

25/11/2022

# CÁC ĐỐI TƯỢNG HÌNH HỌC TRONG KHÔNG GIAN 3 CHIỀU

Môn học: Cấu trúc dữ liệu và giải thuật nâng cao

Mã lớp: CS532.N11

Giảng viên hướng dẫn: Nguyễn Thanh Sơn

# Thành viên thực hiện



Tên: Huỳnh Việt Tuấn Kiệt

MSSV: 20521494

Khoa: Khoa học máy tính

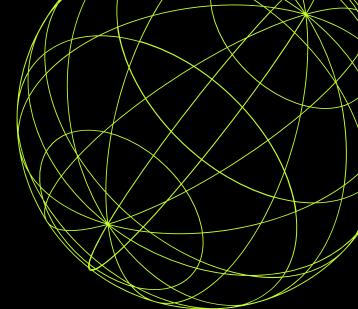


Tên: Nguyễn Đức Anh Phúc

MSSV: 20520276

Khoa: Khoa học máy tính

# Introduction & Motivation

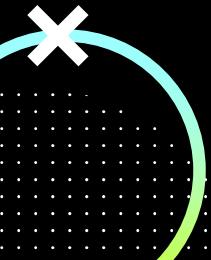


## Giới thiệu

- Hình học 3D là những khối có 3 chiều: Dài, rộng, cao
- Các vật thể 3D có thể được nhìn thấy xung quanh ta: Khối rubik, viên xúc xắc, quyển sách, quả bóng, que kem, chai nước, ...
- Trong góc nhìn toán học, hình học 3D liên quan đến các đối tượng 3D như hình trụ, hình lập phương, hình cầu, ...

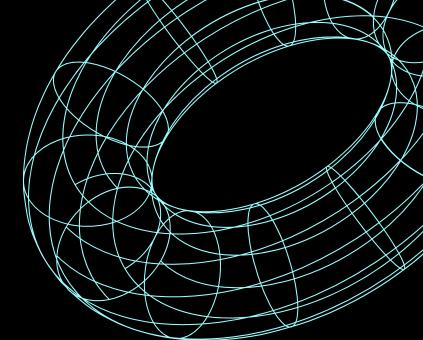
## Động lực nghiên cứu

- Đồ họa máy tính (Computer Graphics) liên quan đến việc tạo ra hình ảnh với sự trợ giúp của máy tính **sử dụng biểu diễn 3D của dữ liệu hình học**.
- Ngày nay, đồ họa máy tính là công nghệ cốt lõi trong nhiếp ảnh kỹ thuật số, phim, trò chơi video, điện thoại di động và màn hình máy tính, cùng nhiều ứng dụng chuyên biệt





# Nội Dung



01

## Các khái niệm cơ bản

Khái niệm một số đối tượng  
hình học trong không gian 3D

02

## Các phép biến đổi

Các phép biến đổi áp dụng trên  
đối tượng hình học trong không  
gian 3D

03

## Bài toán

Các bài toán áp dụng trên  
đối tượng hình học trong  
không gian 3D

04

## Ứng dụng

Ứng dụng của đối tượng  
hình học không gian 3D  
trong thực tế

05

## Thực hành

Demo biểu diễn các đối  
tượng hình học trong không  
gian 3D

# 01

## Các khái niệm cơ bản

Khái niệm một số đối tượng hình học trong không gian 3D



# Nội Dung

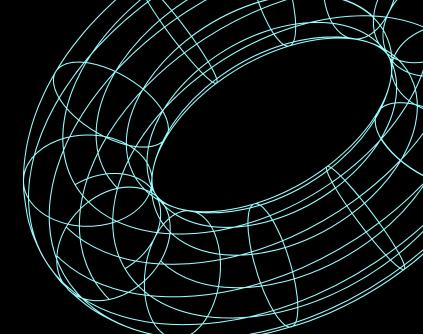
**01** Điểm (point)

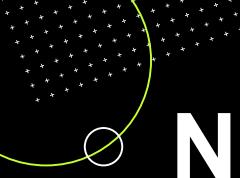
**02** Đường thẳng (line)

**03** Vector

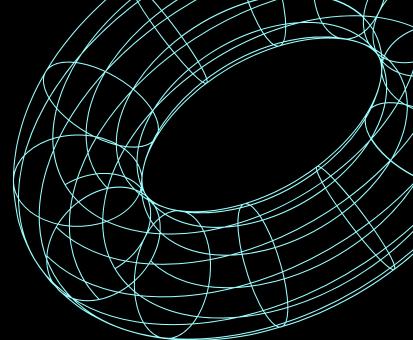
**04** Mặt phẳng (plane)

**05** Khối (solid)





# Nội Dung



**01** Điểm (point)

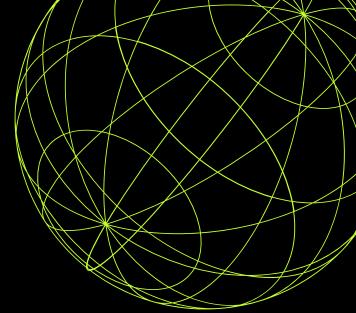
**02** Đường thẳng (line)

**03** Vector

**04** Mặt phẳng (plane)

**05** Khối (solid)

# Điểm (Point)



## Khái niệm

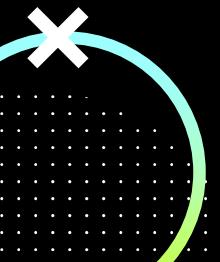
- Một điểm được xác định là một vị trí trong bất kì không gian nào

## Biểu diễn

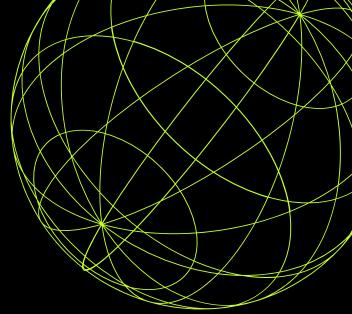
- Một điểm được biểu diễn bởi một dấu chấm (.)

## Kích thước

- Một điểm không có bất kì kích thước nào



# Phân loại điểm

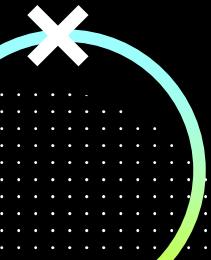


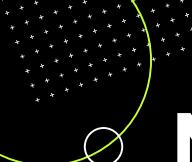
## Điểm thẳng hàng và điểm không thẳng hàng

- Điểm thẳng hàng: 3 hay nhiều điểm cùng nằm trên một đường thẳng
- Điểm không thẳng hàng: Nhóm các điểm không cùng nằm trên một đường thẳng

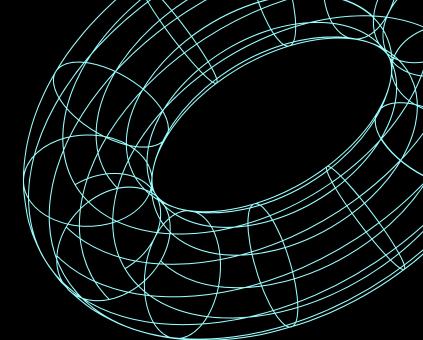
## Điểm đồng phẳng và điểm không đồng phẳng

- Điểm đồng phẳng: Nhóm các điểm cùng nằm trên một mặt phẳng
- Điểm không đồng phẳng: Nhóm các điểm không cùng nằm trên một mặt phẳng





# Nội Dung



**01** Điểm (point)

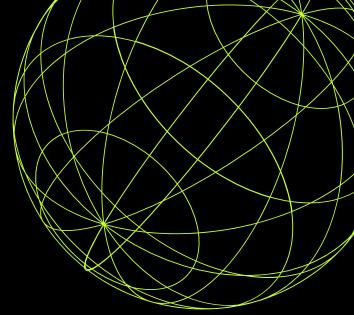
**02** Đường thẳng (line)

**03** Vector

**04** Mặt phẳng (plane)

**05** Khối (solid)

# Đường thẳng (Line)

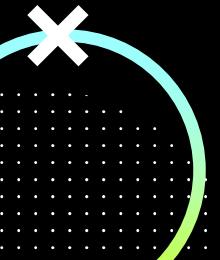


## Khái niệm

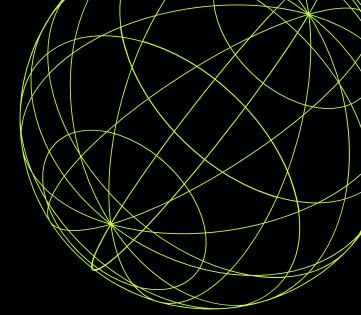
- Một đường thẳng được tạo bởi một tập hợp các điểm thẳng hàng được kéo dài vô hạn theo các hướng ngược nhau

## Kích thước

- Đường thẳng là hình một chiều, có chiều dài nhưng không có chiều rộng



# Phân loại đường thẳng



## Đường ngang (Horizontal Line)

- Một đường được ánh xạ theo phương song song với trục  $x$  trong một mặt phẳng

## Đường dọc (Vertical Line)

- Một đường được ánh xạ theo phương song song với trục  $y$  trong một mặt phẳng

## Đường giao nhau (Intersecting Lines)

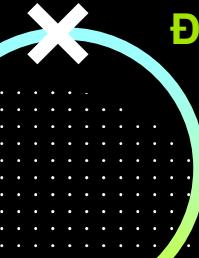
- Hai đường thẳng chéo nhau và gặp nhau tại một điểm

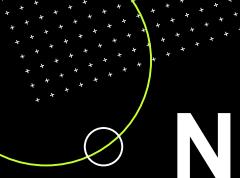
## Đường thẳng vuông góc (Perpendicular Lines)

- Hai đường cắt nhau tạo thành góc  $90^\circ$

## Đường thẳng song song (Parallel Line)

- Hai đường thẳng không cắt nhau tại bất cứ điểm nào





# Nội Dung

01 Điểm (point)

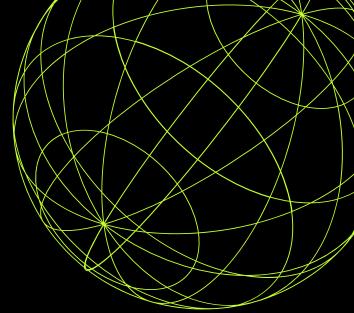
02 Đường thẳng (line)

03 Vector

04 Mặt phẳng (plane)

05 Khối (solid)

# Vector



## Khái niệm

- Vector là đối tượng hình học mô tả chuyển động từ điểm này đến điểm khác

## Thuộc tính

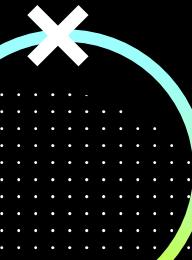
- Vector là đối tượng hình học có hướng và độ lớn

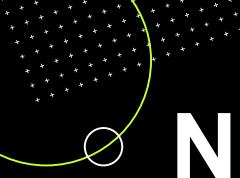
## Biểu diễn

- Vector được biểu diễn bằng một đoạn thẳng có mũi tên
  - Độ dài đoạn thẳng chỉ độ lớn vector
  - Mũi tên chỉ hướng

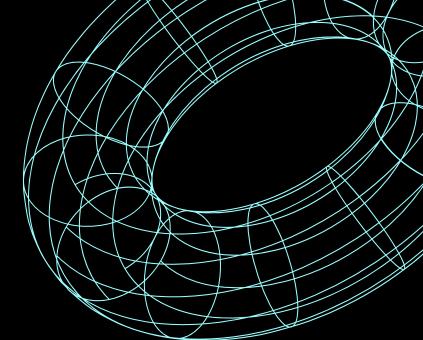
## Ví dụ

- Vận tốc, gia tốc, lực, ...





