Semantic Segmentation

Tran Minh Thanh – 19127550@student.hcmus.edu.vn
Ngo Nhat Du – 19127120@student.hcmus.edu.vn

Abstract **- Trong bài viết này, giải quyết loại phân đoạn hình ảnh cụ thể hơn, Semantic Segmentation (Phân đoạn ngữ nghĩa). Phân đoạn hình ảnh là một nhiệm vụ thị giác máy tính được hỗ trợ bởi một số lượng lớn các nghiên cứu liên quan đến các thuật toán dựa trên xử lý hình ảnh và các kỹ thuật dựa trên học tập. Trong thị giác máy tính, hầu hết các mô hình phân đoạn hình ảnh bao gồm một mạng encoder-decoder so với một mạng encoder duy nhất trong phân loại. Semantic Segmentation đã trở thành ứng dụng quan trọng quan trọng trong xử lý hình ảnh, video và lĩnh vực thị giác máy tính, việc sử dụng phương pháp xử lý ảnh này được áp dụng trong nhiều ngành công nghiệp từ y tế đến vận chuyển và hơn thế nữa. Semantic Segmentation mô tả suy luận chi tiết bằng cách dự đoán nhãn cho từng pixel riêng lẻ trong hình ảnh hoặc video đầu vào. Mỗi pixel được dán nhãn tương ứng với mỗi lớp đối tượng trong đó có chứa chính pixel đó. Mục đích của công việc này là phân đoạn hình ảnh và video mà không làm mất chi tiết tốt của mặt nạ phân đoạn đầu ra hoặc quá tải sử dụng nhớ GPU bằng cách sử dụng kiến trúc Semantic Segmentation được đề xuất, DeepLab. Ở đây, cũng sẽ thảo luận về ứng dụng và sự khác biệt giữa Semantic Segmentation, Instance Segmentation và Panoptic Segmentation để so sánh dễ dàng hơn. Khi đề cập đến sự khác biệt của từng loại phân đoạn hình ảnh, những ưu điểm và nhược điểm cũng sẽ được đề cập. Hơn nữa, công việc này sẽ thừa nhận thêm một số mạng sâu tiêu chuẩn, vì Semantic Segmentation liên quan đến vô số nghiên cứu liên quan đến Deep Neural Network (Mạng học sâu).**

Index Terms **- Computer Vision; Convolutional Neural Networks; Deep Learning; Fully Convolutional Network; Image Segmentation; Instance Segmentation; Object Detection; Panoptic Segmentation; Semantic Segmentation;**

I. GIỚI THIỆU

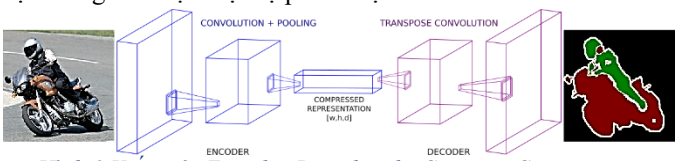
Các kỹ thuật Semantic Segmentation hiện đại chủ yếu dựa trên học sâu [1 - 7], một Convolutional Neural Networks (CNN) lấy hình ảnh hoặc video đầu vào và xuất ra một bản đồ phân đoạn. Semantic Segmentation, còn được gọi là phân loại cấp pixel, là một nhiệm vụ thị giác máy tính liên quan đến việc nhóm lại với nhau các phần tương tự của hình ảnh hoặc video thuộc cùng một lớp. Ý tưởng của Semantic Segmentation là nhận ra và hiểu những gì có trong hình ảnh hoặc video ở cấp độ pixel. Điều đó có nghĩa là, mục đích chính là dự đoán một nhãn cho các đơn vị đầu vào, trong trường hợp này là pixel của hình ảnh hoặc video có lớp tương ứng, nhiệm vụ này được gọi là dự đoán dày đặc. Để giải thích đơn giản hơn, mục tiêu là lấy đầu vào dưới dạng hình ảnh hoặc video và tạo đầu ra dưới dạng bản đồ phân đoạn, trong đó giá trị pixel (từ 0 đến 255) của đầu vào được chuyển thành giá trị được dán nhãn lớp (từ 0 đến n) và được trực quan hóa như sau:

Semantic Segmentation theo ba bước:

Phân loại hình ảnh: Phân loại một đối tượng nhất định trong hình ảnh hoặc video.

Nhận dạng và Phát hiện đối tượng: Tìm kiếm các đối tượng và vẽ một Hộp ràng buộc xung quanh chúng.

Semantic Segmentation: Nhóm các pixel trong một hình ảnh cục bằng cách tạo mặt nạ phân đoạn.



Hình 1 Kiến trúc Encoder-Decoder cho Semantic Segmentaion

A. Background

Phương pháp ngưỡng: phân đoạn các pixel tương tự bằng cách sử dụng ngưỡng để tìm các giá trị đỉnh của biểu đồ tần suất của hình ảnh.

Phân đoạn dựa trên cạnh: phát hiện sự gián đoạn ranh giới của một đối tượng trong một hình ảnh để xác định hình dạng của đối tượng, hỗ trợ phân đoạn các đối tượng khác nhau trong hình ảnh.

Phân khúc dựa trên khu vực: theo một tiêu chí nhất định, phân đoạn này phân chia hình ảnh thành các khu vực giống nhau. Phương pháp này liên quan đến một thuật toán tạo ra các phân đoạn bằng cách chia một hình ảnh thành các thành phần có đặc điểm pixel phù hợp.

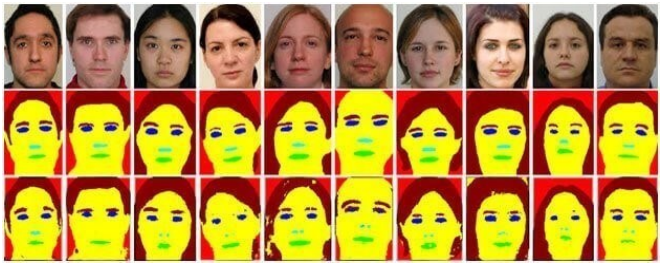
Phân đoạn hình ảnh dựa trên cụm: phân đoạn hình ảnh theo một tập hợp các điểm dữ liệu được nhóm lại.

Phân đoạn hình ảnh dựa trên học sâu: Trong các mạng thần kinh phức tạp phân đoạn hình ảnh được sử dụng để phân đoạn từng phiên bản đối tượng trong một hình ảnh mà **MASK-RCNN** [20] là một thuật toán phổ biến.

