



Instance Segmentation

PHÂN ĐOẠN CÁ THỂ BẰNG MẠNG HỌC SÂU MASK R-CNN

Nhóm RGB

- 1712718 – Huỳnh Thanh Sang
- 19120424 – Phan Nguyễn Thanh Tùng
- 19120106 – Nguyễn Ngọc Khôi Nguyên

Tổng quan

I. Động lực nghiên cứu

II. Phát biểu bài toán

III. Công trình liên quan

IV. Phương pháp

V. Ứng dụng

VI. Tài liệu tham khảo

I. Động lực nghiên cứu

I. Động lực nghiên cứu



Làm sao để **bao sát** các **cá thể** mong muốn
bất kể **hình dáng**, **kích thước** hay **màu sắc** ?



II. Phát biểu bài toán

II. Phát biểu bài toán



Phân đoạn cá thể (Instance Segmentation)

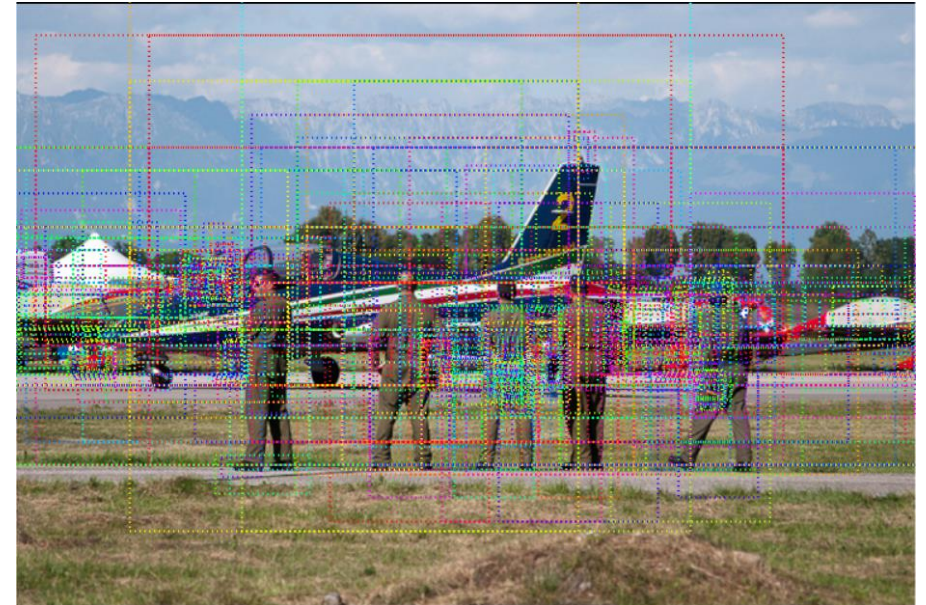
Input: Ảnh cần phân đoạn có kích thước $W \times H$.

Output: K kết quả phân đoạn:

- Class id (thuộc tập các class muốn phân đoạn)
- Bounding box $[cx, cy, w, h]$ (tọa độ tâm + kích thước)
- Binary mask có kích thước $W \times H$

(Source: https://github.com/matterport/Mask_RCNN)

II. 1. Phát sinh ứng viên (Region Proposal)



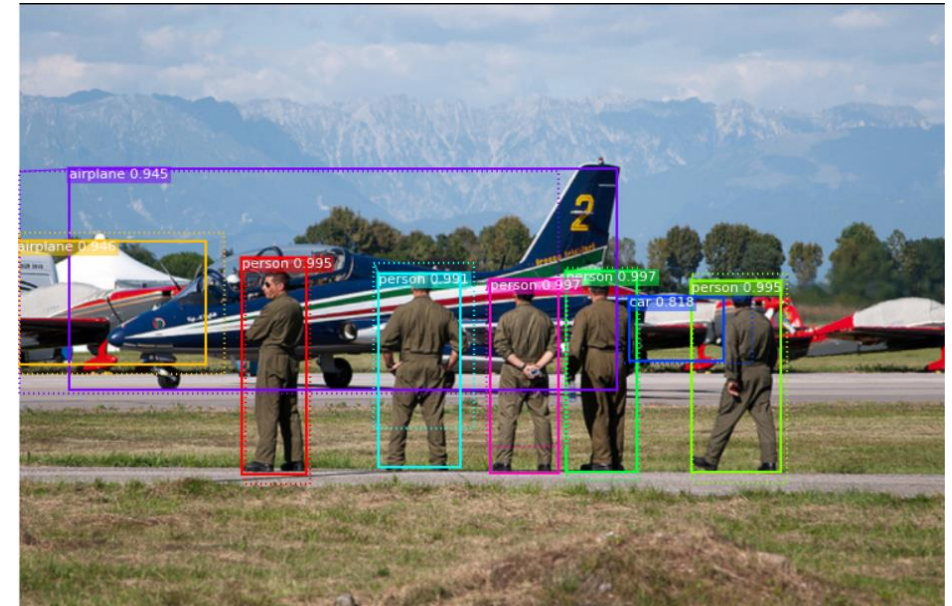
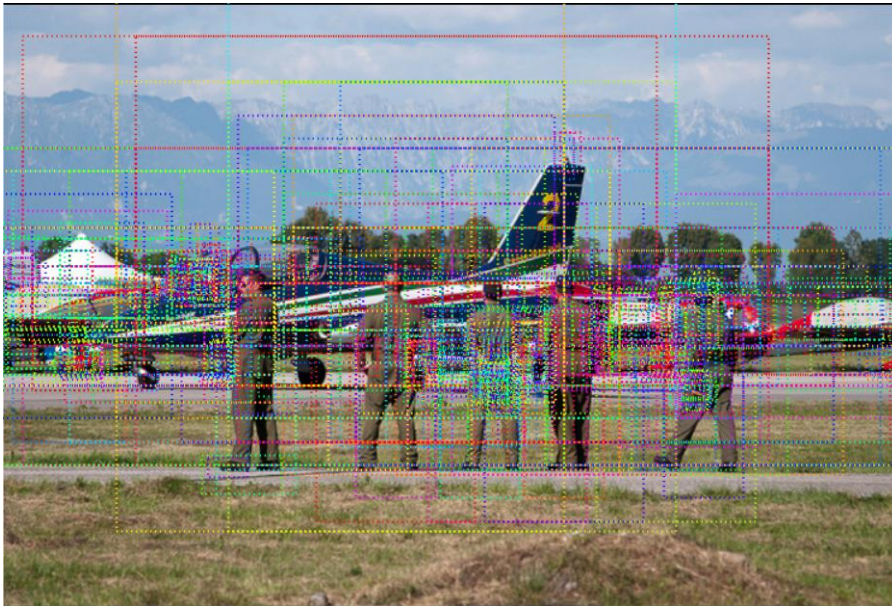
Phát sinh ứng viên (Region Proposal)

Input: Ảnh đầu vào có kích thước $W \times H$.