# Nội Dung



**01** Điểm (point)

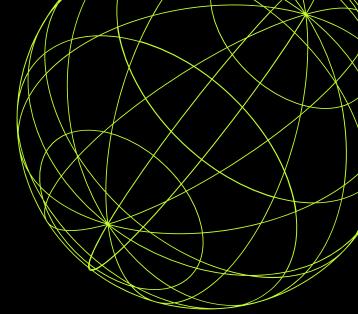
**02** Đường thẳng (line)

**03** Vector

**04** Mặt phẳng (plane)

**05** Khối (solid)

# Mặt phẳng (plane)

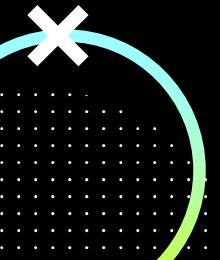


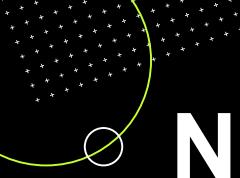
## Khái niệm

- Mặt phẳng là một mặt hai chiều phẳng kéo dài vô tận, không có cạnh và bề dày

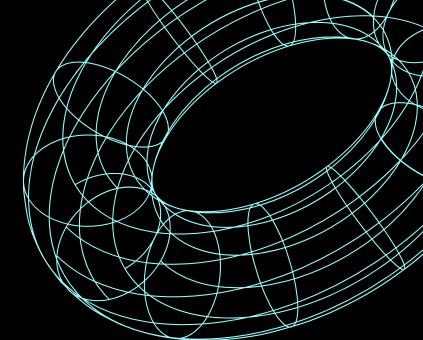
## Khái niệm

- Tường, sàn nhà, bề mặt giấy, mặt bàn, ...





# Nội Dung



01 Điểm (point)

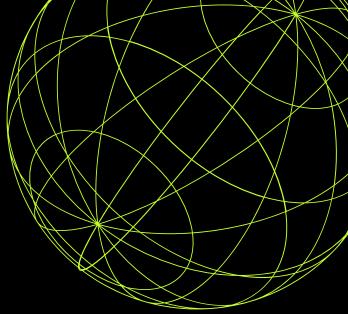
02 Đường thẳng (line)

03 Vector

04 Mặt phẳng (plane)

05 Khối (solid)

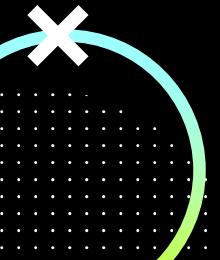
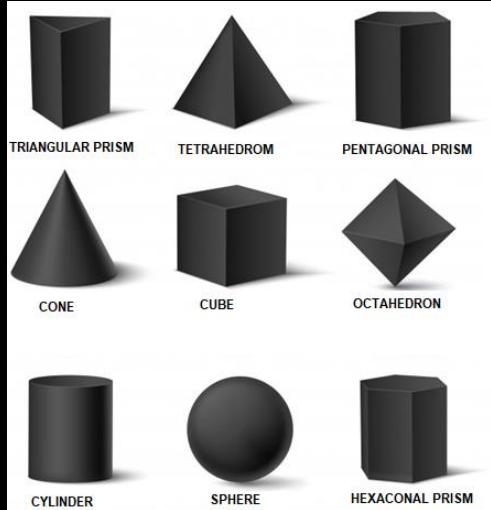
# Khối (Solid)



## Khái niệm

- Bất kỳ vật thể nào chiếm không gian được gọi là khối hoặc vật thể 3 chiều
- Trong hình học, khối (hình dạng ba chiều) là một hình đồng chất hoặc một vật thể hoặc hình dạng có ba kích thước — chiều dài, chiều rộng và chiều cao.

## Ví dụ



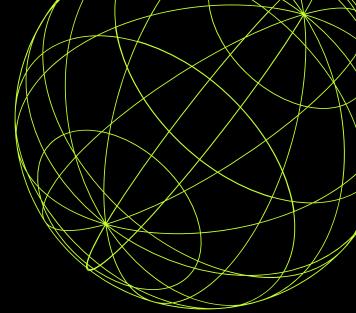
# 02

## Các phép biến đổi 3D

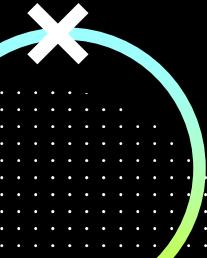
Các phép biến đổi áp dụng trên đối tượng hình học trong không gian 3D

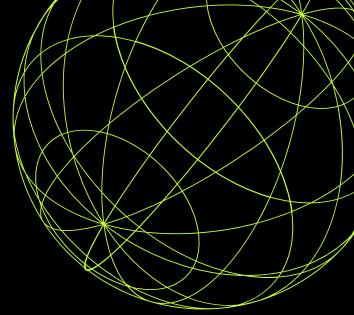


# 3D Transformation



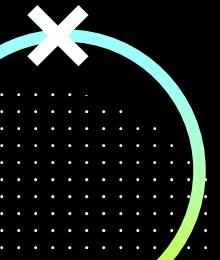
- Các phép biến đổi 3D diễn ra trong không gian 3D
- Các phép biến đổi 3D quan trọng và có chút phức tạp hơn so với các phép biến đổi 2D
- Các phép biến đổi rất hữu ích trong việc thay đổi vị trí, kích thước, hướng, hình dạng của đối tượng
- Khi các đối tượng 2D hay 3D trên máy tính được biến đổi (di chuyển, xoay, thay đổi kích thước, ...), các phép biến đổi tuyến tính này có thể được biểu diễn dưới dạng ma trận.

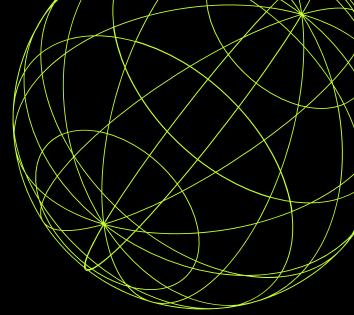




# Tại sao phải sử dụng ma trận?

Matrices allow arbitrary linear transformations to be displayed in a consistent format, suitable for computation

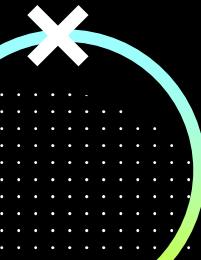


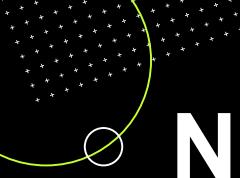


## Tại sao phải sử dụng ma trận

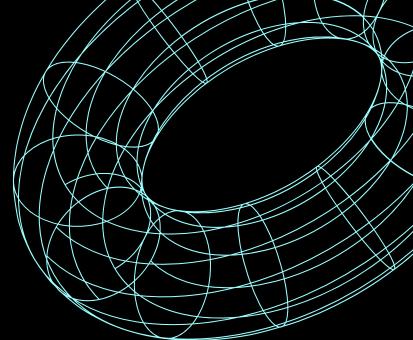
- Ma trận đặc biệt hữu ích trong các phép biến đổi trong lĩnh vực đồ họa máy tính (Computer Graphics), bởi vì mỗi phép biến đổi có thể biểu diễn theo một định dạng nhất quán, dễ dàng biểu diễn dưới dạng cấu trúc dữ liệu máy tính
- Phép biến đổi được biểu diễn dưới dạng ma trận  $M$  có thể được hoàn tác bằng cách áp dụng nghịch đảo của ma trận (ma trận  $M^{-1}$  đảo ngược phép biến đổi được áp dụng bởi ma trận  $M$ )

|





# Nội Dung



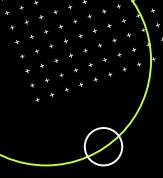
**01** Phép tịnh tiến (Translation)

**02** Phép quay (Rotation)

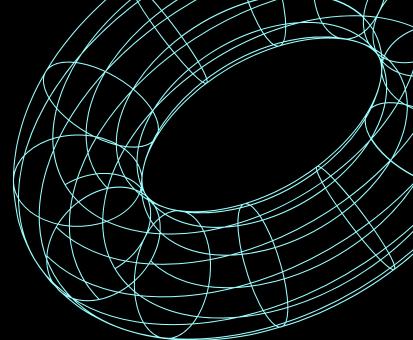
**03** Phép tỉ lệ (Scaling)

**04** Phép phản xạ (Reflection)

**05** Phép cắt (Shear)



# Nội Dung



**01** Phép tịnh tiến (Translation)

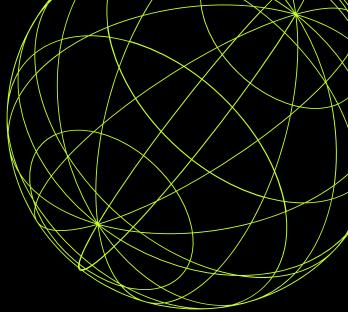
**02** Phép quay (Rotation)

**03** Phép tỉ lệ (Scaling)

**04** Phép phản xạ (Reflection)

**05** Phép cắt (Shear)

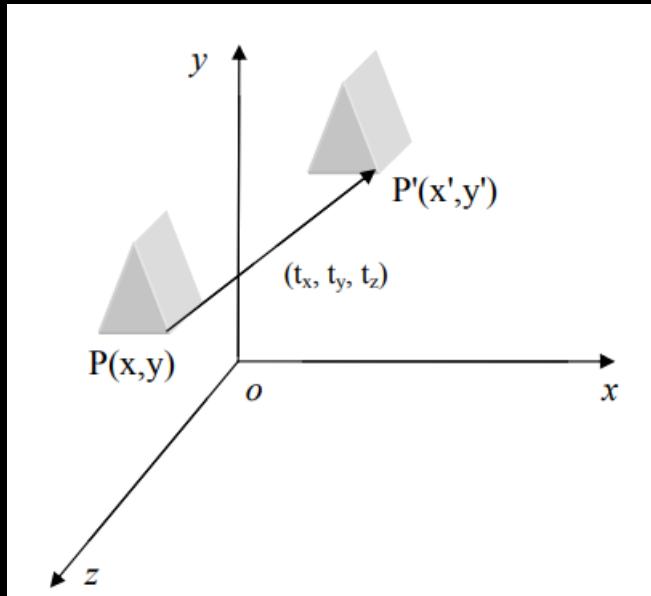
# Phép tịnh tiến (Translation)



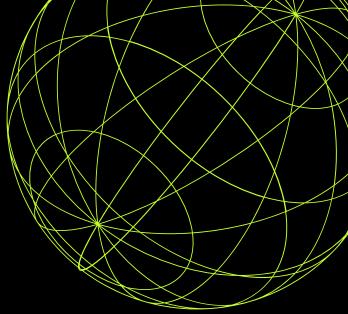
## Khái niệm

- Tịnh tiến 3D là quá trình dịch chuyển một đối tượng từ vị trí này sang vị trí khác trong không gian 3D

## Ví dụ



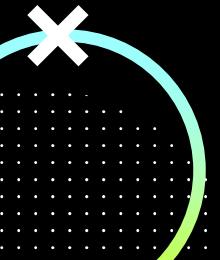
# Phép tịnh tiến (Translation)



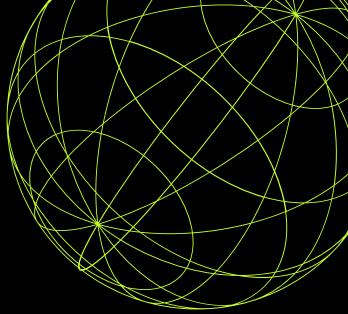
## Tham số

- Tọa độ khởi đầu của đối tượng  $O$  ( $X, Y, Z$ )
- Tọa độ mới của đối tượng  $O$  sau khi tịnh tiến ( $X', Y', Z'$ )
- Vector tịnh tiến ( $T_x, T_y, T_z$ )

|



# Phép tịnh tiến (Translation)

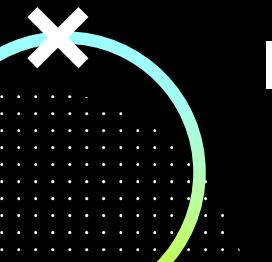


## Phương trình tịnh tiến

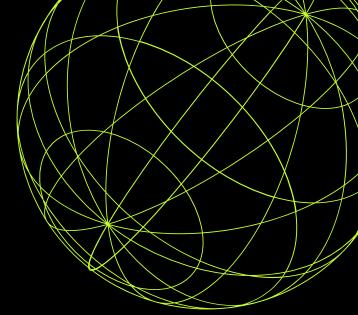
- $X' = X + T_X$
- $Y' = Y + T_Y$
- $Z' = Z + T_Z$