B. Ứng dụng

Nhận dạng khuôn mặt

Bằng cách thực hiện phương pháp này trong các hệ thống thị giác máy tính, các nhiệm vụ như nhận dạng tuổi, dự đoán giới tính, dân tộc và biểu hiện trên nhận dạng, v.v. có thể dễ dàng nhận dạng được. Chúng được thực hiện bằng cách tách các vùng mặt như miệng, mắt, mũi, cằm, tóc, v.v. Phân đoạn khuôn mặt rất hữu ích trong nhiều ứng dụng khuôn mặt của thị giác máy tính, chẳng hạn như ước tính giới tính, biểu hiện, tuổi tác và dân tộc. Các yếu tố đáng chú ý ảnh hưởng đến dataset phân đoạn khuôn mặt và phát triển mô hình là các biến thể trong điều kiện ánh sáng, biểu cảm khuôn mặt, định hướng khuôn mặt, tắc nghẽn và độ phân giải hình ảnh.

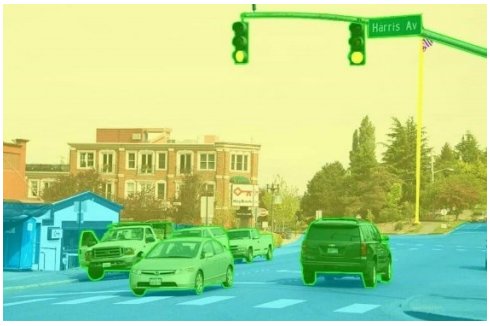


Phân loại thời trang

Phân tích quần áo là một nhiệm vụ rất phức tạp so với những người khác do số lượng lớn các lớp học. Điều này phân biệt chính nó với các vấn đề phân đoạn đối tượng hoặc cảnh nói chung vì phân loại quần áo hạt mịn đòi hỏi sự phán xét cấp cao hơn dựa trên ngữ nghĩa của quần áo, sự thay đổi của tư thế con người và số lượng lớp học có khả năng đáng kể. Phân tích quần áo đã được nghiên cứu tích cực trong cộng đồng thị giác vì giá trị to lớn trong các ứng dụng trong thế giới thực, tức là thương mại điện tử. Một số dataset như dataset Fashionista và CFPD cung cấp quyền truy cập mở vào Semantic Segmentation cho các mặt hàng quần áo.

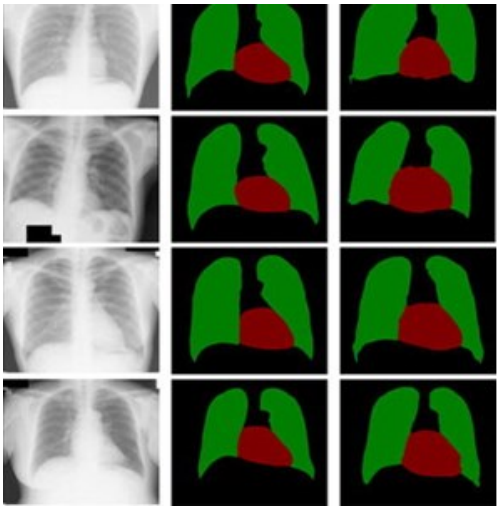
Xe tự lái

Tự lái xe là một nhiệm vụ cực kỳ phức tạp đòi hỏi phải phân tích, nhận thức và hành động theo thời gian thực. Semantic Segmentation giúp các phương tiện tự trị xác định các đối tượng như tín hiệu giao thông, làn đường, người đi bộ, v.v. bất kỳ vật cản nào gặp nhau trên đường đi. Trang bị cho xe hơi nhận thức cần thiết để hiểu môi trường của chúng để những chiếc xe tự lái có thể tích hợp một cách an toàn vào những con đường hiện có.



Hình ảnh và chẩn đoán y tế

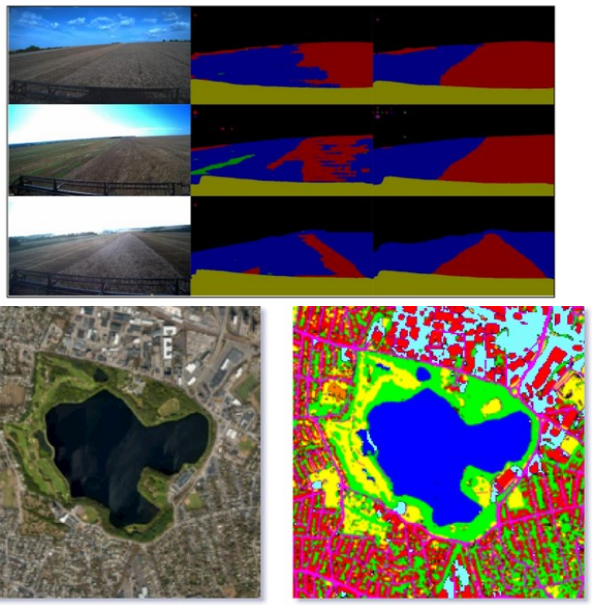
Các chuyên gia chẩn đoán thất bại trong việc phân tích chính xác các biểu đồ y tế do sự chồng chéo hoặc phức tạp của hình ảnh y tế. Phương pháp này thực hiện phân loại, làm cho các xét nghiệm chẩn đoán đơn giản hơn và tạo ra kết quả nhanh hơn.



Chế độ xem hình ảnh trên không / Cảm biến địa lý

Thông tin che phủ đất rất quan trọng đối với các ứng dụng khác nhau, chẳng hạn như giám sát các khu vực phá rừng và đô thị hóa. Để nhận ra loại bìa đất (ví dụ: các khu vực đô thị, nông nghiệp, nước, v.v.) cho mỗi pixel trên hình ảnh vệ tinh, phân loại độ che phủ đất có thể được coi là một nhiệm vụ Semantic Segmentation đa cấp. Phát hiện đường và tòa nhà cũng là một chủ đề nghiên cứu quan trọng để quản lý giao thông, quy hoạch thành phố và giám sát đường bộ. Hình ảnh vệ tinh được chú thích chính xác ở đây để theo dõi các lĩnh vực như vậy và thu thập thông tin hữu ích để sửa chữa hiệu quả canh tác và nâng cao năng suất cây trồng. Robot canh tác chính xác có thể làm giảm số lượng thuốc diệt cỏ cần được

phun ra trên các cánh đồng và Semantic Segmentation của cây trồng và cỏ dại hỗ trợ chúng trong thời gian thực để kích hoạt các hành động làm cỏ. Các kỹ thuật tầm nhìn hình ảnh tiên tiến như vậy cho nông nghiệp có thể làm giảm giám sát thủ công của nông nghiệp.



