



# **Bài 1:** **Giới thiệu bài toán** **Semantic Segmentation**

AI Research Team - Sun\* AI Research

# Nội dung chính

01

**Giới thiệu bài  
toán**

02

**Giới thiệu thư viện  
Segmentation  
Models PyTorch**



01

# Giới thiệu bài toán

# Bài toán phân vùng ảnh

## Image Segmentation

**Classification**



**CAT**

**Semantic Segmentation**



**GRASS, CAT,  
TREE, SKY**

**Object Detection**



**DOG, DOG, CAT**

**Instance Segmentation**



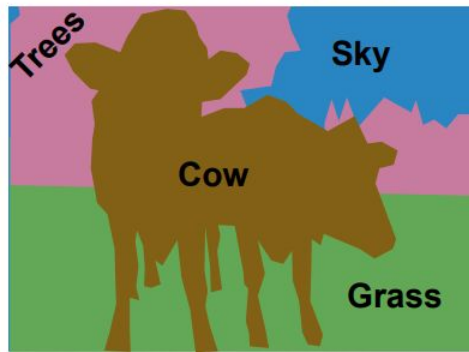
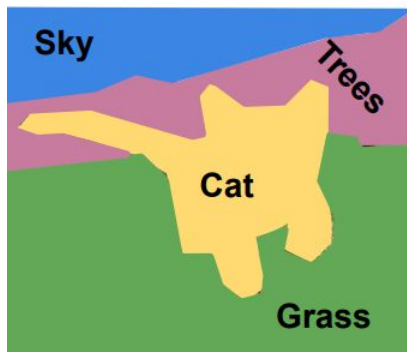
**DOG, DOG, CAT**

- Là một bài toán rất quan trọng trong thị giác máy tính.
- Nhiệm vụ của bài toán: phân loại từng pixel trong ảnh.
- Semantic Segmentation, Instance Segmentation, Panoptic Segmentation

Nguồn CS231n - Stanford

# Bài toán Semantic Segmentation

- Phân loại từng pixel trong ảnh
- Không quan tâm tới các đối tượng khác nhau trong ảnh, chỉ quan tâm tới việc phân loại từng pixel
- Nhiều kỹ thuật để giải quyết bài toán: Các phương pháp phân vùng truyền thống + Deep Learning