## Ma trận tịnh tiến

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & T_X \\ 0 & 1 & 0 & T_Y \\ 0 & 0 & 1 & T_Z \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$



# Phép tịnh tiến (Translation)



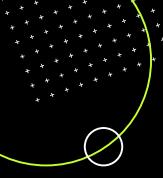
## Phương trình tịnh tiến

- $X' = X + T_X$
- $Y' = Y + T_Y$
- $Z' = Z + T_Z$

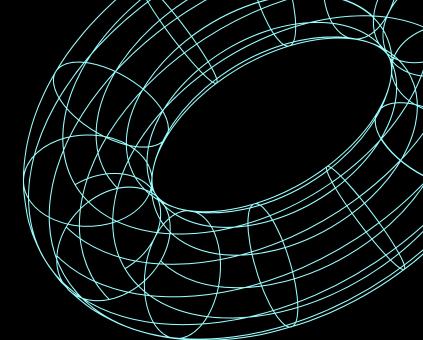
## Ma trận hoàn tác tịnh tiến

$$\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & T_X \\ 0 & 1 & 0 & T_Y \\ 0 & 0 & 1 & T_Z \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} \times \begin{bmatrix} X' \\ Y' \\ Z' \\ 1' \end{bmatrix}$$





# Nội Dung



**01** Phép tịnh tiến (Translation)

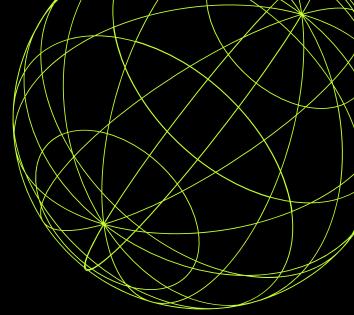
**02** Phép quay (Rotation)

**03** Phép tỉ lệ (Scaling)

**04** Phép phản xạ (Reflection)

**05** Phép cắt (Shear)

# Phép quay (Rotation)



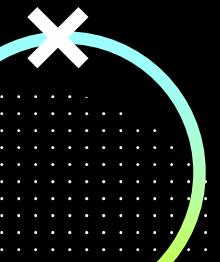
## Khái niệm

- Phép quay 3D là quá trình xoay một đối tượng theo một góc trong không gian 3D

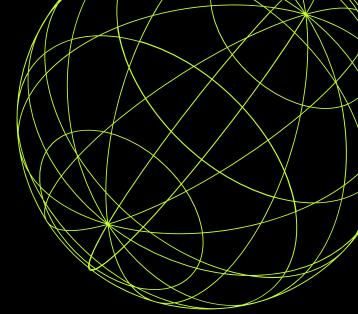
## Các loại phép quay

- Phép quay quanh các trục tọa độ
- Phép quay quanh một trục song song với trục tọa độ
- Phép quay quanh một trục bất kì

|

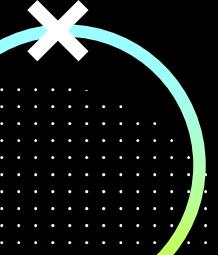
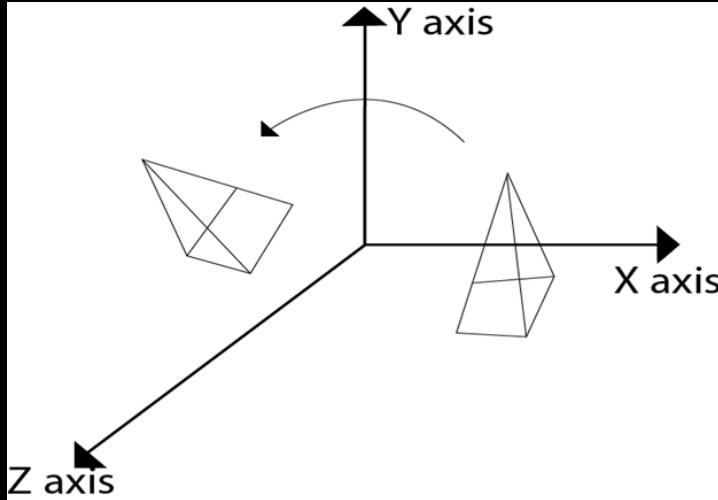


# Phép quay (Rotation)



## Tham số

- Tọa độ khởi đầu của đối tượng  $O (X, Y, Z)$
- Tọa độ mới của đối tượng  $O$  sau khi quay  $(X', Y', Z')$
- Góc quay  $\theta$



# Phép quay quanh các trục tọa độ



## Quay quanh trục Ox

- $X' = X$
- $Y' = Y \times \cos \theta - Z \times \sin \theta$
- $Z' = Y \times \sin \theta + Z \times \cos \theta$

## Ma trận quay quanh trục Ox

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta & 0 \\ 0 & \sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$



# Phép quay quanh các trục tọa độ



## Quay quanh trục Oy

- $X' = X \times \cos \theta + Z \times \sin \theta$
- $Y' = Y$
- $Z' = -X \times \sin \theta + Z \times \cos \theta$

## Ma trận quay quanh trục Oy

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \end{bmatrix} \times \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$



# Phép quay quanh các trục tọa độ



## Quay quanh trục Oz

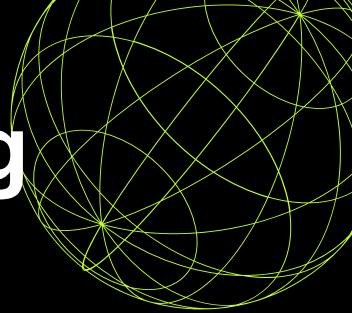
- $X' = X \times \cos \theta - Y \times \sin \theta$
- $Y' = X \times \sin \theta + Y \times \cos \theta$
- $Z' = Z$

## Ma trận quay quanh trục Oz

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$



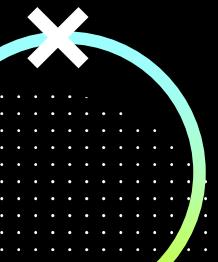
# Phép quay quanh một trục song song với trục tọa độ



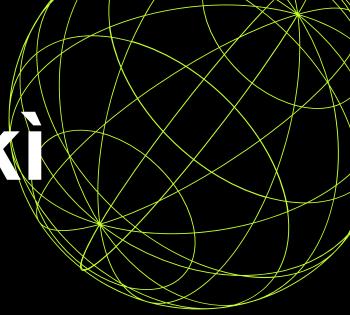
## Giả sử trục quay song song với trục Ox

- Áp dụng phép tịnh tiến  $T$  để đưa trục quay về trục Ox
- Áp dụng phép quay  $R(\theta)$  quay đối tượng quanh trục Ox một góc  $\theta$
- Áp dụng phép tịnh tiến  $T^{-1}$  để đưa trục quay về vị trí ban đầu

\* Các trường hợp còn lại tương tự

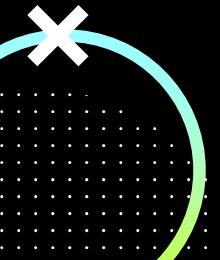


# Phép quay quanh một trục bất kì

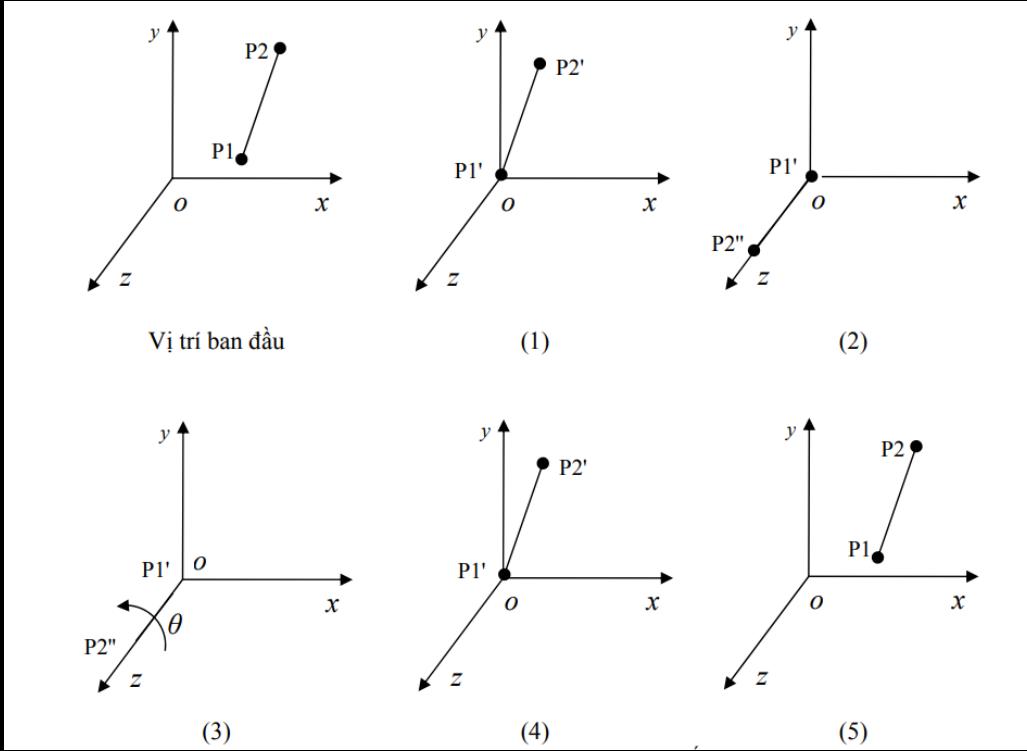


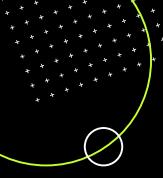
## Giả sử trục quay song song với trục

- Áp dụng phép tịnh tiến  $T$  để đưa trục quay về vị trí đi qua gốc tọa độ
- Áp dụng phép quay  $R(\alpha)$  đưa trục quay về vị trí trùng với một trục tọa độ
- Áp dụng phép quay  $R(\theta)$  quay vật thể quanh trục quay (trục tọa độ)
- Áp dụng lần lượt phép quay  $R^{-1}(\alpha)$  và phép tịnh tiến  $T^{-1}$  đưa trục quay về vị trí ban đầu

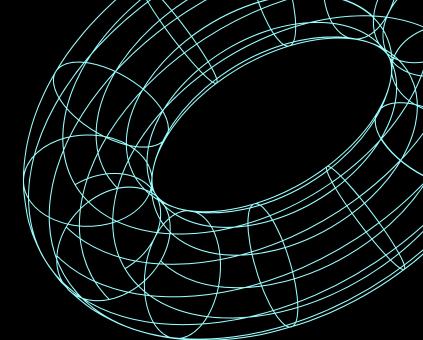


# Phép quay quanh một trục bất kì





# Nội Dung



**01** Phép tịnh tiến (Translation)

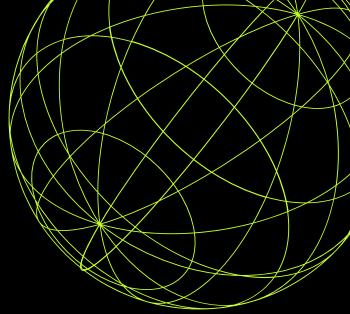
**02** Phép quay (Rotation)