II. CÁC CÔNG TRÌNH LIÊN QUAN

A. Phương pháp phân đoạn hình ảnh

Instance Segmentation *(Phân đoạn Cá thể)*

Instance Segmentation [8] hiện nay là các lĩnh vực quan trọng, phức tạp và đầy thách thức đáng kể trong nghiên cứu thị giác máy tính hoặc máy tính. Mục tiêu là dự đoán nhãn lớp đối tượng và mặt nạ phiên bản của các đối tượng cụ thể pixel, có nghĩa là bản địa hóa các lớp phiên bản đối tượng khác nhau được trình bày trong vô số hình ảnh. Có hai phương pháp cho loại phân khúc này, đó là phân khúc dựa trên khu vực và Semantic Segmentation dựa trên.

Instance Segmentation phát hiện các đối tượng riêng lẻ trong một thể loại được xác định, có nghĩa là xử lý nhiều đối tượng cùng lớp như các phiên bản riêng biệt với các nhãn đặc biệt. Cách tiếp cận này được sử dụng để nghiên cứu các đối tượng được xác định rõ ràng, dễ phân loại và phát hiện hơn. Cụ thể hơn, tạo ra các mặt nạ phân đoạn riêng lẻ cho từng và mọi đối tượng trong hình ảnh.

Được thúc đẩy bởi hiệu quả của R-CNN, nhiều cách tiếp cận phân khúc phiên bản dựa trên các đề xuất phân đoạn. Các phương pháp trước đó [9 - 11] sử dụng các phân đoạn từ dưới lên [12, 13]. Trong khi DeepMask [14] và các tác phẩm sau [15, 16] học cách đề xuất các ứng cử viên phân đoạn, sau đó được phân loại bởi Fast R-CNN [17].

Panoptic Segmentation *(Phân đoạn Khái quát)*

Panoptic Segmentation [18] kết hợp cả Semantic Segmentation và Instance Segmentation, xác định các đối tượng liên quan đến nhãn lớp và cũng xác định tất cả các trường hợp trong hình ảnh nhất định. Đầu vào của phương pháp này được đưa vào hai mạng:

- **Mạng hoàn toàn phức tạp (FCN)** [19]**:** chịu trách nhiệm chụp các mẫu từ các đối tượng không thể đếm được và mang lại các Semantic Segmentation. FCN sử dụng các kết nối bỏ qua, cho phép xây dựng lại ranh giới phân đoạn chính xác và đưa ra dự đoán địa phương xác định chính xác cấu trúc toàn cầu hoặc tổng thể của đối tượng.
- **Mặt nạ R-CNN** [14]**:** chịu trách nhiệm chụp các mẫu của các đối tượng có thể đếm được và mang lại các Instance Segmentation. Mask R-CNN bao gồm Mạng đề xuất khu vực (RPN) là một quá trình mà mạng mang lại các khu vực quan tâm (ROI) và R-CNN nhanh hơn tận dụng ROI để thực hiện phân loại và tạo ra các hộp giới hạn.

Tuy nhiên, cách tiếp cận này có nhiều nhược điểm, chẳng hạn như tính toán không hiệu quả, không nhất quán giữa các đầu ra mạng và không có khả năng học các mẫu hữu ích. Để giải quyết những vấn đề này, Phân khúc panoptic hiệu quả (EfficientPS [20]), một kiến trúc mới đã được đề xuất để cải thiện hiệu quả và hiệu suất.

B. Mạng sâu phổ biến [25]:

Phân loại ImageNet với Deep Convolutional Neural Networks (DCNNs)

Convolutional Neural Networks (CNNs) có thể được kiểm soát bằng cách thay đổi độ sâu và chiều rộng của chúng, và cũng đưa ra các giả định mạnh mẽ và chủ yếu là chính xác về bản chất của hình ảnh (cụ thể là tính đứng yên của số liệu thống kê và địa phương phụ thuộc pixel), đó là lý do tại sao các CNN có ít kết nối và thông số hơn nhiều, vì vậy chúng được đào tạo dễ dàng. GPU hiện tại, kết hợp với việc thực hiện tối ưu hóa cao của sự phức tạp 2D, đủ mạnh để tạo điều kiện cho việc đào tạo các CNN lớn thú vị và các dataset gần đây như ImageNet chứa đủ các ví dụ được dán nhãn để đào tạo các mô hình như vậy mà không bị quá tải nghiêm trọng. DNN là nền tảng của các mạng Semantic Segmentation, có đóng góp đáng kể cho nghiên cứu về thị giác máy tính.

AlexNet [21]: CNN sâu tiên phong của Toronto đã giành chiến thắng trong cuộc thi ImageNet năm 2012 với độ chính xác thử nghiệm là 84,6%. AlexNet bao gồm 5 lớp phức tạp, các lớp tối đa, ReLUs là phi tuyến tính, 3 lớp hoàn toàn phức tạp và dropout.

VGG-16[22]: Mô hình của Oxford này đã giành chiến thắng trong cuộc thi ImageNet 2013 với độ chính xác 92,7%. Mô hình này sử dụng một chồng các lớp phức tạp với các trường tiếp nhận nhỏ trong các lớp đầu tiên thay vì vài lớp với các trường tiếp nhận lớn. VGGNet có mặt khắp nơi trong cộng đồng nghiên cứu để trích xuất các tính năng từ hình ảnh, sử dụng một chồng các lớp phức tạp với các trường tiếp nhận nhỏ.

GoogLeNet [23]: Mạng của Google này đã giành chiến thắng trong cuộc thi ImageNet năm 2014 với độ chính xác

93,3%. GoogleNet được tạo thành bởi 22 lớp và một building block mới được giới thiệu được gọi là mô-đun eption inc. Mô-đun bao gồm một lớp Network-in-Network, một hoạt động gộp lại, một lớp phức tạp có kích thước lớn và lớp phức tạp có kích thước nhỏ.