Output: Bounding box: $[cx, cy, w, h]$ tại những vị trí nghi ngờ có cá thể → **Region of Interest (ROI)**

(Source: https://github.com/matterport/Mask_RCNN)

II. 2. Phân lớp hình ảnh (Image Classification)



Phân lớp hình ảnh (Image Classification)

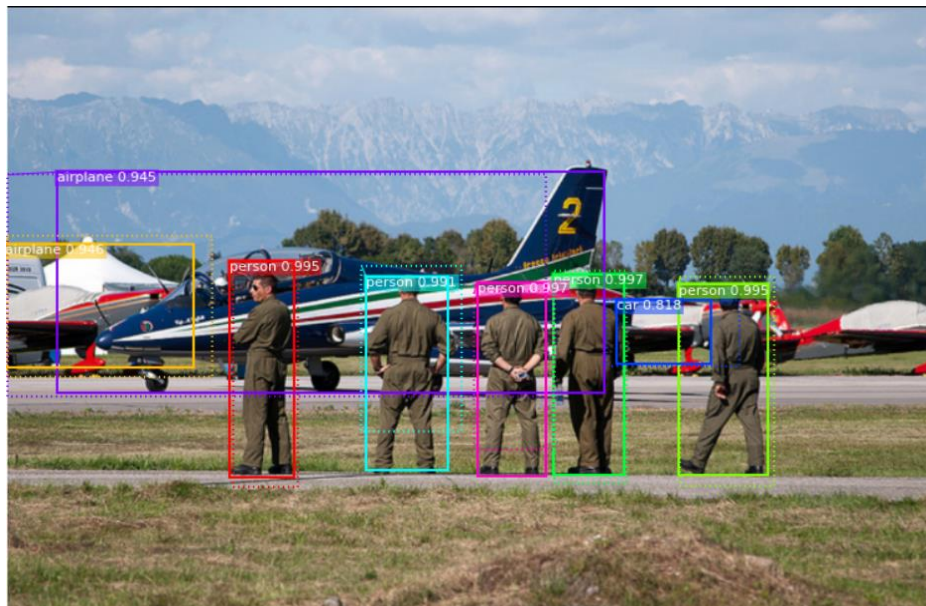
Input: Ảnh cần phân loại có kích thước $w \times h$ (có thể được cắt ra từ ảnh lớn theo vị trí bounding box).

Output: Class id tương ứng với ảnh (bao gồm background class).

-> Loại những ảnh có class id là background.

(Source: https://github.com/matterport/Mask_RCNN)

II. 3. Phân đoạn hình ảnh (Image Segmentation)



Phân đoạn hình ảnh (Image Segmentation)

Input: Ảnh cần phân đoạn có kích thước $w \times h$ (có thể được cắt ra từ ảnh lớn theo vị trí bounding box) + class id tương ứng.

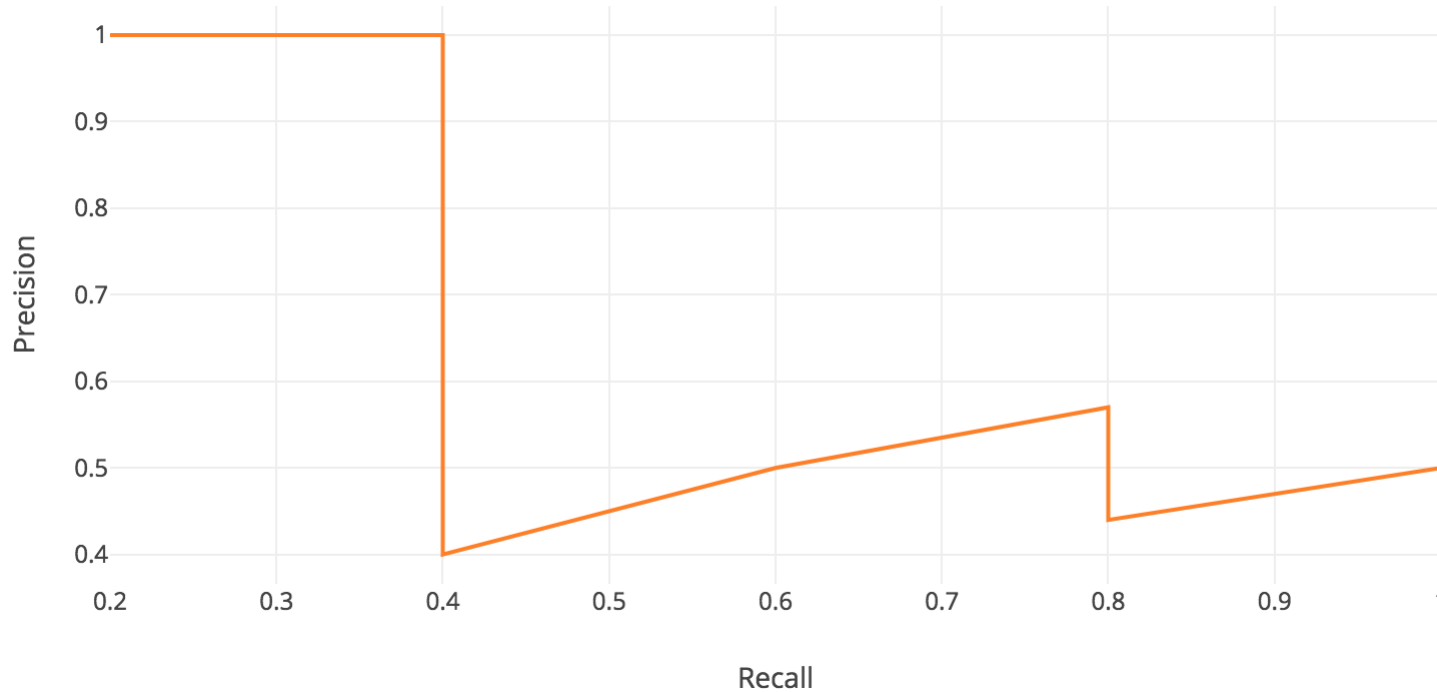
Output: Binary mask có kích thước $w \times h$ biểu diễn những pixel của cá thể thuộc class id input.