**03** Phép tỉ lệ (Scaling)

**04** Phép phản xạ (Reflection)

**05** Phép cắt (Shear)

# Phép tỉ lệ (Scaling)

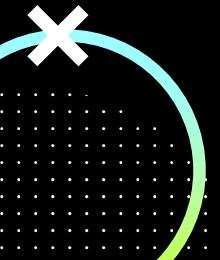
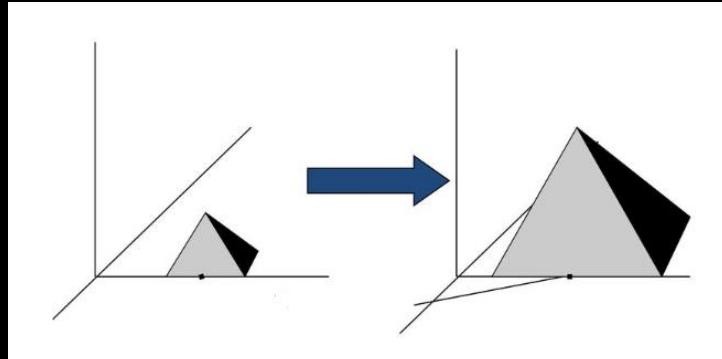


## Khái niệm

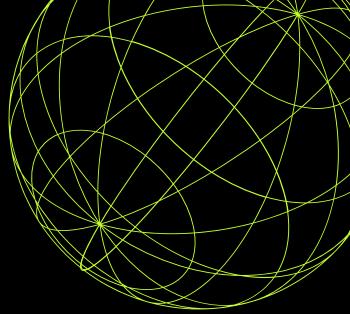
- Phép tỉ lệ là quá trình điều chỉnh hay thay thế kích thước của đối tượng

## Hệ số tỉ lệ

- Nếu hệ số tỉ lệ  $> 1$ , thì kích thước đối tượng tăng
- Nếu hệ số tỉ lệ  $< 1$ , thì kích thước đối tượng giảm



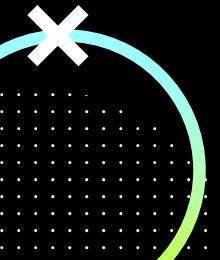
# Phép tỉ lệ (Scaling)



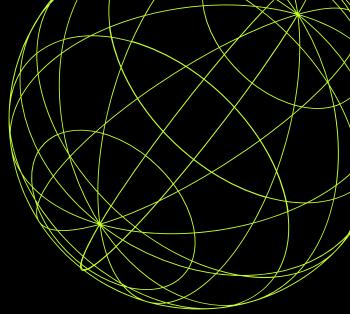
## Tham số

- Tọa độ khởi đầu của đối tượng  $O$  ( $X, Y, Z$ )
- Tọa độ mới của đối tượng  $O$  ( $X', Y', Z'$ )
- Hệ số tỉ lệ ( $S_x, S_y, S_z$ )

|



# Phép tỉ lệ (Scaling)

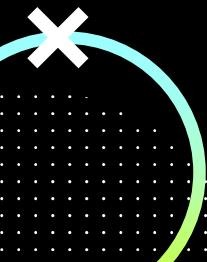


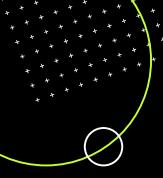
## Phương trình tỉ lệ

- $X' = X \times S_X$
- $Y' = Y \times S_Y$
- $Z' = Z \times S_Z$

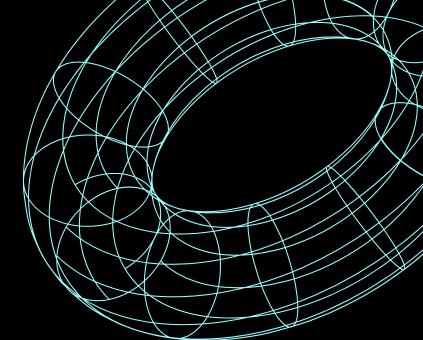
## Ma trận tỉ lệ

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$





# Nội Dung



**01** Phép tịnh tiến (Translation)

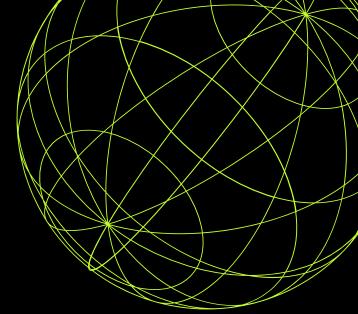
**02** Phép quay (Rotation)

**03** Phép tỉ lệ (Scaling)

**04** Phép phản xạ (Reflection)

**05** Phép cắt (Shear)

# Phép phản xạ (Reflection)

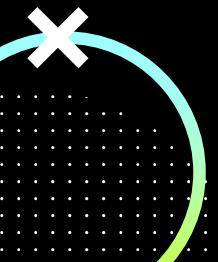


## Khái niệm

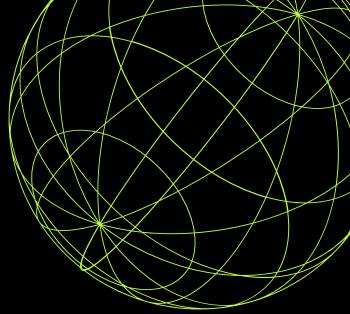
- Phép phản xạ là một loại phép quay, trong đó góc quay bằng  $180^0$
- Đối tượng phản xạ luôn được hình thành bên kia của “Gương”
- Kích thước của đối tượng phản xạ bằng với kích thước của đối tượng chính

## Các loại phản xạ

- Phản xạ qua mặt phẳng  $XY$
- Phản xạ qua mặt phẳng  $YZ$
- Phản xạ qua mặt phẳng  $XZ$

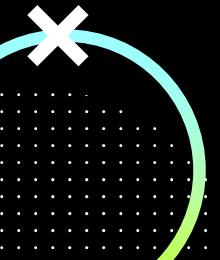


# Phép phản xạ (Reflection)

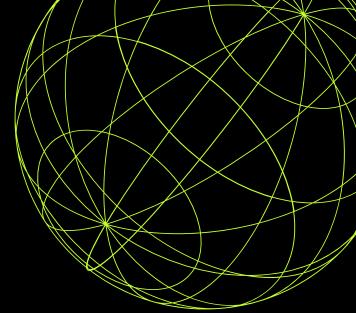


## Tham số

- Tọa độ khởi đầu của đối tượng  $O$  ( $X, Y, Z$ )
- Tọa độ mới của đối tượng  $O$  sau khi phản xạ ( $X', Y', Z'$ )



# Phép phản xạ (Reflection)



Phản xạ qua mặt phẳng XY

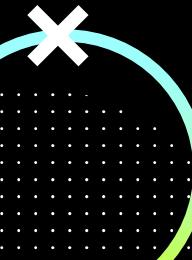
$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

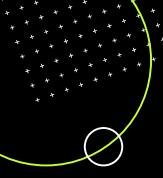
Phản xạ qua mặt phẳng YZ

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

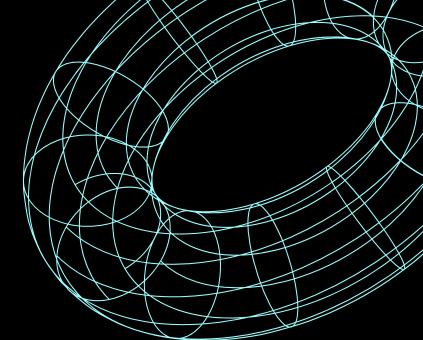
Phản xạ qua mặt phẳng XZ

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$





# Nội Dung



**01** Phép tịnh tiến (Translation)

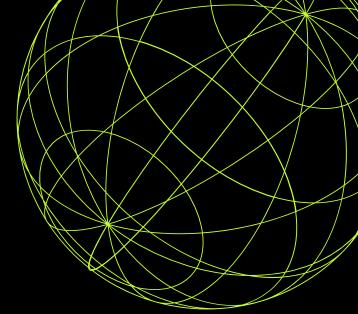
**02** Phép quay (Rotation)

**03** Phép tỉ lệ (Scaling)

**04** Phép phản xạ (Reflection)

**05** Phép cắt (Shear)

# Phép cắt (Shear)

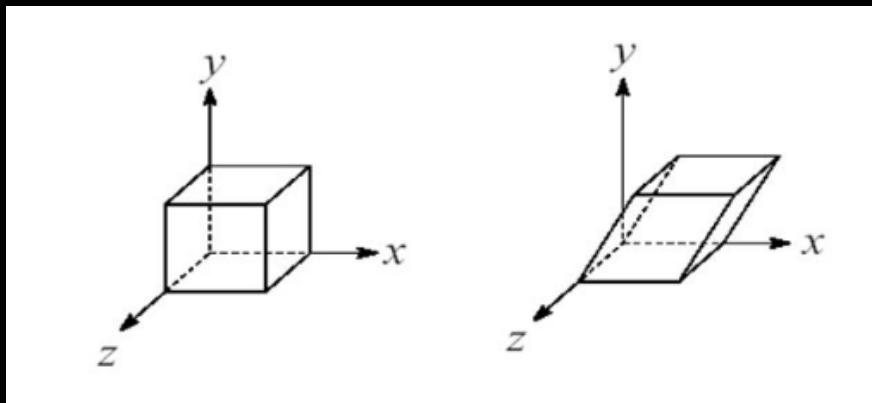


## Khái niệm

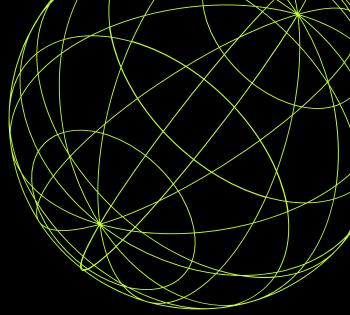
- Cắt 3D là một kỹ thuật lý tưởng để thay đổi hình dạng của một đối tượng hiện có trong mặt phẳng 3 chiều

## Các loại phép cắt

- Cắt theo hướng X
- Cắt theo hướng Y
- Cắt theo hướng Z

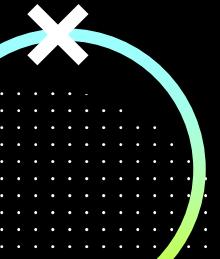


# Phép cắt (Shear)

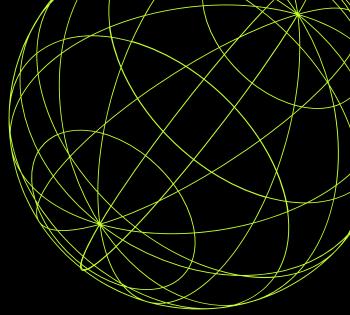


## Tham số

- Tọa độ khởi đầu của đối tượng  $O$  ( $X, Y, Z$ )
- Tọa độ mới của đối tượng  $O$  sau khi cắt ( $X', Y', Z'$ )
- Tham số cắt theo các hướng  $X, Y, Z$  ( $S_X, S_Y, S_Z$ )



# Phép cắt (Shear)



## Phương trình cắt theo trục X

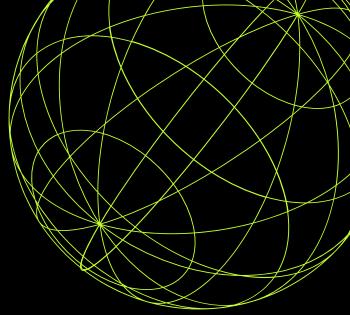
- $X' = X$
- $Y' = Y + S_Y \times X$
- $Z' = Z + S_Z \times X$

## Ma trận cắt theo trục X

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ S_Y & 1 & 0 & 0 \\ S_Z & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$