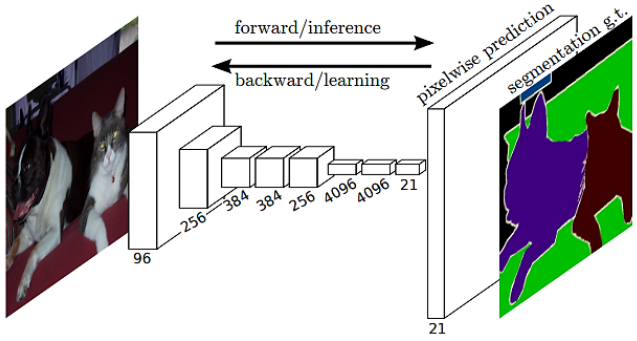
ResNet [24]: Mô hình này của Microsoft đã giành chiến thắng trong cuộc thi ImageNet 2016 với độ chính xác 96,4%. ResNet nổi tiếng với độ sâu (152 lớp) và sự ra đời của các khối còn lại. Các khối còn lại giải quyết vấn đề đào tạo một kiến trúc thực sự sâu sắc bằng cách giới thiệu các kết nối bỏ qua danh tính để các lớp có thể sao chép đầu vào của chúng sang lớp tiếp theo.

Model	Time	Accuracy	Number of Parameters	Number of Layers
AlexNet	2012	51.2%	60M	8
VGGNet	2014	71.5%	138M	16
GoogleNet	2014	69.8%	6.8M	22
ResNet	2015	78.6%	55M	152

Hình 2 Tóm tắt các mô hình CNN khác nhau về nhiệm vụ phân loại ImageNet.

C. Kiến trúc Semantic Segmentation:

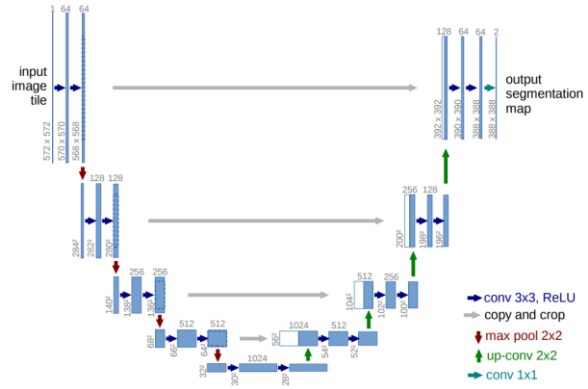
Fully Convolutional Network (FCN / VGG16) [5]



Hình 3 Các mạng hoàn toàn phức tạp có thể học cách đưa ra dự đoán dày đặc cho các tác vụ trên mỗi pixel như Semantic Segmentation.

Full Convolutional Networks (FCNs) được đào tạo từ đầu đến cuối để phân đoạn hình ảnh, lấy một hình ảnh với kích thước tùy ý và tạo ra một hình ảnh phân đoạn với cùng kích thước. Cụ thể, FCN chỉ bao gồm các lớp phức tạp.

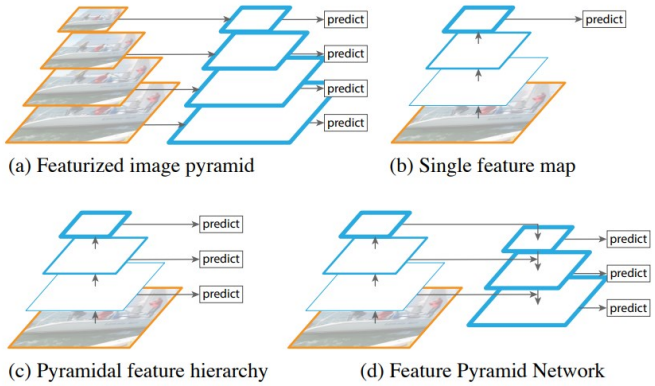
U-Net – Mô hình dựa trên encoder / decoder [7]



Hình 4 Kiến trúc U-net (ví dụ cho 32x32 pixel ở độ phân giải thấp nhất). Mỗi hộp màu xanh tương ứng với một bản đồ tính năng đa kênh. Số lượng kênh được ký hiệu trên đầu hộp. Kích thước x-y được cung cấp ở cạnh dưới bên trái của hộp. Hộp trắng đại diện cho bản đồ tính năng được sao chép. Các mũi tên biểu thị các thao tác khác nhau.

Kiến trúc U-Net được sử dụng để mở rộng kiến trúc FCN. Phần contracting tính toán các tính năng và phần mở rộng là cho các mẫu cục không gian trong hình ảnh. Downsampling có kiến trúc giống như FCN được sử dụng để trích xuất các tính năng. Việc nâng cấp sử dụng giải phóng (deconvolution) giảm số lượng bản đồ tính năng, vẫn làm tăng chiều cao và chiều rộng của chúng. Các kiến trúc U-Net bao gồm ứng dụng lặp đi lặp lại của hai phức tạp 3x3 mà mỗi tiếp theo là ReLU và hoạt động gộp tối đa 2x2 với hai cho downsampling. Với mỗi giai đoạn downsampling, tăng gấp đôi số lượng kênh đặc trưng. Tại một con đường mở rộng, bao gồm một upsampling tiếp theo là một tích chập 2x2, làm giảm một nửa số lượng các kênh tính năng, một liên kết với bản đồ tính năng cắt xén tương ứng, và hai phức tạp 3x3, với mỗi tiếp theo là ReLU. U-net là đối xứng và bỏ qua các kết nối bằng cách áp dụng một toán tử ghép nối. Lợi ích lớn của U-net là chạy nhanh hơn nhiều so với FCN hoặc Mask RCNN.

Mạng kim tự tháp Tính năng (FPN) – Mô hình đa quy mô [43]



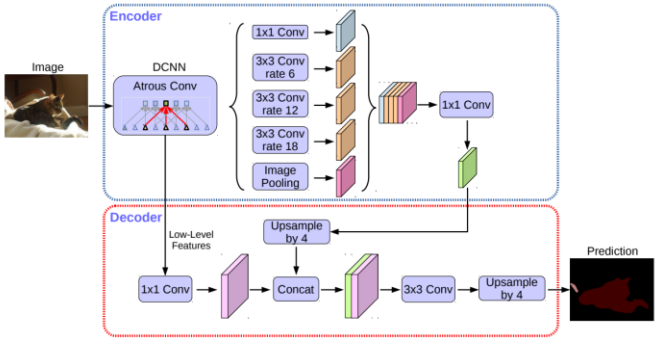
Hình 5 (a) Sử dụng một kim tự tháp hình ảnh để xây dựng một kim tự tháp tính năng. Các tính năng được tính toán trên mỗi thang đo hình ảnh một cách độc lập, chậm. (b) Các hệ thống phát hiện gần đây đã chọn chỉ sử dụng các tính năng quy mô duy nhất để phát hiện nhanh hơn. (c) Một giải pháp thay thế là sử dụng lại hệ thống phân cấp tính năng kim tự tháp được tính toán bởi ConvNet như thể là một kim tự tháp hình ảnh đặc trưng. (d) Mạng kim tự tháp tính năng được đề xuất của (FPN) nhanh như (b) và (c), nhưng chính xác hơn. Trong hình này, bản đồ tính năng được chỉ định bằng các đường viền màu xanh và các đường viền dày hơn biểu thị các tính năng mạnh hơn về mặt ngữ nghĩa.