Nguồn CS231n - Stanford

# Bài toán Semantic Segmentation



0: Background/Unknown

1: Person

2: Purse

3: Plants/Grass

4: Sidewalk

5: Building/Structures

Nguồn <https://www.jeremyjordan.me/semantic-segmentation/>

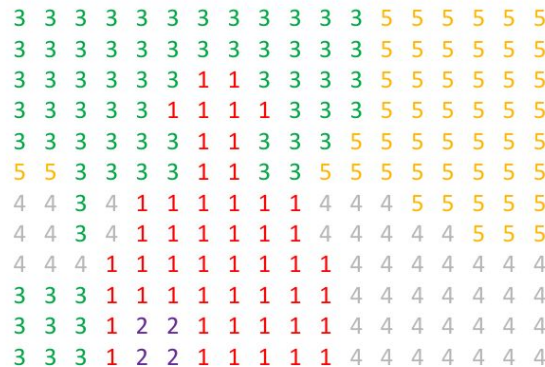
# Tập dữ liệu Semantic Segmentation



Input



1: Person  
2: Purse  
3: Plants/Grass  
4: Sidewalk  
5: Building/Structures



Semantic Labels

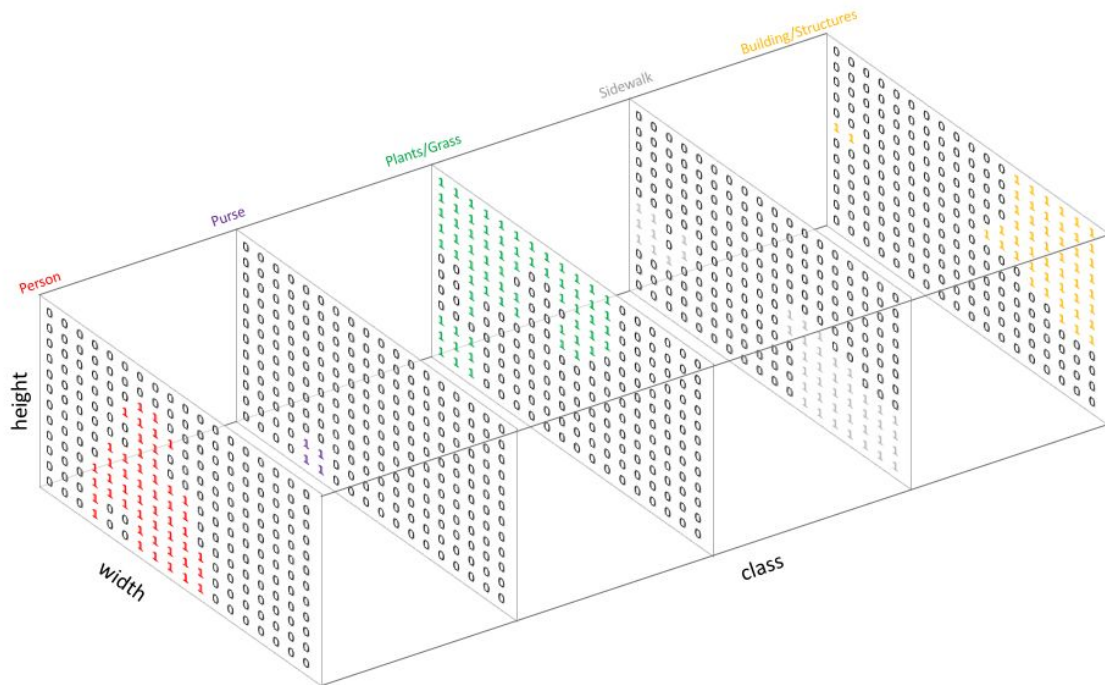
- CityScapes, ADE20K, ...
- $(X, y) = (\text{Image}, \text{Mask})$
- Mask thường là ảnh RGB mô tả kết quả phân vùng, cần viết code đọc ảnh Mask và chuyển ảnh Mask về dạng ma trận 2 chiều (HxW)
- Các giá trị trong ma trận 2 chiều đó ứng với các class tương ứng với từng pixel

Nguồn <https://www.jeremyjordan.me/semantic-segmentation/>

# Đầu ra bài toán Semantic Segmentation

## Đối với mô hình Deep Learning

- Input: Image (H, W, 3)
- Output: Mask (H, W, C)



Nguồn <https://www.jeremyjordan.me/semantic-segmentation/>



# Đánh giá kết quả bài toán Semantic Segmentation

0	0	0	0	0	0	0	0	0	0
0	1	1	1	1	1	0	0	0	0
0	1	1	1	1	1	0	0	0	0
0	1	1	1	1	1	0	0	0	0
0	0	1	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

**Ground Truth**

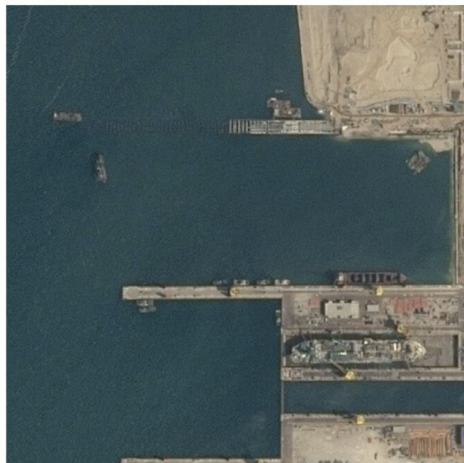
0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	1	1	1	1	1	0	0	0	0
0	0	1	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

**Predict**

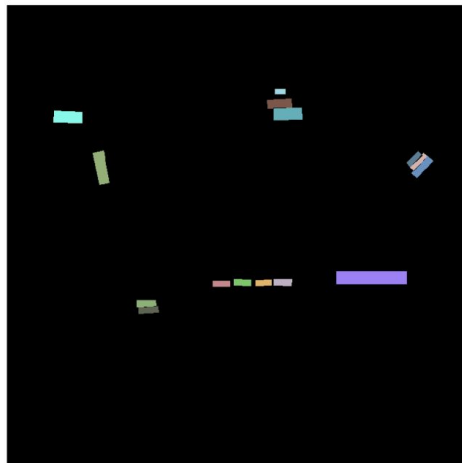
- Ảnh kích thước 10x10
- Nếu sử dụng độ đo là độ chính xác =  $\frac{\text{Số điểm ảnh dự đoán đúng}}{\text{Tổng số điểm ảnh}} = \frac{94}{100} = 94\%$
- Không hợp lý => Cần có độ đo khác phù hợp hơn

# Đánh giá kết quả bài toán Semantic Segmentation

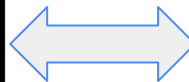
**Pixel Accuracy = Số điểm ảnh dự đoán đúng / Tổng số điểm ảnh**