(Source: https://github.com/matterport/Mask_RCNN)

III. Công trình liên quan

Thước đo đánh giá: Mask AP và Frames per Second

Mask AP: diện tích của đồ thị biểu diễn *precision* và *recall* (càng cao càng tốt)




$$precision = \frac{TP}{\text{Tổng số mask dự đoán}}$$

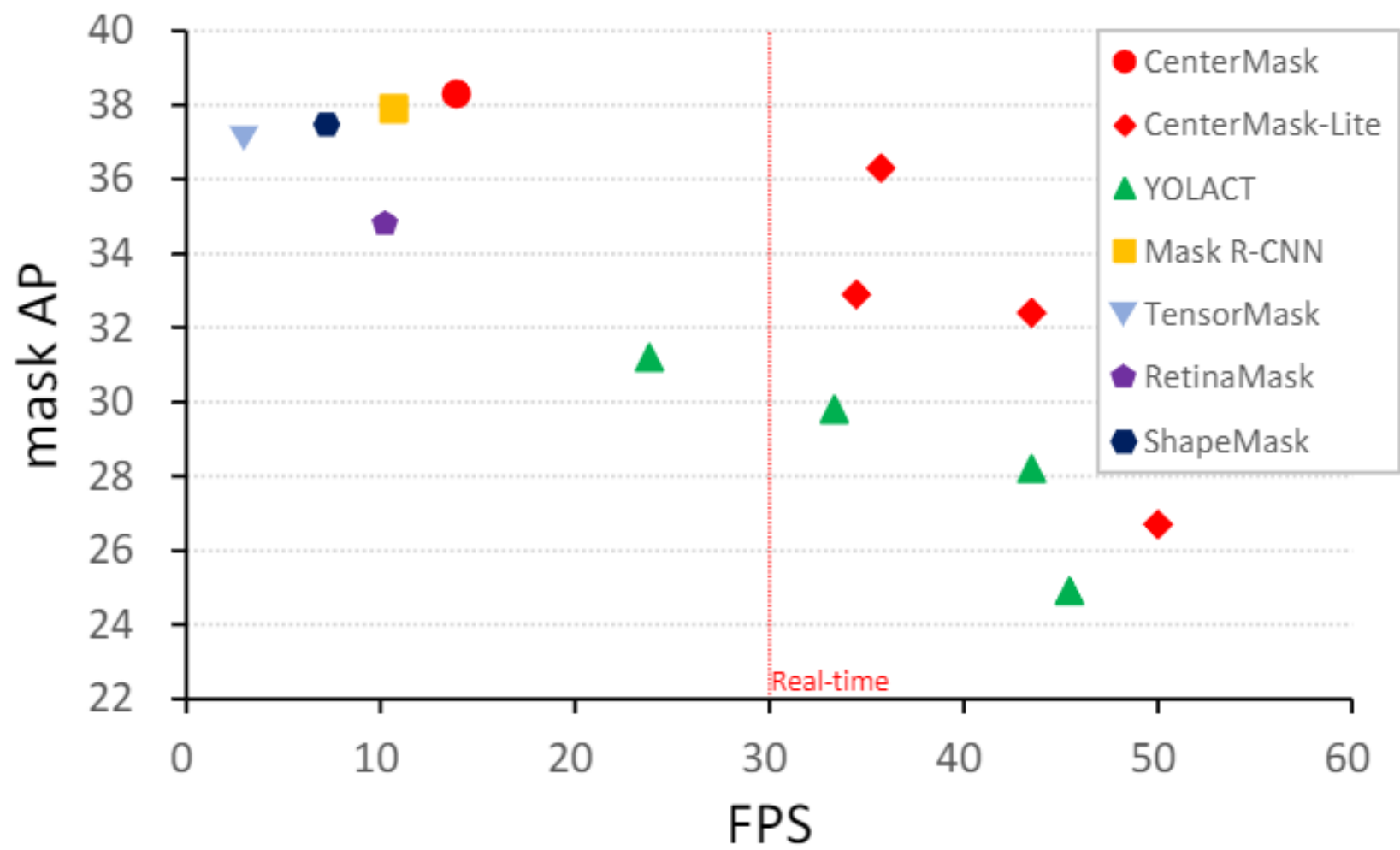
$$recall = \frac{TP}{\text{Tổng số mask groundtruth}}$$

Thước đo đánh giá: Mask AP và Frames per Second

TP (true positive) tăng 1 nếu mask dự đoán có:

- IoU với một mask groundtruth > 0.5.
- Score dự đoán lớn nhất trong tất cả các mask dự đoán có cùng groundtruth.

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$




(Source: CenterMask : Real-Time Anchor-Free Instance Segmentation (CVPR 2020))

IV. Phương pháp

Giai đoạn thực hiện

Offline:

- Chuẩn bị dữ liệu.
- Đưa dữ liệu vào để huấn luyện mô hình Mask RCNN.

Online:

- Tiền xử lý ảnh input (nếu cần thiết).
- Đưa input vào mô hình đã huấn luyện để phân đoạn cá thể.

Chuẩn bị dữ liệu



Cá thể muốn phân đoạn:

- Polyline bao sát -> Bounding box
- Class label



Càng đa dạng về **hình dáng**, **kích thước** và **màu sắc** càng tốt

Chuẩn bị dữ liệu



MS COCO

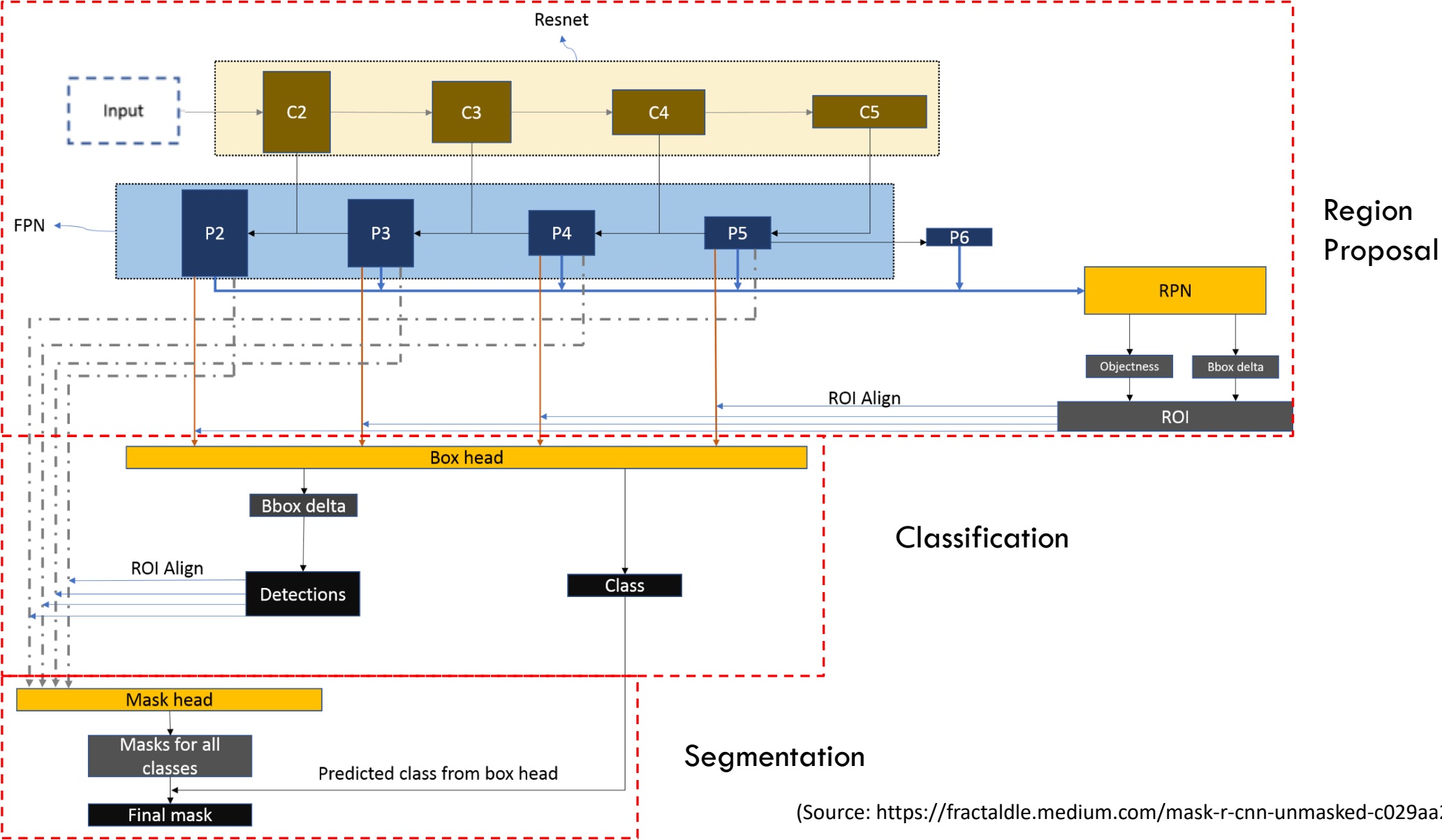
- > 200K ảnh được gán nhãn
- 80 lớp đối tượng
- 1 triệu rưỡi cá thể



LABELME

Thủ công

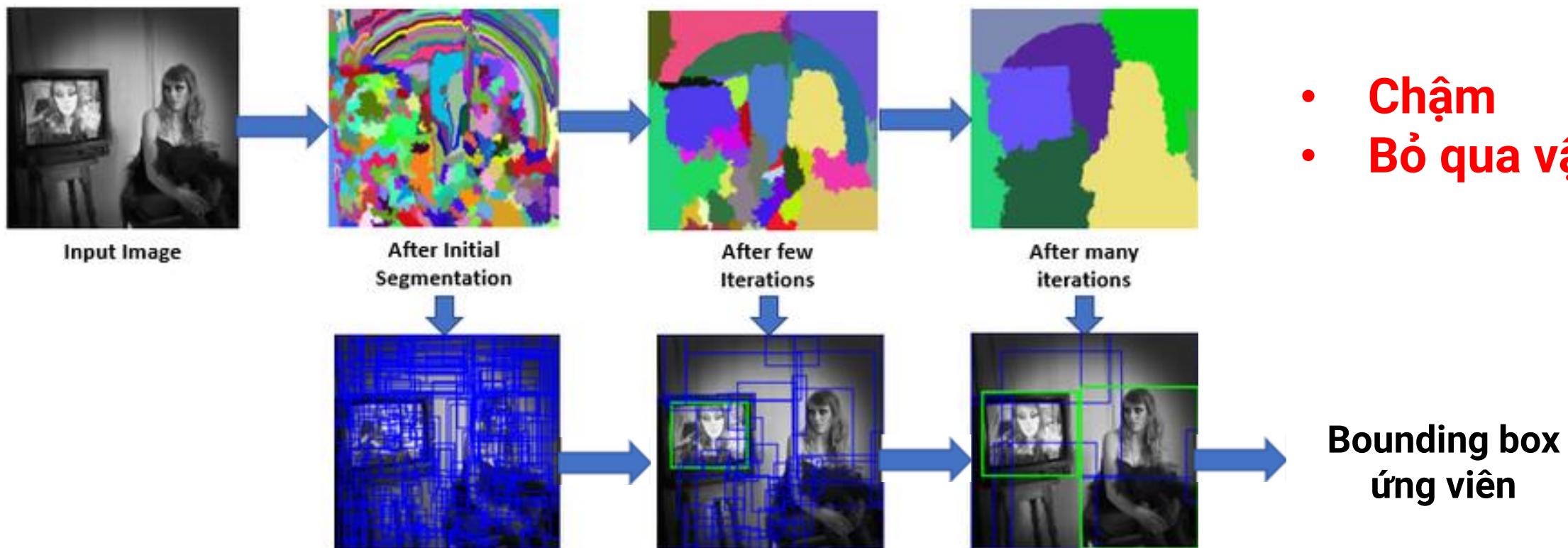
Mask R-CNN



(Source: <https://fractalidle.medium.com/mask-r-cnn-unmasked-c029aa2f1296>)

Mask R-CNN: Phát sinh ứng viên

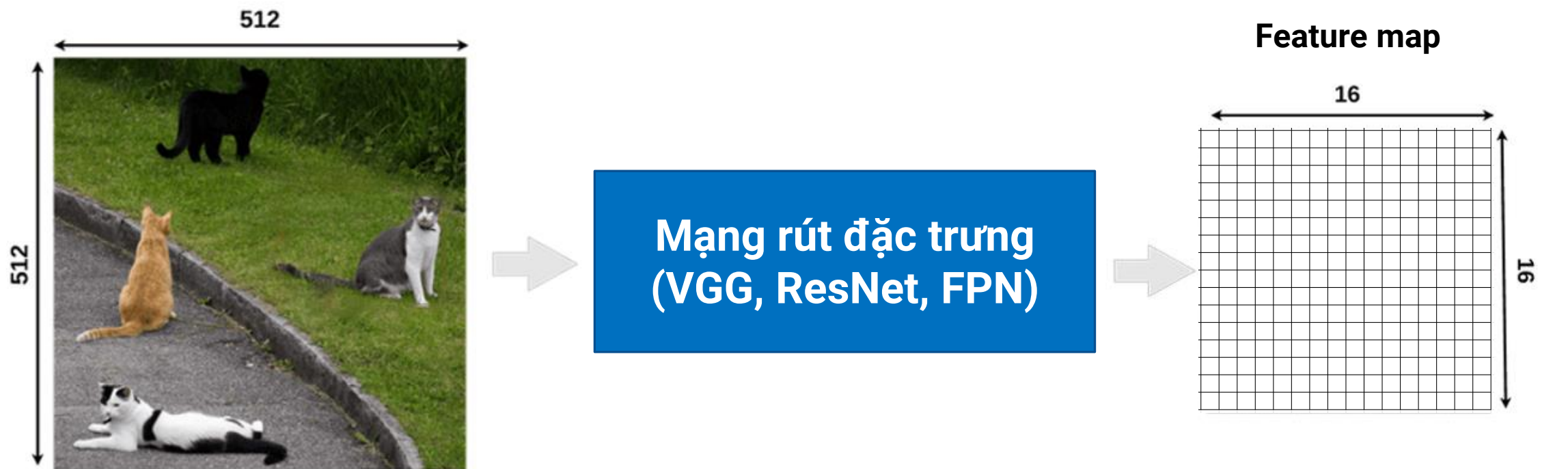
Selective search



(Source: <https://www.geeksforgeeks.org/selective-search-for-object-detection-r-cnn>)

