏ các node không chứa vật thể để tối ưu về lưu trữ.



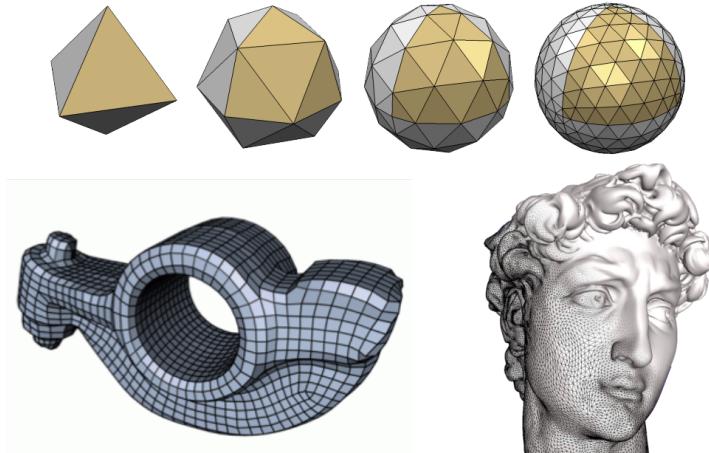
2.3 Point Cloud

Dám mây điểm (Point Cloud) là phương pháp biểu diễn vật thể bằng các điểm rời rạc. Các điểm này có thể chứa thông tin về vị trí 3D, màu sắc, cường độ hay pháp tuyến. Biểu diễn này thường được sinh ra từ các máy scan. *Point Cloud* thể hiện được thông tin về hình học và thể tích như Mesh hay Voxel nên gặp khó khăn trong các bài toán Segmentation, Recognition,... Ngoài ra *Point Cloud* cũng có thể chứa nhiều.



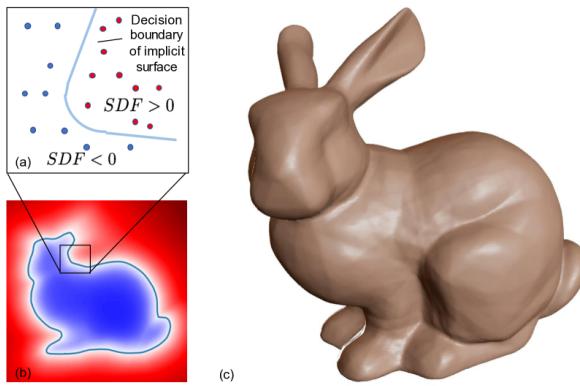
2.4 Mesh

Mesh biểu diễn bề mặt vật thể bằng cách kết hợp các đa giác phẳng lại với nhau (thường là tam giác). Phương pháp này cho phép biểu diễn các hình học phức tạp với độ phân giải tỉ lệ với số lượng đa giác.



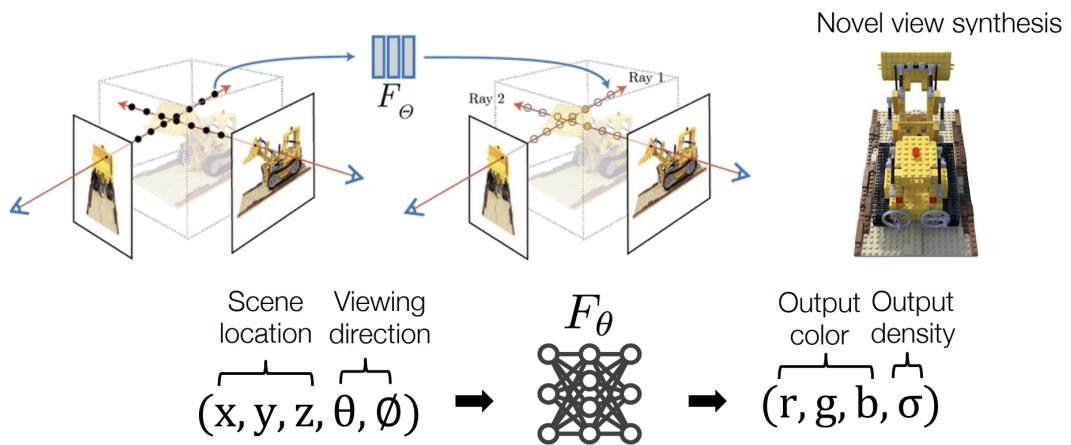
2.5 Signed Distance Function (SDF)

Là phương pháp xây dựng một hàm \mathbf{F} cho mỗi điểm (point) trong không gian là khoảng cách từ nó đến mặt phẳng gần nhất của vật thể. $\mathbf{F} > 0$ nếu điểm nằm ngoài vật thể, $\mathbf{F} = 0$ nếu nằm trên bề mặt và $\mathbf{F} < 0$ khi nằm trong vật thể. Ta sẽ xây dựng hàm \mathbf{F} dựa trên một Point Cloud hay Voxel Grid, sau đó các điểm mà $\mathbf{F} = 0$ sẽ là bề mặt của vật thể.



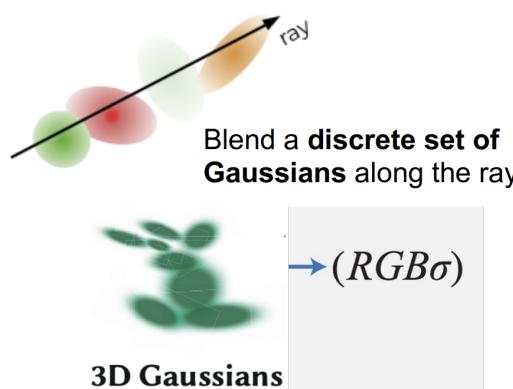
2.6 Neural Radiance Field (NeRF)

NeRF là cách mô hình lại hình dạng 3D thông qua các góc nhìn 2D khác nhau của vật thể bằng mạng Neural Network. Phương pháp này cần được huấn luyện trong thời gian rất lâu.



2.7 3D Gaussian Splatting

Vật thể được biểu diễn là các hạt Gaussian 3D (vị trí, màu, hình dạng, hướng, độ trong suốt). Ta có thể tưởng tượng hạt Gaussian là một "đám mây màu mờ" trong không gian. Ánh xạ các hạt này vào không gian 2D ta sẽ được ảnh 2D với tốc độ nhanh hơn nhiều so với NeRF.



3D Gaussian Splatting

Tài liệu

- [1] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. *3D Gaussian Splatting for Real-Time Radiance Field Rendering*. ACM SIGGRAPH 2023 Conference Proceedings, 2023. DOI: 10.1145/3592142.
- [2] David Eigen, Christian Puhrsch, and Rob Fergus. *Depth Map Prediction from a Single Image using a Multi-Scale Deep Network*. In Advances in Neural Information Processing Systems (NeurIPS), volume 27, pages 2366–2374, 2014.
- [3] Christopher B. Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. *3DR2N2: A Unified Approach for Single and Multi-view 3D Object Reconstruction*. In European Conference on Computer Vision (ECCV), pages 628–644, 2016.