Mô hình đa quy mô đã được triển khai trong các kiến trúc mạng thần kinh khác nhau, với mô hình nổi bật nhất là Pyramid Network (FPN). Kiến trúc FPN bao gồm một con đường từ dưới lên, một con đường từ trên xuống và các kết nối bên để tham gia vào các tính năng có độ phân giải thấp. Con đường từ dưới lên có một hình ảnh với kích thước tùy ý, xử lý các lớp phức tạp và được ghép bằng các lớp gộp lại. Con đường từ trên xuống bao gồm một quá trình nâng cấp. Đối với

phân đoạn hình ảnh, các tác giả sử dụng hai perceptrons nhiều lớp (MLP) để tạo ra các mặt nạ.

D. Encoder – Decoder

Mạng encoder – decoder đã được áp dụng thành công cho nhiều nhiệm vụ thị giác máy tính, bao gồm ước tính tư thế con người [46], phát hiện đối tượng [43][47][48] và phân đoạn ngữ nghĩa. Thông thường, mạng encoder – decoder chứa (1) mô-đun encoder thu nhỏ dần bản đồ đối tượng và thu thập thông tin ngữ nghĩa cao hơn, và (2) mô-đun decoder khôi phục dần thông tin không gian. Sử dụng DeepLabv3 làm mô-đun encoder và thêm một mô-đun decoder đơn giản nhưng hiệu quả để thu được các phân đoạn sắc nét hơn.



Hình 6 DeepLabv3 + mở rộng DeepLabv3 bằng cách sử dụng cấu trúc encoder – decoder.

DeepLabv3 + mở rộng DeepLabv3 bằng cách sử dụng cấu trúc encoder – decoder. Mô-đun của encoder sẽ encoder thông tin ngữ cảnh đa tỷ lệ bằng cách áp dụng tích chập bất thường ở nhiều tỷ lệ, trong khi mô-đun decoder đơn giản nhưng hiệu quả sẽ tinh chỉnh kết quả phân đoạn dọc theo ranh giới đối tượng.

III. FRAMEWORK

A. Kiến trúc

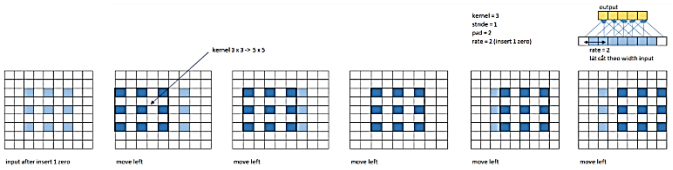
Atrous Convolution

Deep Convolutional Neural Networks (DCNNs) [35] đã đẩy nhanh hiệu suất của các hệ thống thị giác máy tính để một loạt các vấn đề cấp cao, bao gồm phân loại hình ảnh [21 - 23][36] và phát hiện đối tượng [9][17][38], nơi đào tạo đầu cuối từ DCNNs có sự thay đổi tốt hơn đáng kể so với các tính năng thủ công dựa vào các hệ thống. Tuy nhiên, vẫn còn nhiều thách thức trong Semantic Segmentation bằng cách sử dụng DCNNs.

Vấn đề đầu tiên là độ phân giải tính năng bị giảm trong các lớp DCNN liên tiếp ban đầu được thiết kế cho phân loại hình ảnh liên tục kết hợp max-pooling và downsampling, dẫn đến độ phân giải không gian giảm đáng kể. Để cải thiện những nhược điểm này và tạo ra hiệu quả các bản đồ tính năng dày đặc hơn, downsampling phải được loại bỏ khỏi vài lớp DCNN tối đa cuối cùng và được thay thế bằng toán tử cải tiến của các lọc trong các lớp phức tạp phụ, dẫn đến bản đồ tính năng tỷ lệ lấy mẫu cao hơn. Kỹ thuật này được biết đến như là sự phức tạp atrous hoặc còn được gọi là Dilated Convolution, giới thiệu tốc độ giãn nở cho các lớp phức tạp. Sự phức tạp giãn nở $x(i)$ của một tín hiệu được định nghĩa là:

$$y_i = \sum_{k=1}^{K} x[i + rk]w[k],$$

cho mỗi tế bào trên đầu ra, sự phức tạp Atrous được tính bằng cách nhân sự phức tạp lọc với bản đồ tính năng. Ở đây, tỷ lệ atrous r tương ứng với khoảng cách giữa các hàng và cột với các giá trị 0, nếu tỷ lệ atrous bằng r, thì khoảng cách của các hàng và cột liên tiếp là $r - 1$ hàng và cột.

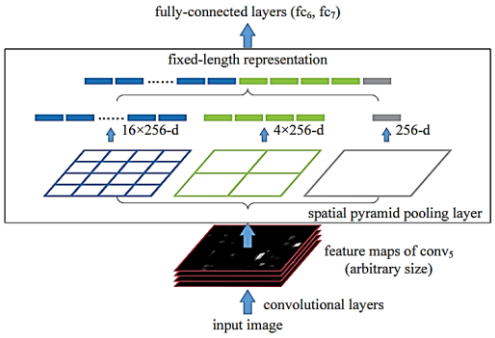


Hình 7 Atrous Convolution (tỷ lệ r = 2)

Strategy	Structure	Corpus	Original Architecture
Atrous Convolution	DeepLab	Atrous ('Hole') Convolution	FCN-VGG
	DeepLab V2	Atrous Spatial Pyramid Pooling (ASPP) Method effectively enlarge the field of view of filters to incorporate multi-scale context.	FCN-ResNet
	DeepLab V3	Rethink Atrous Convolution Augment the Atrous Spatial Pyramid Pooling (ASPP)	DeepLab V2
	DeepLab V3+	Encoder Decoder Approach Xception	DeepLab V3

Hình 8 Atrous Convolution trong DeepLab

Thách thức thứ hai là sự tồn tại của đối tượng nhiều quy mô (multiple scales object). Giải pháp cho vấn đề này được thúc đẩy bởi việc gộp Spatial Pyramid [39][40], một sơ đồ hiệu quả về mặt tính toán để thực hiện bản đồ này bằng cách sử dụng nhiều lớp phức tạp atrous parallel ở các tỷ lệ khác nhau trước khi kết hợp được đề xuất. Kỹ thuật này được gọi là Atrous Spatial Pyramid Pooling (ASPP).