Input



Ground Truth



>90%

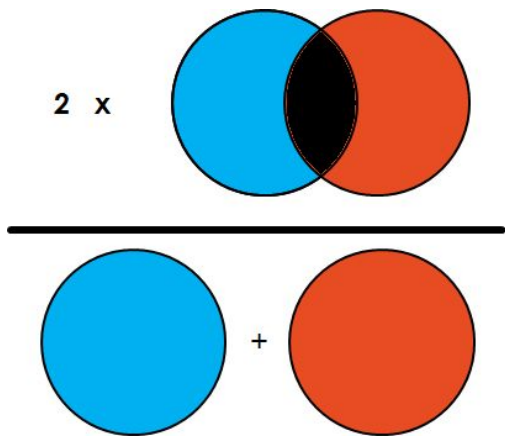


Output

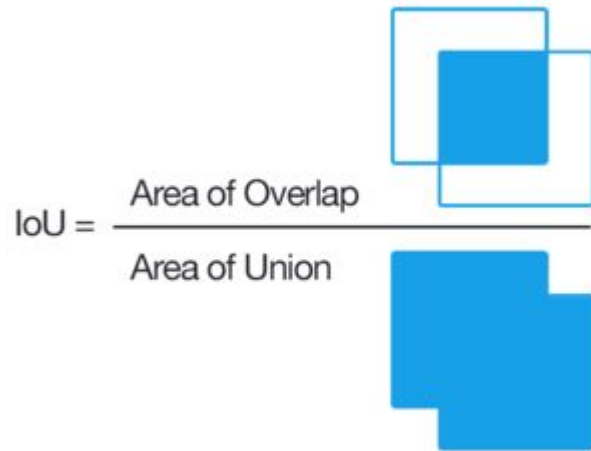
Nguồn: <https://towardsdatascience.com/metrics-to-evaluate-your-semantic-segmentation-model-6bcb99639aa2>

# Đánh giá kết quả bài toán Semantic Segmentation

## Dice Score & IoU



$$DSC = \frac{2|X \cap Y|}{|X| + |Y|}$$



$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

Nguồn: <https://towardsdatascience.com/metrics-to-evaluate-your-semantic-segmentation-model-6bcb99639aa2>

# Đánh giá kết quả bài toán Semantic Segmentation

0	0	0	0	0	0	0	0	0	0
0	1	1	1	1	1	0	0	0	0
0	1	1	1	1	1	0	0	0	0
0	1	1	1	1	1	0	0	0	0
0	0	1	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

**Ground Truth**

0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	1	1	1	1	1	0	0	0	0
0	0	1	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

**Predict**

- Ảnh kích thước 10x10
- Dice Score =  $(2*11)/(17+11) = 78.5\%$
- IoU Score =  $11/17 = 64.7\%$



02

**Giới thiệu thư viện  
Segmentation Models  
PyTorch**

# Giới thiệu thư viện

- Thư viện dựa trên Framework PyTorch
- Dễ dàng sử dụng
- Tích hợp nhiều mô hình & backbone khác nhau
- Hỗ trợ nhiều hàm loss khác nhau cho bài toán segmentation, hỗ trợ các metrics phổ biến như IoU, Dice, ...



Python library with Neural Networks for Image Segmentation based on [PyTorch](#).



The main features of this library are:

- High level API (just two lines to create a neural network)
- 9 models architectures for binary and multi class segmentation (including legendary Unet)
- 119 available encoders (and 400+ encoders from [timm](#))
- All encoders have pre-trained weights for faster and better convergence
- Popular metrics and losses for training routines

# Nội dung khóa học

- Trình bày lý thuyết và hướng dẫn thực hành code PyTorch cho bài toán semantic segmentation
- Các mô hình deep learning trình bày trong khóa học chủ yếu dựa trên thư viện Semantic Segmentation Models PyTorch
- Kết quả thu được:
  - Nắm được lý thuyết về bài toán Semantic Segmentation
  - Vận dụng kiến thức + code được pipeline hoàn chỉnh dựa trên PyTorch



Python library with Neural Networks for Image Segmentation based on [PyTorch](#).



The main features of this library are:

- High level API (just two lines to create a neural network)
- 9 models architectures for binary and multi class segmentation (including legendary Unet)
- 119 available encoders (and 400+ encoders from [timm](#))
- All encoders have pre-trained weights for faster and better convergence
- Popular metrics and losses for training routines