# Phép cắt (Shear)



## Phương trình cắt theo trục Y

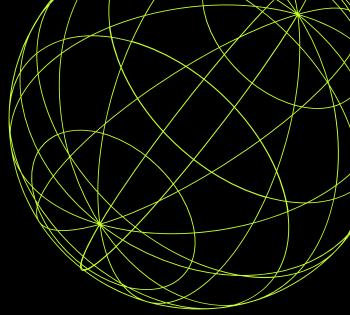
- $X' = X + S_X \times Y$
- $Y' = Y$
- $Z' = Z + S_Z \times Y$

## Ma trận cắt theo trục Y

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & S_X & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & S_Z & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$



# Phép cắt (Shear)



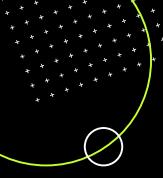
## Phương trình cắt theo trục Z

- $X' = X + S_X \times Z$
- $Y' = Y + S_Y \times Z$
- $Z' = Z$

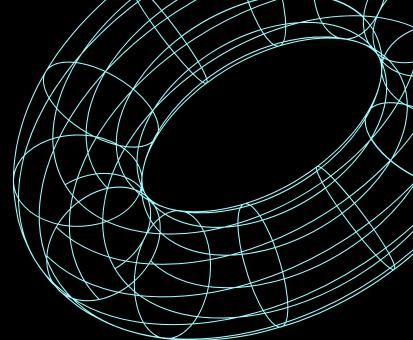
## Ma trận cắt theo trục Z

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & S_X & 0 \\ 0 & 1 & S_Y & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$





# Nội Dung



## Perspective Projection

# Perpective Projection



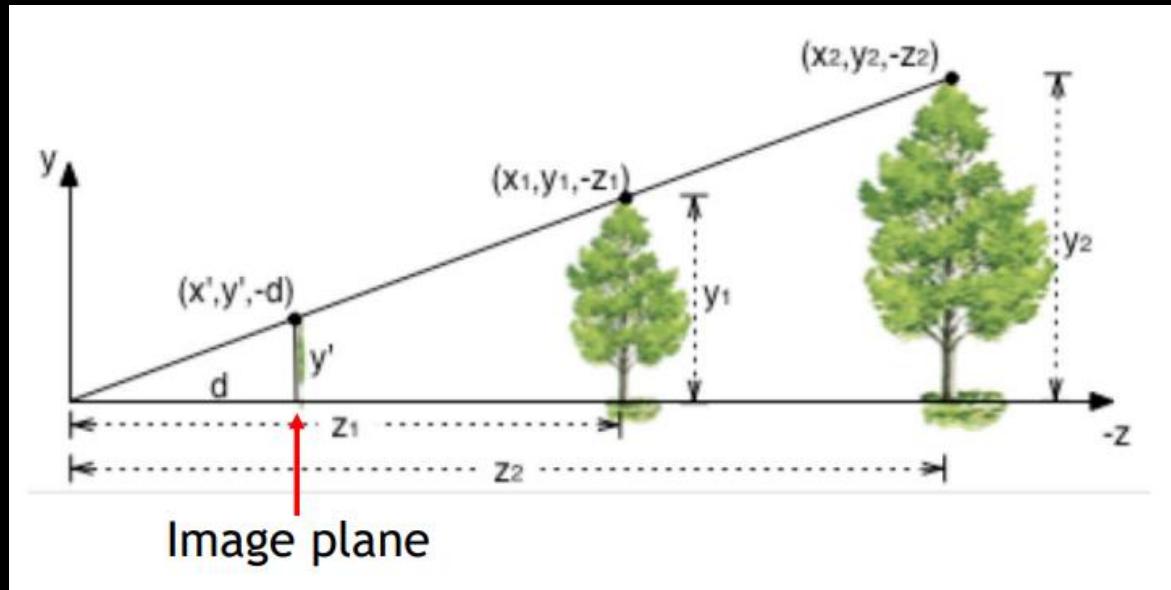
Các đường thẳng không còn song song,  
hội tụ tại 1 điểm



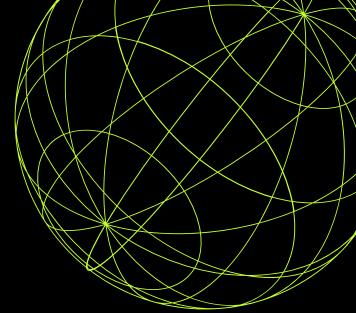
# Perspective Projection

Phương trình perspective projection

- $y' = \frac{yd}{z_1}$
- $x' = \frac{xd}{z_1}$
- $z' = d$



# Perspective Projection

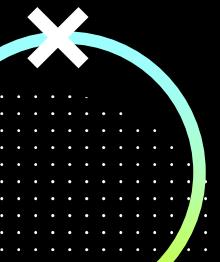


Phương trình perspective projection

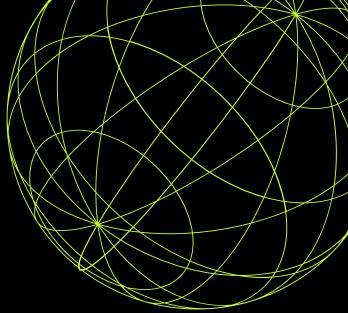
- $y' = \frac{yd}{z}$
- $x' = \frac{xd}{z}$
- $z' = d$

Ma trận perspective projection

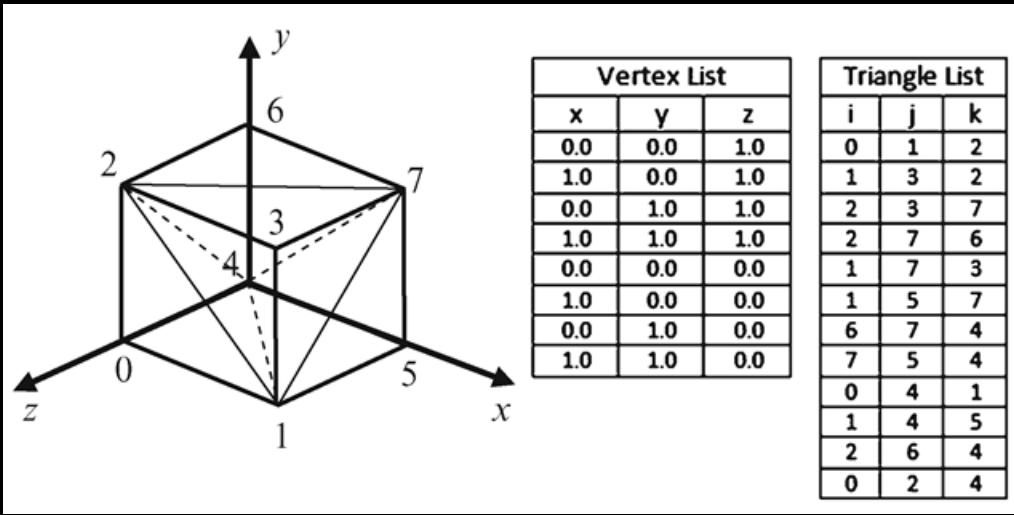
$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/d & 0 \end{bmatrix} \times \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \\ Z/d \end{bmatrix} = \begin{bmatrix} Xd/Z \\ Yd/Z \\ d \\ 1 \end{bmatrix}$$



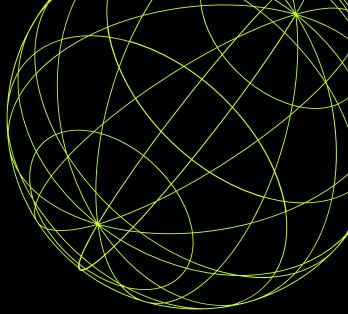
# Cấu trúc mesh (.obj)



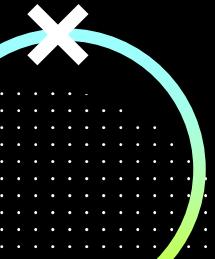
- Mesh là một biểu diễn đối tượng 3D
- Danh sách đỉnh (vertices) chứa tọa độ 3 chiều của các đỉnh
- Danh sách đa giác (faces) chứa các giá trị số nguyên chỉ mục của các đỉnh



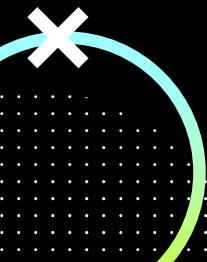
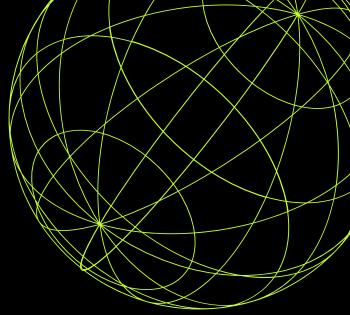
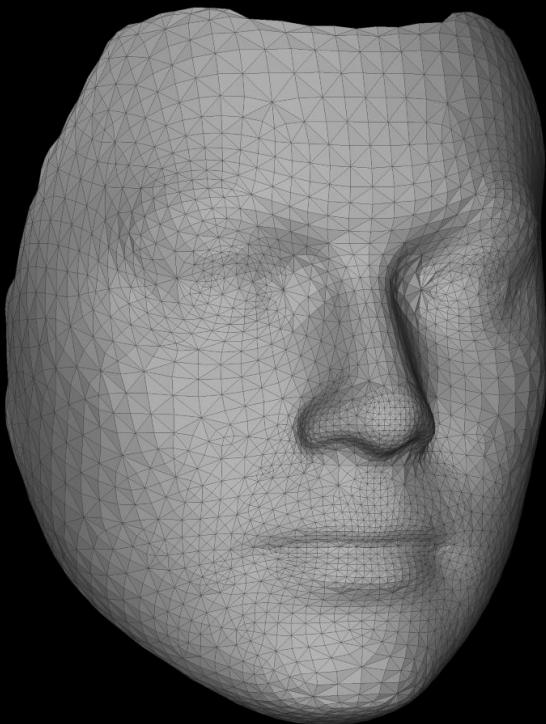
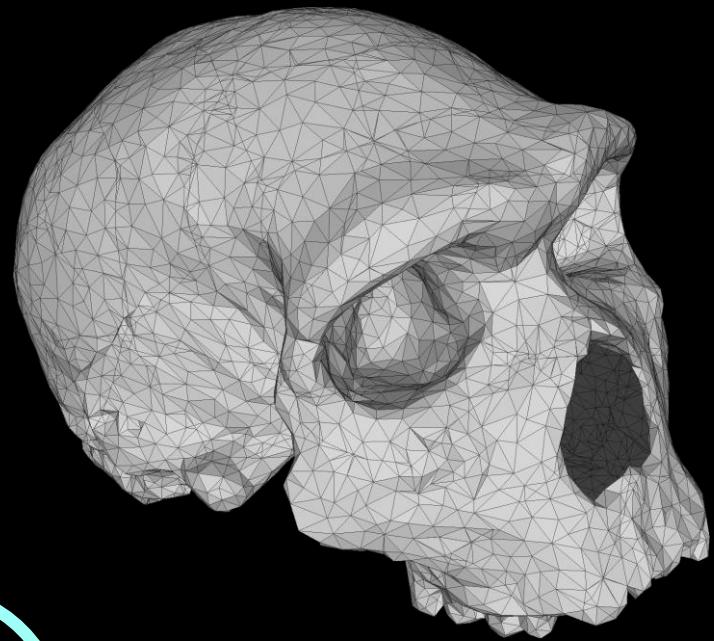
# Cấu trúc mesh (.obj)



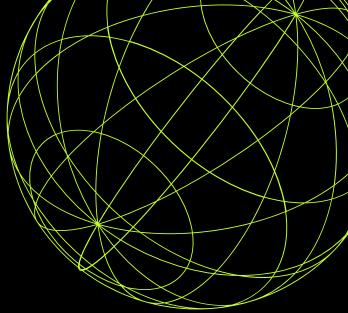
```
# OBJ file format with ext .obj
# vertex count = 2503
# face count = 4968
...
v -1.3895490e-002 1.6787168e-001 -2.1897230e-002
v -6.9413000e-002 1.5121847e-001 -4.4538540e-002
v -5.5039800e-002 5.7309700e-002 1.6990900e-002
f 1069 1647 1578
f 1058 909 939
f 421 1176 238
...
```



# Cấu trúc mesh (.obj)



# Cấu trúc mesh (.obj)

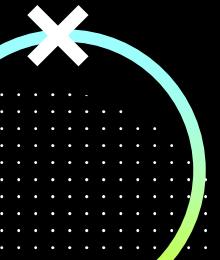


## Ưu điểm

- Có thể mô hình hóa hầu hết các đối tượng
- Dễ biểu diễn dưới dạng một tập các điểm
- Dễ dàng biến đổi (transformation)
- Dễ dàng để vẽ trên máy tính

## Nhược điểm

- Các bề mặt cong chỉ được mô tả gần đúng
- Khó mô phỏng một số loại vật thể như tóc, chất lỏng, ...