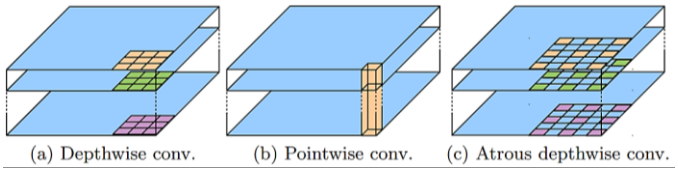
Hình 9 Một cấu trúc mạng với một lớp Atrous Spatial Pyramid Pooling.

ASPP đã được giới thiệu cùng với DeepLabv2, bao gồm chuẩn hóa hàng loạt (BN) từ Inception-v2. Khi tỷ lệ lấy mẫu trở nên lớn hơn, số lượng trọng lượng lọc hợp lệ trở nên nhỏ hơn. Với gộp kim tự tháp không gian, hình ảnh đầu vào có thể có bất kỳ kích thước nào, cho phép cả tỷ lệ khung hình và tỷ lệ tùy ý.

Thách thức thứ ba liên quan đến sự suy giảm độ chính xác nội địa, có nghĩa là một phân loại tập trung vào đối tượng đòi hỏi sự bất biến đối với các biến đổi không gian, vốn đã hạn chế độ chính xác không gian của DCNN. Một giải pháp cho tình huống này là bỏ qua các lớp sử dụng để trích xuất các tính

năng "siêu cột" từ nhiều lớp mạng khác nhau khi tính toán kết quả phân đoạn cuối cùng [19][41].

Depthwise separable convolution



Hình 10 (3 × 3) Depthwise separable convolution phân tách tích chập tiêu chuẩn thành (a) Depthwise convolution (áp dụng một lọc đơn cho mỗi kênh đầu vào) và (b) Pointwise Convolution (kết hợp các đầu ra từ Depthwise convolution qua các kênh). Đối với Atrous depthwise convolution, Atrous convolution được sử dụng Depthwise convolution, như được thể hiện trong (c) với tỷ lệ = 2.

Phân tích nhân tử hóa một phép chập chuẩn thành một depthwise convolution, sau đó là một pointwise convolution (tức là tích chập 1 × 1), làm giảm đáng kể độ phức tạp của phép tính. Cụ thể, depthwise convolution thực hiện spatial convolution độc lập cho mỗi kênh đầu vào, trong khi pointwise convolution được sử dụng để kết hợp đầu ra từ depthwise convolution. Trong việc triển khai TensorFlow [49] của depthwise seperable convolution, atrous convolution đã được hỗ trợ trong depthwise convolution (tức là spatial convolution), như được minh họa trong Hình 10. Trong tác phẩm này, coi tích chập kết quả là atrous seperable convolution, và nhận thấy rằng là atrous seperable convolution làm giảm đáng kể độ phức tạp tính toán của mô hình được đề xuất trong khi vẫn duy trì hiệu suất tương tự (hoặc tốt hơn).

DeepLabv3

DeepLab là mô hình Semantic Segmentation hiện đại được Google thiết kế từ năm 2016. Trong khi DeepLabv3 được cải thiện và vượt trội so với DeepLabv1 [26] và DeepLabv2 [27] với vô số sửa đổi. Mô hình DeepLab được tạo thành từ hai giai đoạn:

- Giai đoạn encoder: Encoder là một mạng phức tạp được đào tạo sẵn bao gồm một lớp downsampling, lấy hình ảnh đầu vào và trích xuất thông tin cần thiết.
- Giai đoạn decoder: Decoder bao gồm một lớp upsampling nhằm phục hồi thông tin không gian và một lớp decoder, tái tạo lại đầu ra của các kích thước thích hợp.

Trong DeepLabv2, các tính năng chính là:

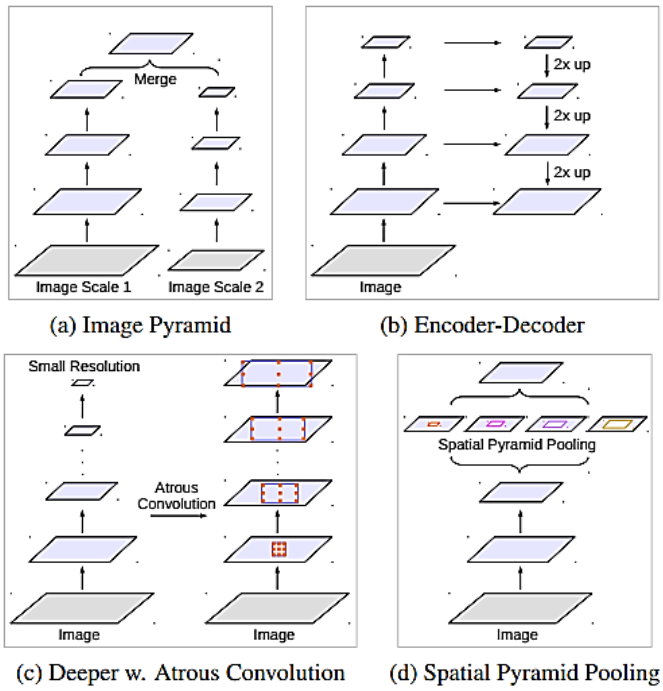
- Chụp các đối tượng và bối cảnh hình ảnh ở nhiều quy mô để phân đoạn mạnh mẽ các đối tượng bằng cách sử dụng Atrous Spatial Pyramid Pooling (ASPP).
- Giải quyết sự suy giảm độ phân giải gây ra bởi max-pooling và striding trong mạng bằng cách sử dụng sự phức tạp giãn nở.
- Cải thiện bản địa hóa ranh giới đối tượng bằng cách sử dụng các mô hình đồ họa xác suất và mạng thần kinh phức tạp sâu (DCNNs).