Mask R-CNN: Phát sinh ứng viên

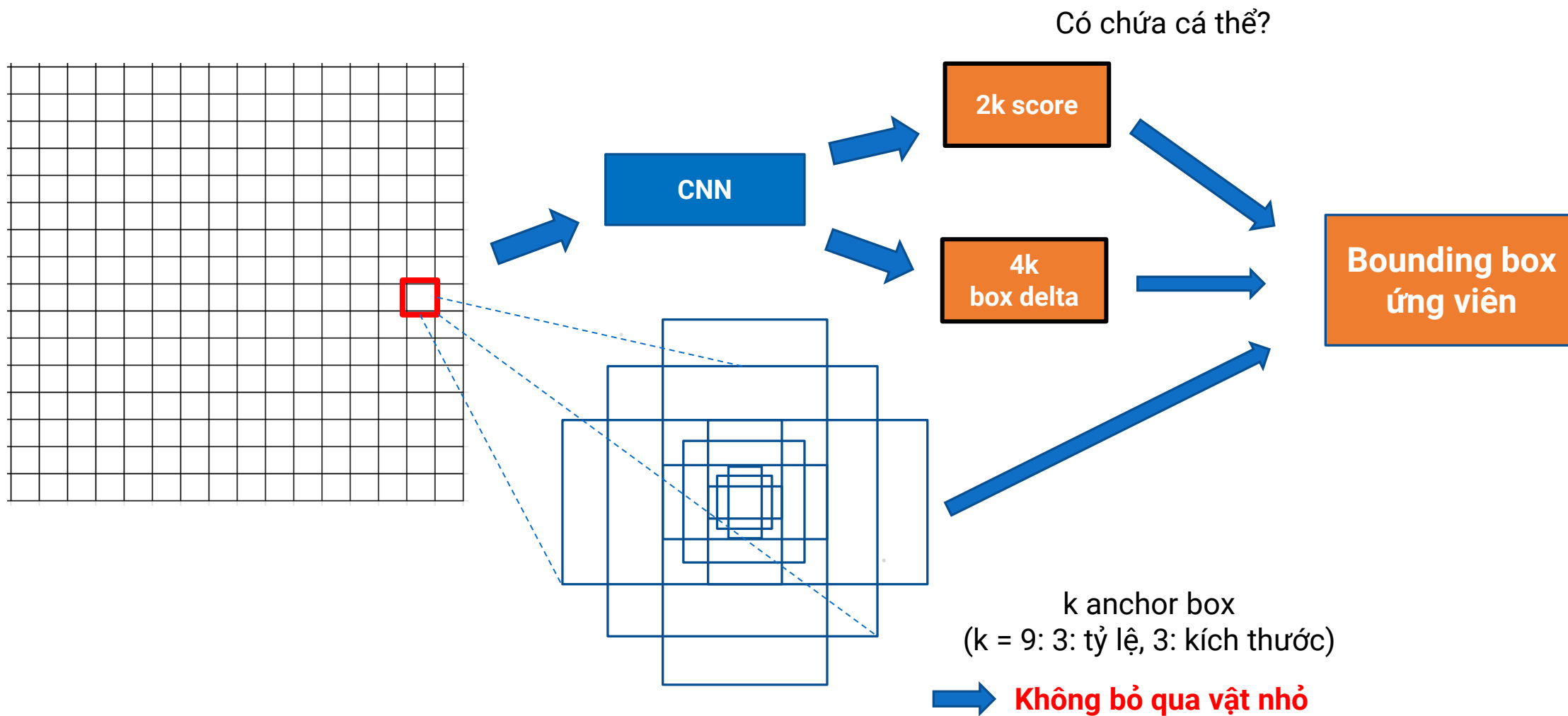
Mạng học sâu



Giảm **kích thước** -> **Nhanh**
Nhưng vẫn **đảm bảo thông tin**

Mask R-CNN: Phát sinh ứng viên

Mạng học sâu



Mask R-CNN: Phát sinh ứng viên

Mạng học sâu

Score loss function:

$$loss_{rpn_cls} = loss_{pos} + loss_{neg}$$

$$= -\frac{1}{k} \sum_{i=1}^k gt_scores[i] \cdot \log(scores[2i-1]) + (1 - gt_scores[i]) \cdot \log(scores[2i])$$

- k: số anchor box.
- gt_scores: nếu anchor box thứ i có IoU với bất kỳ groundtruth bbox > 0.7 thì gt_scores[i] = 1, ngược lại gt_scores[i] = 0.
- scores: kết quả dự đoán (thuộc [0, 1]). (anchor thứ i có xác suất scores[2i-1] chứa cá thể và scores[2i] không chứa cá thể -> chọn kết quả lớn hơn)

Mask R-CNN: Phát sinh ứng viên

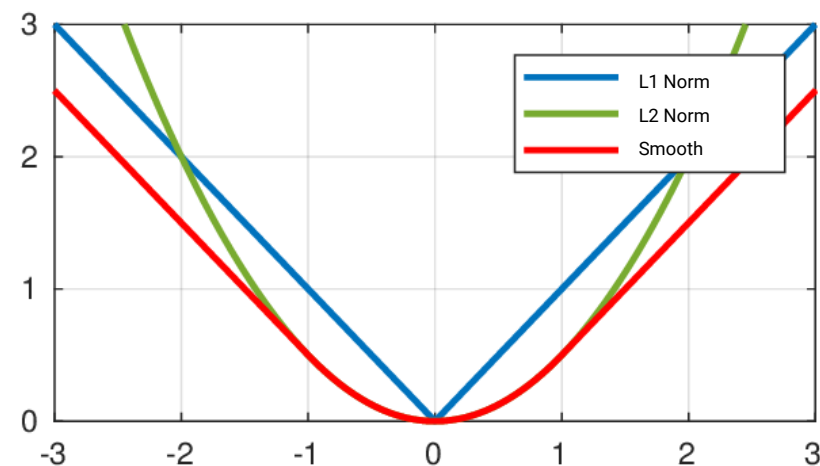
Mạng học sâu

Anchor box delta loss function:

$$loss_{rpn_delta} = \frac{1}{k} \sum_{i=1}^k \sum_{j \in cx, cy, w, h} smooth(gt_delta[i][j], delta[i][j])$$

$$smooth(x, y) = \begin{cases} 0.5(x - y)^2, & \text{nếu } |x - y| < 1 \\ |x - y| - 0.5, & \text{nếu ngược lại} \end{cases}$$