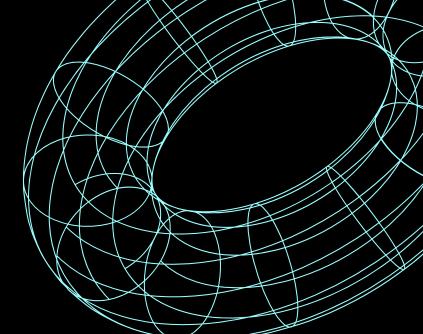
# 03

## Bài toán

Các bài toán áp dụng trên đối tượng hình học trong  
không gian 3D



# Nội Dung

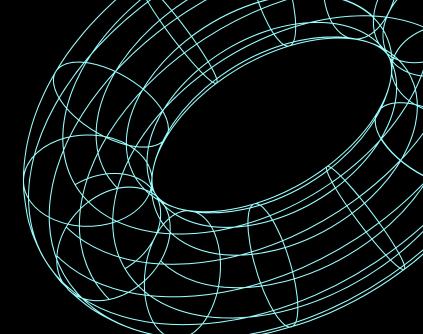


**01** Object Convex-hull

**02** 3D reconstruction



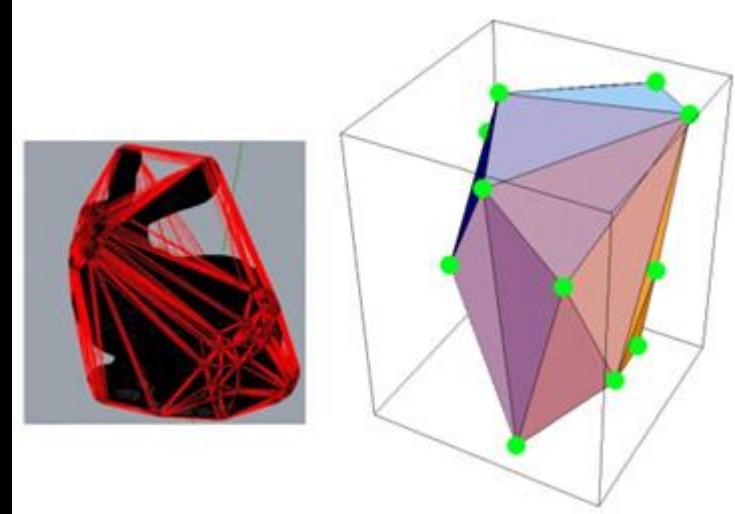
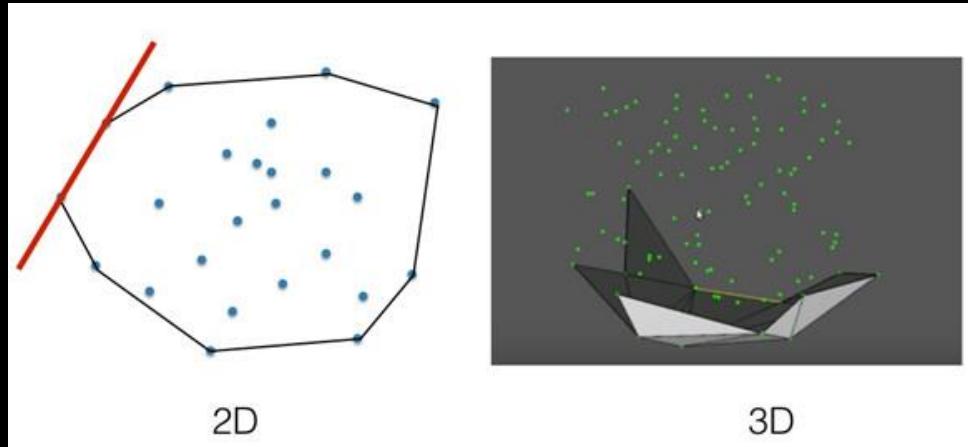
# Nội Dung



01 Object Convex-hull

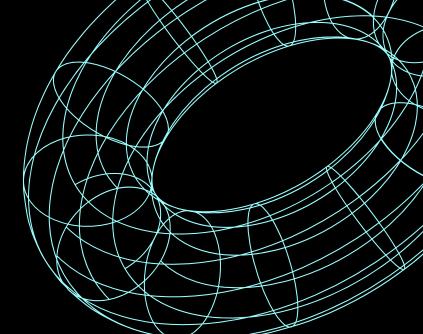
02 3D reconstruction

# Object Convex-hull





# Nội Dung

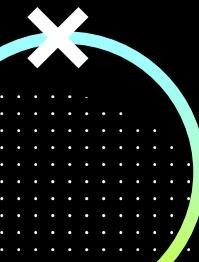
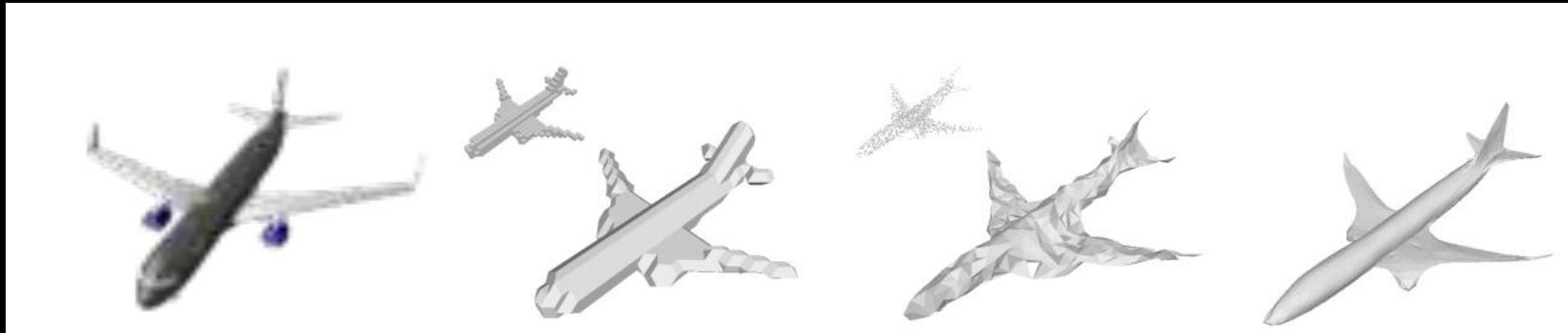


01 Object Convex-hull

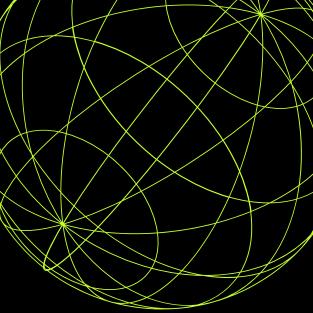
02 3D Reconstruction

# 3D Reconstruction

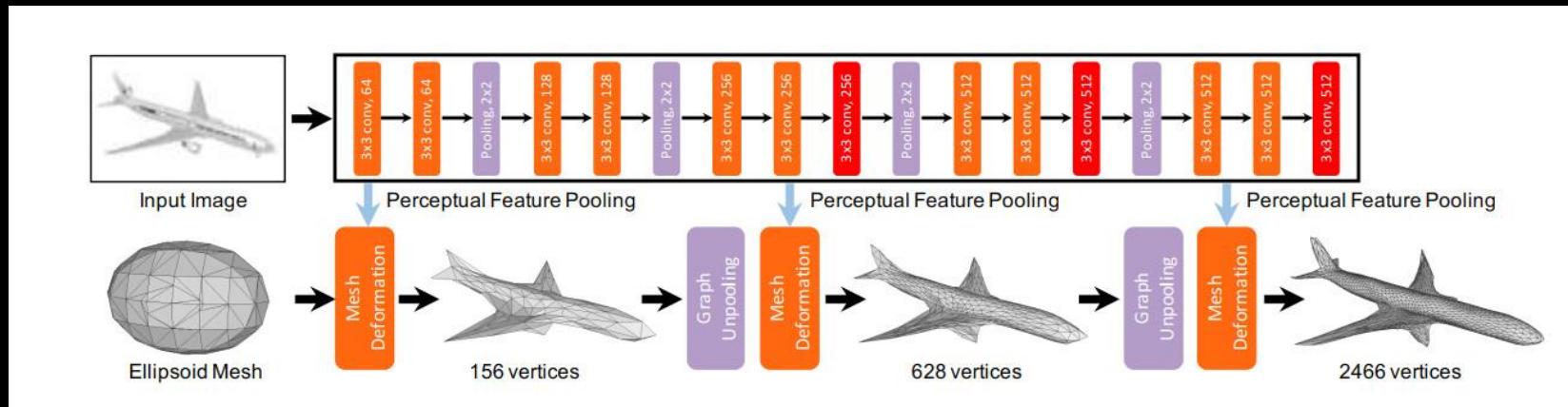
- Ảnh màu (input) + mesh ban đầu → high-quality mesh



# 3D Reconstruction

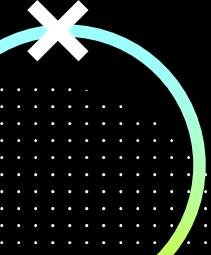
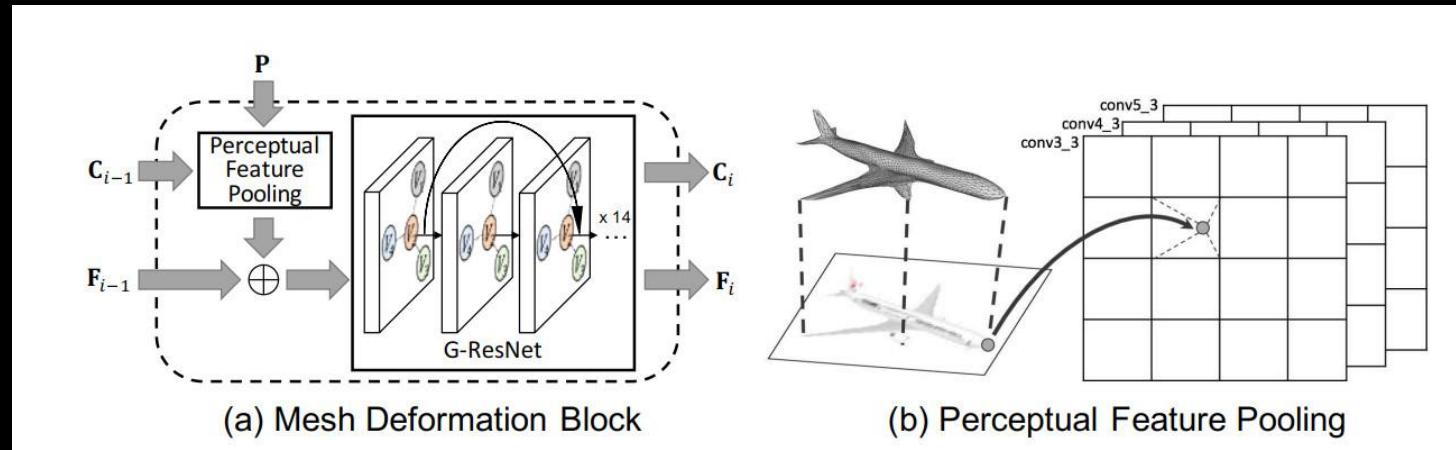
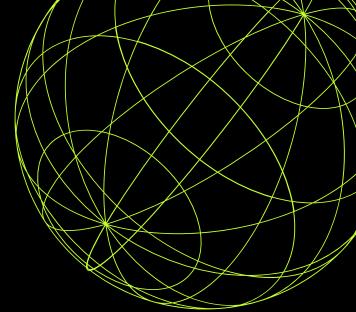