Trong DeepLabv3, những cải tiến chính là:

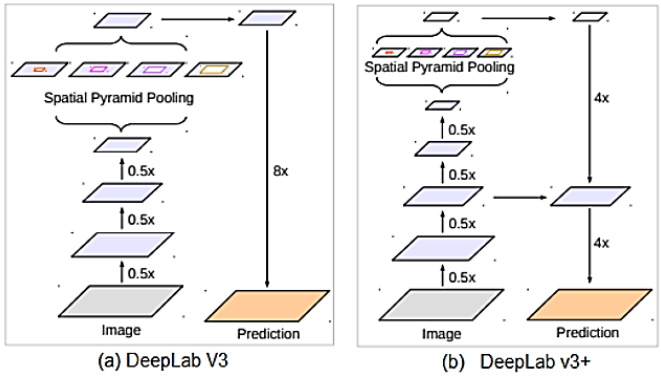
- Tiến hành ASPP phức tạp song song ở nhiều quy mô khác nhau và thêm bình thường hóa hàng loạt (batch normalization), kế thừa ý tưởng từ mạng Inception.
- Đặc biệt, loại bỏ CRF kết nối hoàn toàn ở bước xử lý cuối cùng giúp tăng tốc độ tính toán.

DeepLabv3 sử dụng Resnet-101 được đào tạo sẵn của ImageNet cùng với Atrous Convolutions làm chất chiết xuất tính năng chính. ResNet cuối cùng trong khối mô hình sửa đổi sử dụng các phức tạp atrous với tốc độ giãn nở khác nhau. Hơn nữa, DeepLabv3 sử dụng upsampling hai tuyến và Atrous Spatial Pyramid Pooling cho mô-đun decoder trên đầu trang của khối ResNet sửa đổi. Trong khi DeepLabv3+ sử dụng Aligned Xception làm chiết xuất tính năng chính, với một số sửa đổi.

DeepLab tốt nhất (sử dụng ResNet-101 làm backbone) đã đạt điểm 79,7% mIoU trong thử thách PASCAL VOC 2012, điểm 45,7% mIoU trong thử thách PASCAL-Context và 70,4% điểm mIoU trên Cityscapes.



Hình 11 Kiến trúc DeepLabv3



Hình 12 Kiến trúc DeepLabv3 và DeepLabv3+

DeepLabv3 sử dụng atrous convolution để trích xuất các đối tượng địa lý được tính toán bởi deep convolutional neural networks ở một độ phân giải tùy ý. Ở đây, biểu thị bước sóng đầu ra là tỷ lệ giữa độ phân giải không gian hình ảnh đầu vào với độ phân giải đầu ra cuối cùng (trước khi tổng hợp toàn cục hoặc lớp được kết nối hoàn toàn). Đối với nhiệm vụ phân loại ảnh, độ phân giải không gian của các bản đồ đối tượng địa lý cuối cùng thường nhỏ hơn 32 lần so với độ phân giải ảnh đầu vào và do đó stride đầu ra = 32. Đối với nhiệm vụ phân đoạn ngữ nghĩa, có thể sử dụng stride đầu ra = 16 (hoặc 8) để trích xuất tính năng dày đặc hơn bằng cách loại bỏ khoảng cách trong một (hoặc hai) khối cuối cùng và áp dụng tích chập tương ứng (ví dụ: áp dụng tỷ lệ = 2 và tỷ lệ = 4 cho hai khối cuối cùng tương ứng cho stride đầu ra = 8) . Ngoài ra, DeepLabv3 tăng cường mô-đun Atrous Spatial Pyramid Pooling, mô-đun này thăm dò các tính năng tích tụ ở nhiều tỷ lệ bằng cách áp dụng tích chập bất thường với các tỷ lệ khác nhau, với các tính năng cấp hình ảnh [50]. Sử dụng bản đồ tính năng cuối cùng trước khi đăng nhập vào DeepLabv3 ban đầu làm đầu ra encoder trong cấu trúc encoder-decoder được đề xuất của chúng tôi. Lưu ý rằng bản đồ tính năng đầu ra của encoder chứa 256 kênh và thông tin ngữ nghĩa phong phú. Bên cạnh đó, có thể trích xuất các tính năng ở độ phân giải tùy ý bằng cách áp dụng tích chập bất thường, tùy thuộc vào ngân sách tính toán.

DeepLabv3+ [51]

Trong DeepLabv3+, giai đoạn encoder trích xuất thông tin cơ bản từ hình ảnh bằng CNN trong khi giai đoạn decoder tái tạo lại các kết quả đầu ra dựa trên thông tin thu được từ giai đoạn encoder. decoder lấy lại ví dụ về hình ảnh được encoder có độ phân giải thấp về kích thước hình ảnh gốc để tạo ra kết quả phân đoạn tốt hơn qua các ranh giới đối tượng.

DeepLabV3 + hỗ trợ các mạng nền tảng sau: MobileNetv2, Xception, ResNet, PNASNet, Auto-DeepLab. Trong nghiên cứu này, đường trục mạng Xception được chọn để đào tạo mô hình DeepLabV3 + vì các tác giả của DeepLabV3 + lưu ý rằng phiên bản Xception được sửa đổi làm đường trục hứa hẹn độ chính xác của ImageNet tốt hơn.

B. Backbone / Deep Network

ResNet

Sau kiến trúc dựa trên CNN đầu tiên (AlexNet) giành chiến thắng trong cuộc thi ImageNet 2012. Mỗi kiến trúc chiến thắng tiếp theo sử dụng nhiều lớp hơn trong một mạng lưới thần kinh sâu để giảm tỷ lệ lỗi. Điều này hoạt động cho một số lượng nhỏ hơn các lớp, nhưng khi tăng số lượng lớp, có một vấn đề thường xuyên trong học sâu liên quan đến đó là Biến mất / Bùng nổ gradient. Điều này làm cho gradient trở thành 0 hoặc quá lớn. Do đó, khi tăng số lượng lớp, tỷ lệ lỗi đào tạo và kiểm tra cũng tăng lên. ResNet, được đề xuất vào năm 2015 bởi các nhà nghiên cứu tại Microsoft Research đã giới thiệu một kiến trúc mới gọi là Residual Network.