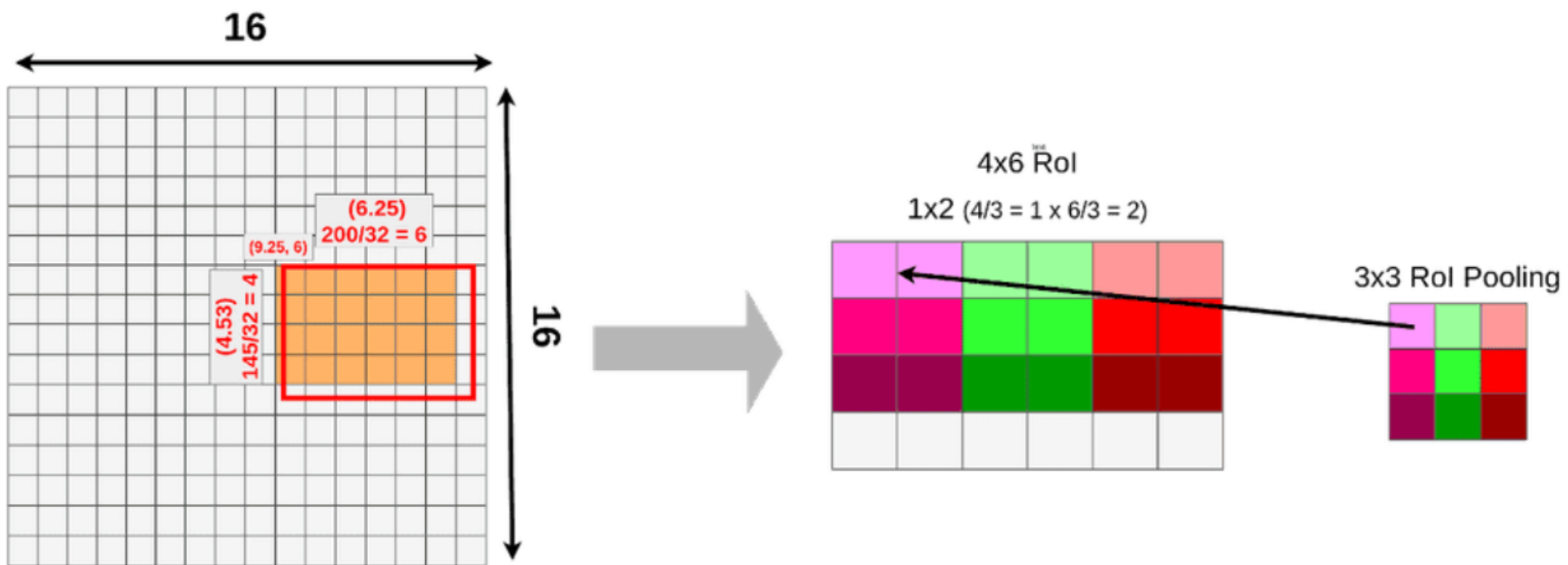
- k: số anchor box.
- gt_delta: groundtruth bounding box – anchor box.
- delta: kết quả hiệu chỉnh anchor box.



Mask R-CNN: Phân lớp

- Cố định kích thước các ứng viên

ROI Pooling

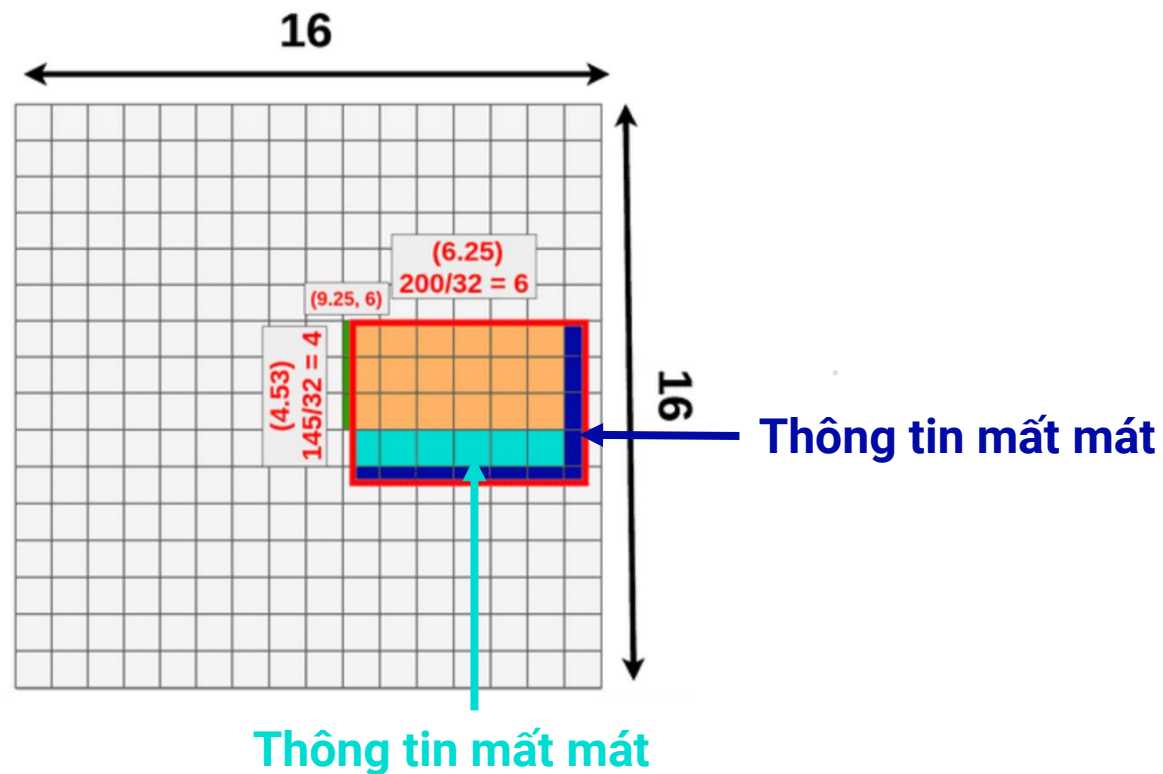


(Source: <https://towardsdatascience.com/understanding-region-of-interest-part-2-roi-align-and-roi-warp-f795196fc193>)