- 3 mesh deformation blocks

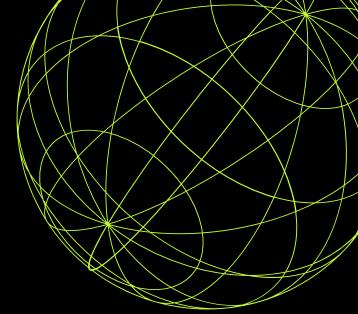


The cascaded mesh deformation network

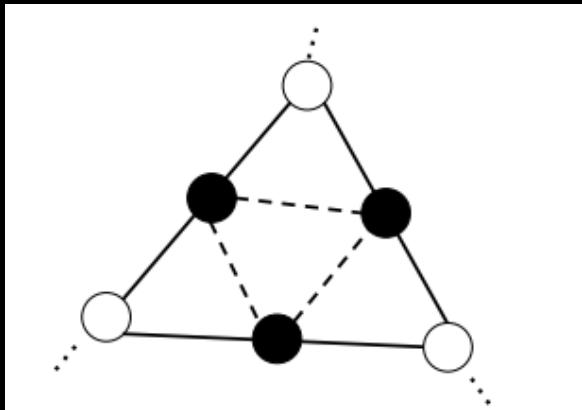
# 3D Reconstruction



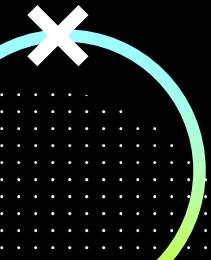
# 3D Reconstruction



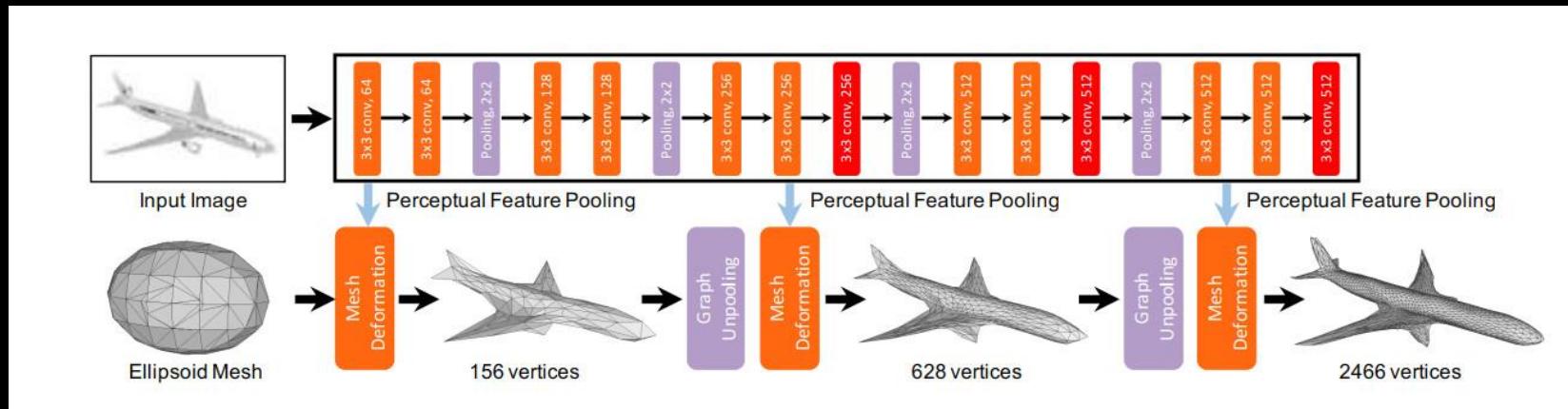
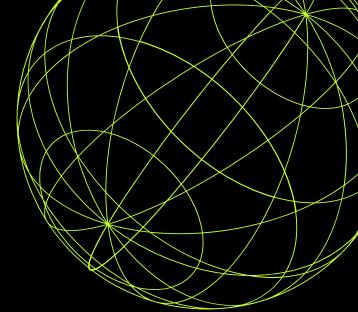
- Đỉnh đen và cạnh (nét đứt) được thêm trong unpooling layer



Graph Unpooling

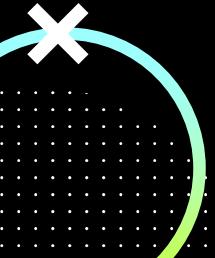
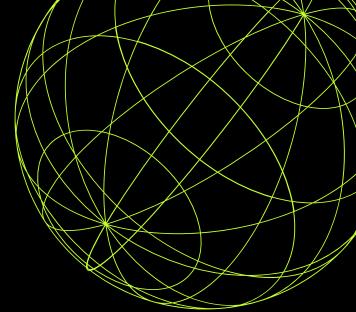


# 3D Reconstruction



The cascaded mesh deformation network

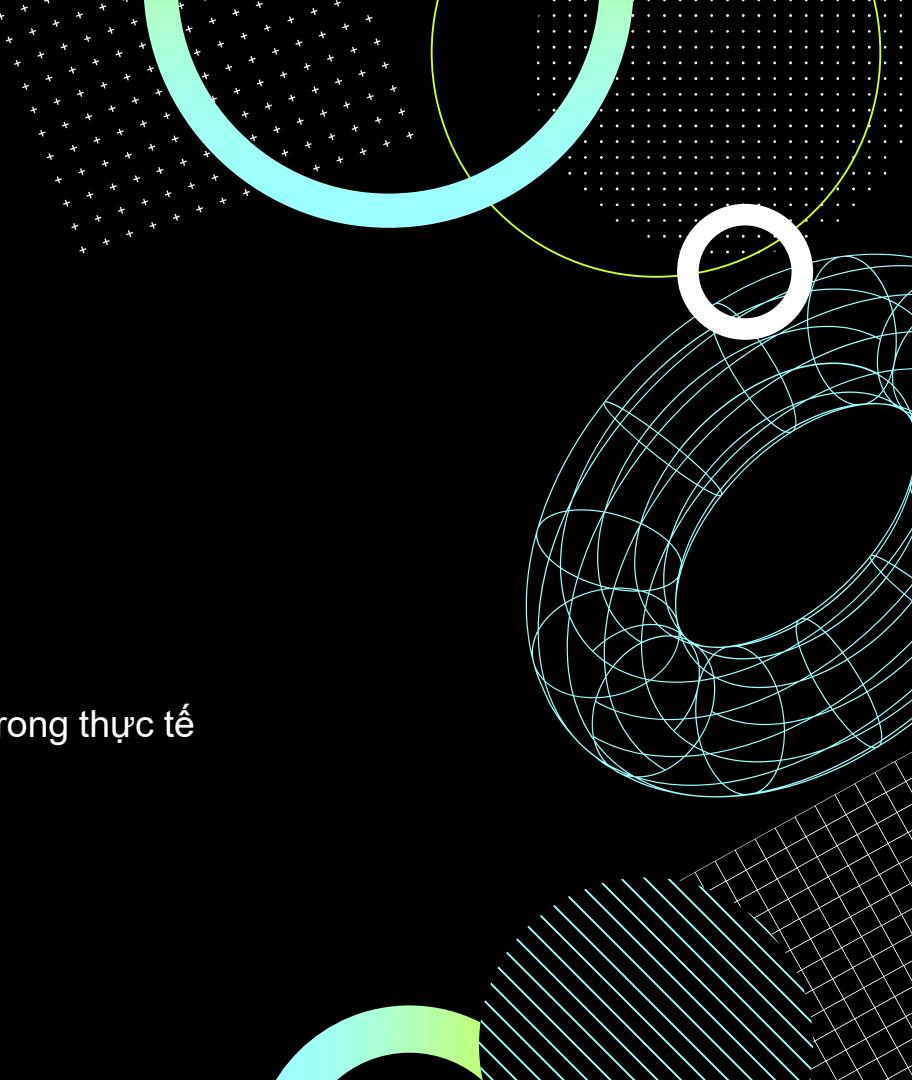
# 3D Reconstruction



# 04

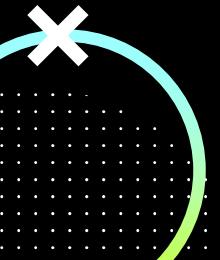
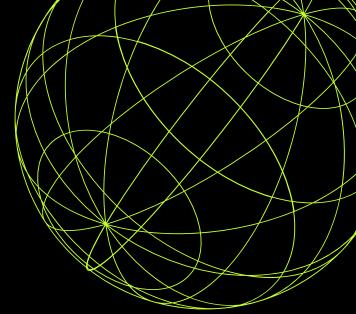
## Ứng dụng

Ứng dụng của đối tượng hình học không gian 3D trong thực tế

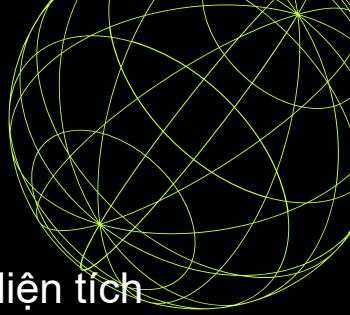


# Applications

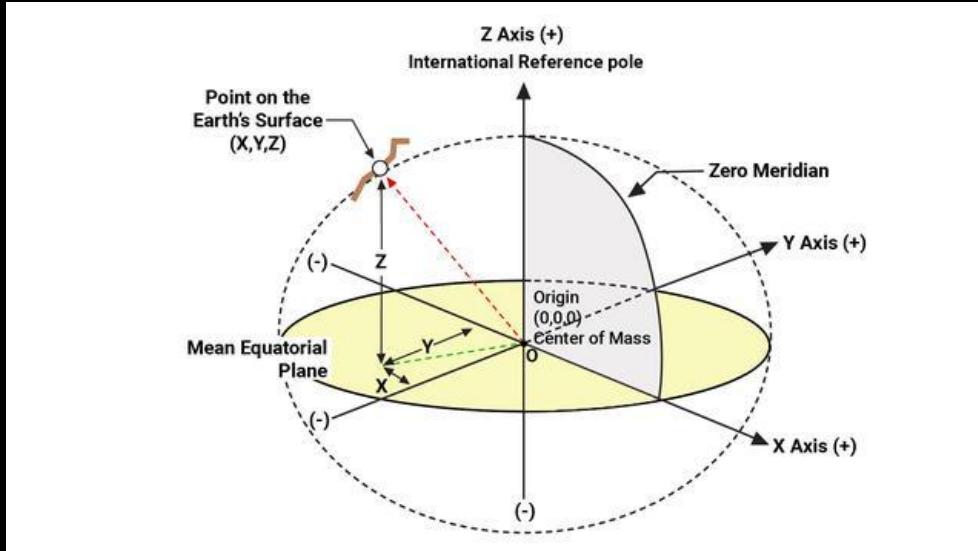
- Location of point on Earth
- Tracking aircrafts
- Architecture
- Medicine
- MRI Scanning
- Technology



# Location of point on Earth



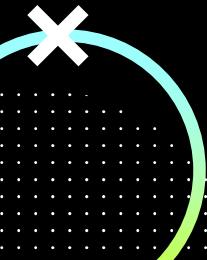
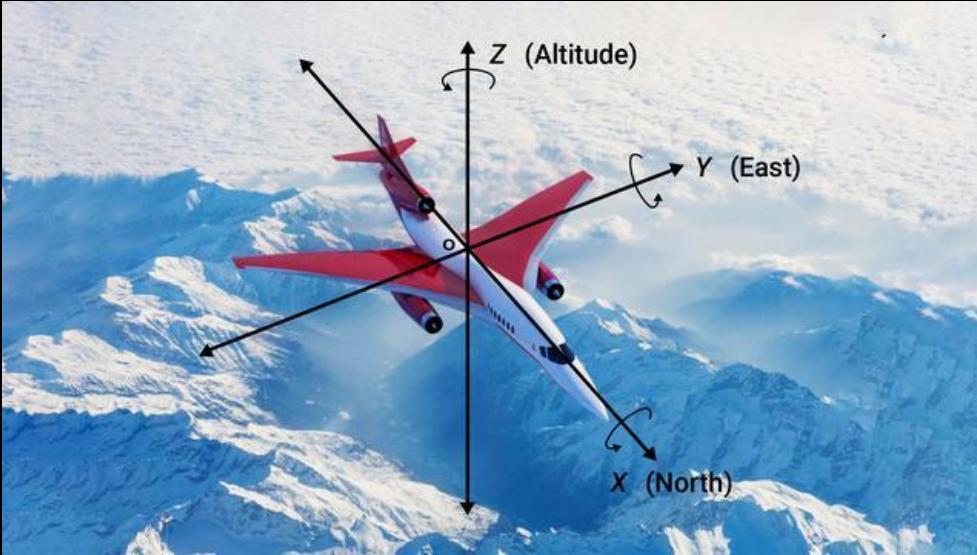
- Tọa độ ba chiều được sử dụng để xác định vị trí của điểm/vị trí trên diện tích bề mặt trái đất.



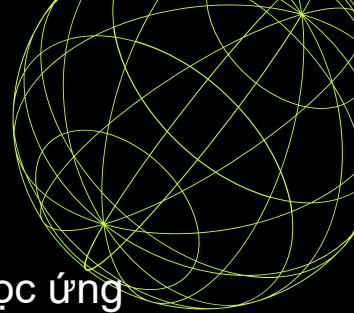
# Tracking aircrafts



- Hệ tọa độ ba chiều đóng vai trò chính trong việc theo dõi vị trí của các chuyến bay/máy bay.



# Architecture



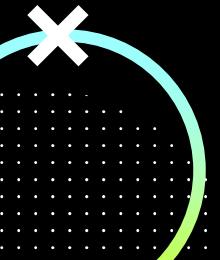
- Hình học kiến trúc là nghiên cứu sự kết hợp giữa kiến trúc và hình học ứng dụng.
- Hình học 3D đóng một vai trò quan trọng trong việc quyết định hình dạng, kích thước của chúng và tất cả các quyết định chiến lược và kỹ thuật trong khi xây dựng tòa nhà, cầu và bất kỳ công trình nào khác.



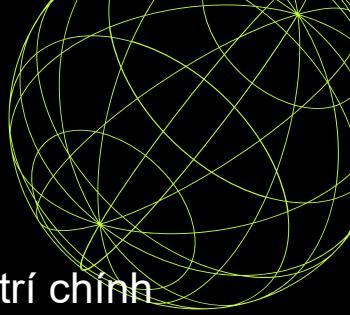
# Medicine



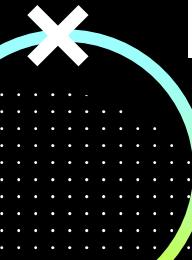
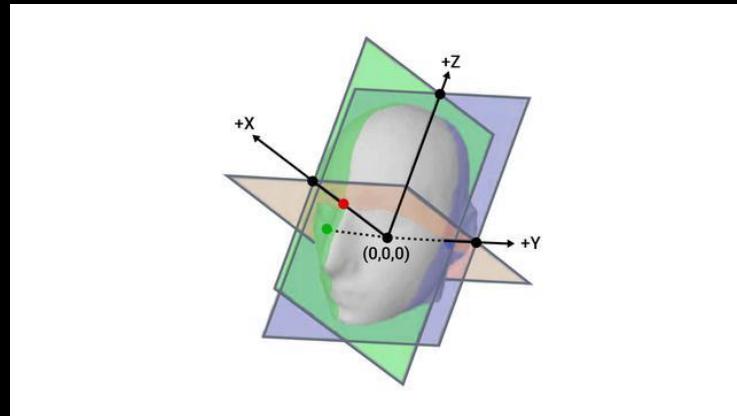
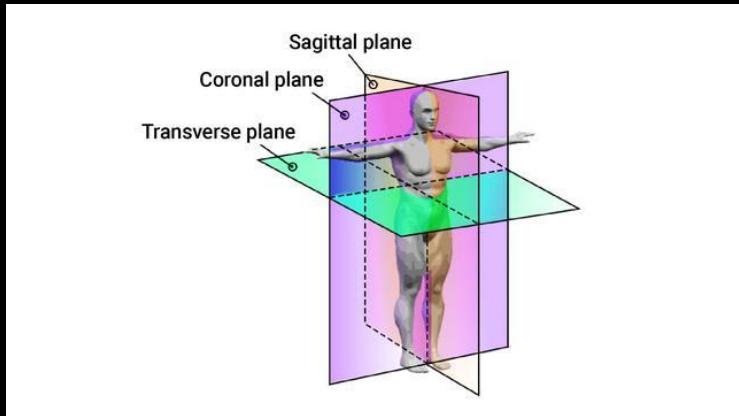
- Hình học 3D được sử dụng để tìm ra các quyết định khác nhau của các bác sĩ liên quan đến phẫu thuật cơ thể, phương pháp điều trị và sản xuất mô cấy và bộ phận giả. Việc xem xét thể tích, diện tích và độ dài của bộ phận giả theo yêu cầu của cơ thể con người được thực hiện bằng cách sử dụng hình học 3D.



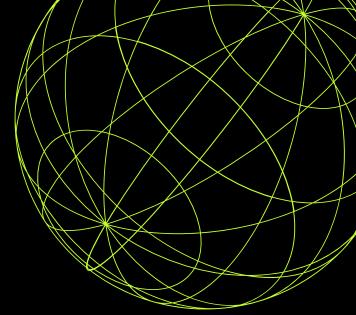
# MRI Scanning



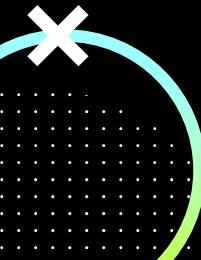
- Quét MRI sử dụng hệ tọa độ ba chiều để quét cơ thể và đánh dấu vị trí chính xác của vấn đề trong cơ thể để đưa ra phương pháp điều trị chính xác và tốt hơn.



# Technology



- Hình học ba chiều có nhiều ứng dụng trong phần mềm máy tính.
- Ví dụ: Trò chơi điện tử có cấu trúc tòa nhà và các đối tượng khác được thiết kế bằng hình học ba chiều.



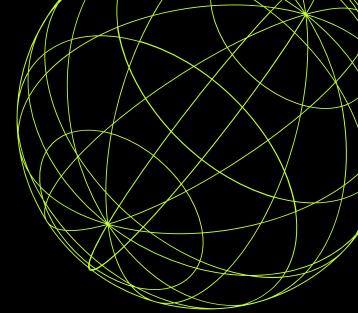
The background features abstract geometric elements: a large white circle in the bottom left, a grid of white lines forming a cone shape, several green spheres of varying sizes, and a series of yellow lines forming a curved surface in the top right.

# 05

## Thực hành

Demo biểu diễn các đối tượng hình học trong không gian 3D

# Tài liệu tham khảo



## Point, line

- <https://www.cuemath.com/geometry/points-and-lines/>

## Vector

- <https://byjus.com/math/vectors/>

## Plane

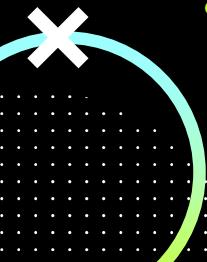
- <https://study.com/academy/lesson/what-is-a-plane-in-geometry-definition-examples.html>

## Linear-transformation

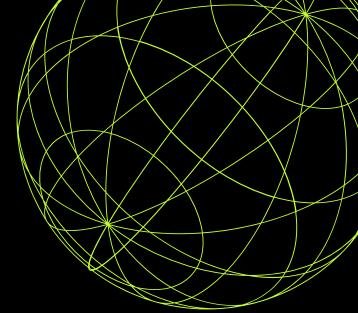
- <https://medium.com/analytics-vidhya/advance-nonlinear-variable-transformations-ececf3f8709e>

## Application

- <https://www.toppr.com/ask/content/posts/three-dimensional-geometry/real-life-applications-28247/>
- <https://www.toppr.com/ask/content/posts/introduction-to-three-dimensional-geometry/real-life-applications-33635/>



# Tài liệu tham khảo



## Translation

- <https://www.gatevidyalay.com/3d-translation-in-computer-graphics-definition-examples/>

## Rotation

- <https://www.gatevidyalay.com/3d-rotation-in-computer-graphics-definition-examples/>

## Scaling

- <https://www.gatevidyalay.com/3d-scaling-in-computer-graphics-definition-examples/>

## Reflection

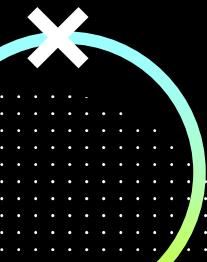
- <https://www.gatevidyalay.com/3d-reflection-in-computer-graphics-definition-examples/>

## Shear

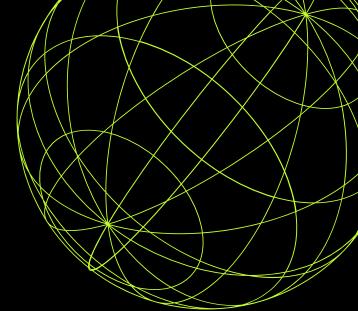
- <https://www.gatevidyalay.com/3d-shearing-in-computer-graphics-definition-examples/>

## 3D Reconstruct

- [https://openaccess.thecvf.com/content\\_ECCV\\_2018/html/Nanyang\\_Wang\\_Pixel2Mesh\\_Generating\\_3D\\_ECCV\\_2018\\_paper.html](https://openaccess.thecvf.com/content_ECCV_2018/html/Nanyang_Wang_Pixel2Mesh_Generating_3D_ECCV_2018_paper.html)



# Resources



## Github

- [https://github.com/HilmKing1509/CS532.N11\\_3D\\_Geometry\\_Objects](https://github.com/HilmKing1509/CS532.N11_3D_Geometry_Objects)

Kiệt Huỳnh Việt Tuấn final

8d88577 1 hour ago 8 commits

No description, website, or topics provided.

Readme

0 stars

1 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

Languages

Jupyter Notebook 100.0%

DAI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN  
**VNUHCM - UIT**

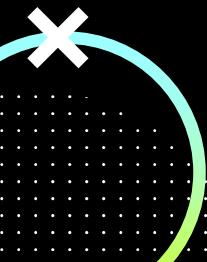
CS531.N11 - 3D GEOMETRIC OBJECTS

MEMBERS

Serial	student ID	Full name	Github	Email
1	20521494	Huỳnh Việt Tuấn Kiệt	hilmking1509	20521494@gm.uit.edu.vn
1	20520276	Nguyễn Đức Anh Phúc	PhucNDA	20520276@gm.uit.edu.vn

SET UP REQUIREMENTS

- JupyterNotebook in `Vscode`
- `conda install -c conda-forge igl`
- `conda install -c conda-forge meshplot`



# THANK YOU FOR YOUR ATTENTION

Huỳnh Viết Tuấn Kiệt  
Nguyễn Đức Anh Phúc