Mask R-CNN: Phân lớp

- Cố định kích thước các ứng viên

Vấn đề của ROI Pooling

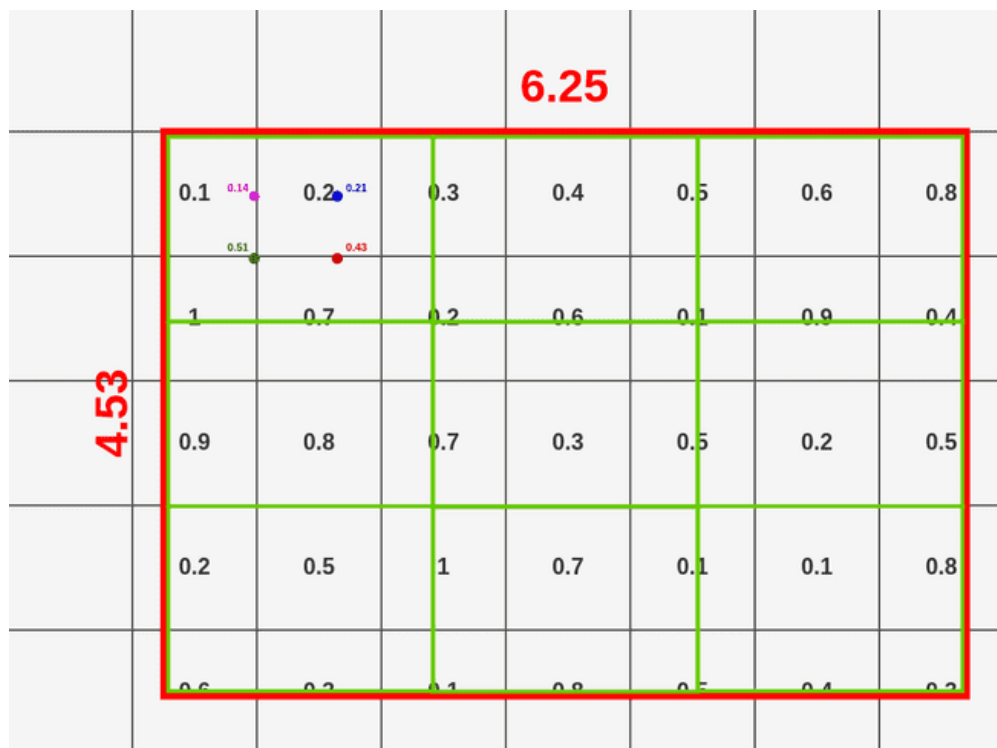


Quá trình làm tròn/
Lượng tử hóa
(Quantization)
gây **mất mát** thông tin

Mask R-CNN: Phân lớp

- Cố định kích thước các ứng viên

ROI Align



$$1 \times 1 = \text{MAX}(0.14, 0.21, 0.51, 0.43) = 0.51$$

3x3 RoIAlign

0.51		

Chọn điểm mốc

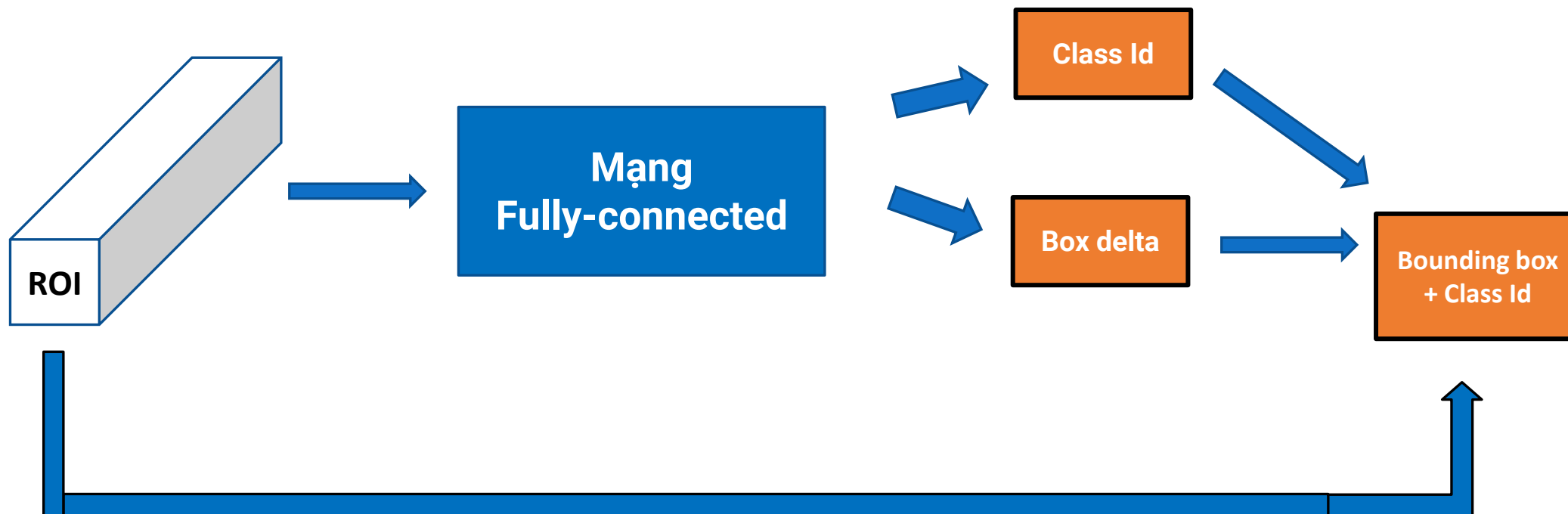


Nội suy song
tuyến tính



Lấy average
hoặc max

Mask R-CNN: Phân lớp



Mask R-CNN: Phân lớp

Loss function

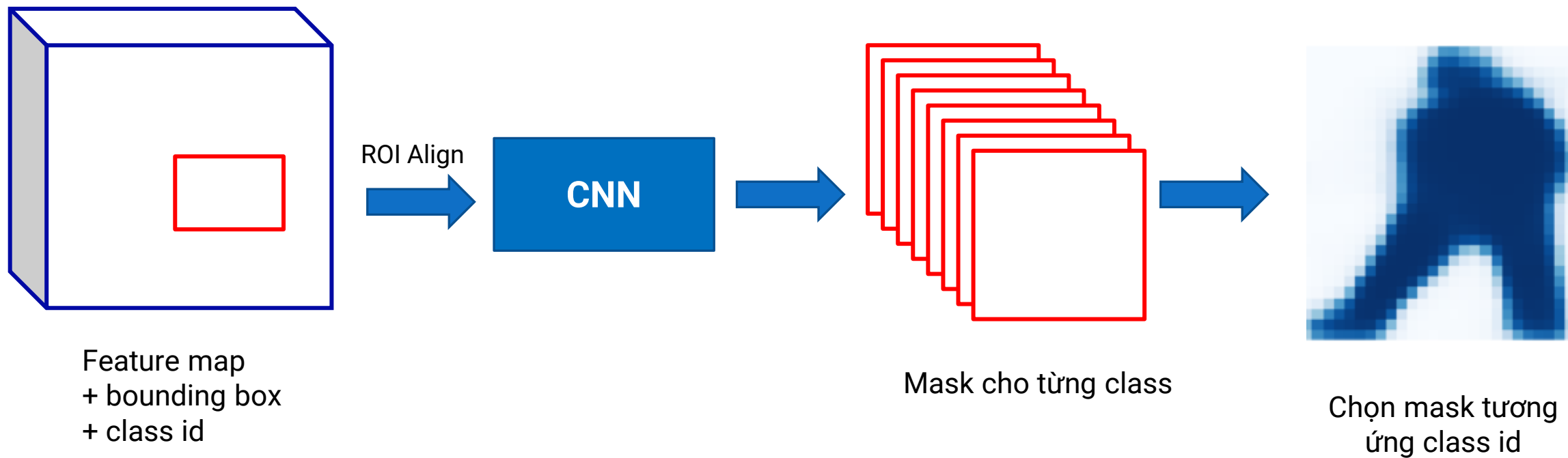
Classification loss:

$$loss_{cls} = -\frac{1}{num_class} \sum_{i=1}^{num_class} gt_class[i] \cdot \log(pred_class[i])$$

Bounding box loss: $loss_{delta}$ giống với $loss_{rpn_delta}$

- num_class : số lượng class được định nghĩa trước.
- gt_class : $gt_class[i] = 1$ nếu groundtruth chứa cá thể thuộc lớp i , $gt_class[j] = 0$ với mọi j khác i .
- $pred_class$: kết quả dự đoán (thuộc $[0,1]$) từng class cho bounding box đang xét.

Mask R-CNN: Phân đoạn



Mask R-CNN: Phân đoạn

Mask loss function

$$\begin{aligned} loss_{mask} = & -\frac{1}{mask_size} \sum_{i=1}^{mask_size} gt_mask[i] \cdot \log(pred_mask[class_id][i]) \\ & + (1 - gt_mask[i]) \cdot \log(1 - pred_mask[class_id][i]) \end{aligned}$$

- mask_size: tổng số pixel của một mask.
- gt_mask: gt_mask[i] = 1 nếu pixel thứ i thuộc groundtruth mask, ngược lại gt_mask[i] = 0.
- pred_class: kết quả dự đoán (thuộc [0,1]) từng pixel cho từng class.

V. Ứng dụng

V. Ứng dụng



(Source: <https://keymakr.com/blog/instance-segmentation-for-waste-management-ais/>)

V. Ứng dụng

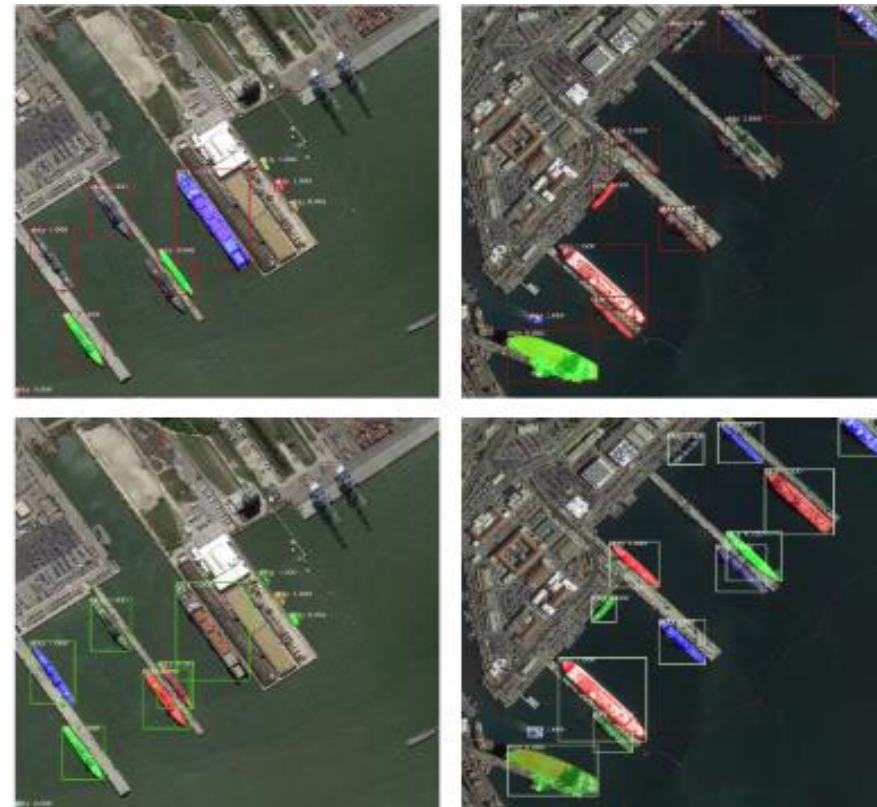


(Jia et al. 2020)

V. Ứng dụng



(Mohanty et al. 2020)



(Nie et al. 2018)

VI. Tài liệu tham khảo

- [1] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.
- [2] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149, 2016.
- [3] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.
- [4] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.
- [5] Shanlan Nie, Zhiguo Jiang, Haopeng Zhang, Bowen Cai, and Yuan Yao. Inshore ship detection based on mask r-cnn. In *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, pages 693–696, 2018.
- [6] Weikuan Jia, Yuyu Tian, Rong Luo, Zhonghua Zhang, Jian Lian, and Yuanjie Zheng. Detection and segmentation of overlapped fruits based on optimized mask r-cnn application in apple harvesting robot. *Computers and Electronics in Agriculture*, 172:105380, 2020.
- [7] Sharada Prasanna Mohanty, Jakub Czakon, Kamil A. Kaczmarek, Andrzej Pyskir, Piotr Tarasiewicz, Saket Kunwar, Janick Rohrbach, Dave Luo, Manjunath Prasad, Sascha Fler, Jan Philip Göpfert, Akshat Tandon, Guillaume Mollard, Nikhil Rayaprolu, Marcel Salathe, and Malte Schilling. Deep learning for understanding satellite imagery: An experimental survey. *Frontiers in Artificial Intelligence*, 3:85, 2020.



**Cảm ơn thầy và các bạn
đã lắng nghe**