Xét $H(x)$ như một bản đồ cơ bản để phù hợp với một vài lớp xếp chồng lên nhau (không nhất thiết phải là toàn mạng), với x biểu thị các đầu vào cho lớp đầu tiên của các lớp này. Nếu đưa ra giả thuyết rằng nhiều lớp phi tuyến có thể xấp xỉ các hàm phức tạp, thì tương đương với giả thuyết rằng chúng có thể xấp xỉ các hàm còn lại (residual function), $H(x) - x$ (giả sử rằng đầu vào và đầu ra có cùng kích thước). Vì vậy, thay vì mong đợi các lớp xếp chồng lên nhau để xấp xỉ $H(x)$, để các lớp này xấp xỉ một hàm còn lại $F(x) = H(x) - x$. Hàm ban đầu trở thành $F(x) + x$. Mặc dù cả hai hình thức sẽ có thể xấp xỉ một cách không mong muốn, nhưng sự dễ dàng học tập có thể khác nhau. Nếu các lớp được thêm vào có thể được xây dựng dưới dạng bản đồ nhận dạng, một mô hình sâu hơn sẽ có lỗi đào tạo không lớn hơn bản đối chiếu. Vấn đề suy thoái cho thấy những người giải quyết có thể gặp khó khăn trong việc xấp xỉ bản đồ nhận dạng bằng nhiều lớp phi tuyến. Với việc cải cách học tập còn lại, nếu bản đồ nhận dạng là tối ưu, người giải có thể chỉ cần đẩy trọng lượng của nhiều lớp phi tuyến về số không để tiếp cận bản đồ nhận dạng. Trong trường hợp thực tế, không có khả năng lập bản đồ nhận dạng tối ưu, nhưng cải cách này có thể giúp điều kiện tiên quyết vấn đề. Nếu chức năng tối ưu gần với bản đồ nhận dạng hơn là bản đồ bằng không, người giải sẽ dễ dàng tìm thấy nhiễu loạn liên quan đến bản đồ nhận dạng hơn là tìm hiểu chức năng như một bản đồ mới. Cho thấy bằng các thí nghiệm rằng các chức năng còn lại đã học nói chung có phản ứng nhỏ, cho thấy rằng bản đồ nhận dạng cung cấp điều kiện tiên quyết hợp lý.

Xem xét một building block được định nghĩa là:

$$y = F(x, \{W_i\}) + x \qquad (1)$$

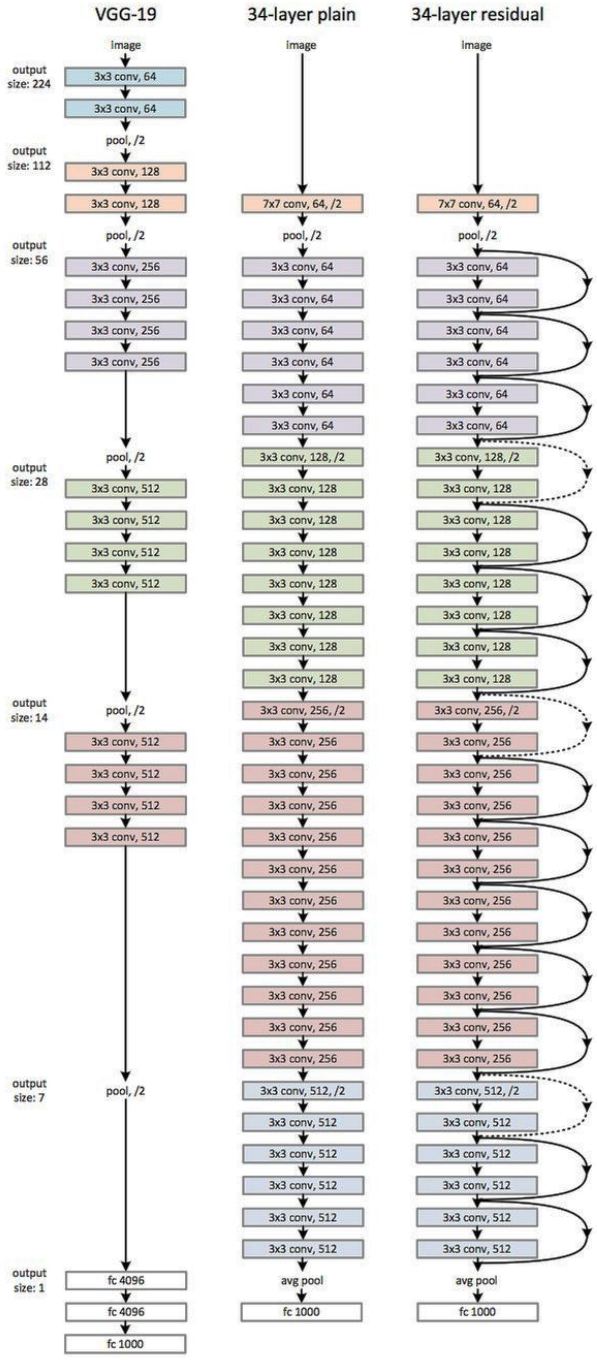
x và y là các vectơ đầu vào và đầu ra của các lớp được xem xét. Phương trình $F(x, W_i)$ đại diện cho bản đồ còn lại cần học.

Các kết nối phím tắt trong phương trình (1) không giới thiệu thêm tham số hoặc độ phức tạp tính toán. Điều này không chỉ hấp dẫn trong thực tế mà còn quan trọng trong so sánh giữa các plain network và còn lại. Có thể so sánh các plain network / còn lại có cùng số lượng tham số, độ sâu, chiều rộng và chi phí tính toán (ngoại trừ việc bổ sung yếu tố không đáng kể). Kích thước của x và F phải bằng nhau trong Phương trình (1). Nếu đây không phải là trường hợp (ví dụ: khi thay đổi kênh đầu vào/đầu ra), có thể thực hiện chiếu tuyến tính W_s bằng các kết nối lối tắt để khớp với kích thước:

$$y = F(x, W_i) + W_s x \qquad (2)$$

Cũng có thể sử dụng một ma trận hình vuông W_s trong phương trình (1). Ở đây sẽ chỉ ra bằng các thí nghiệm rằng bản đồ nhận dạng là đủ để giải quyết vấn đề suy thoái và kinh tế, và do đó W_s chỉ được sử dụng khi kết hợp kích thước.

Mặc dù các ký hiệu trên là về các lớp được kết nối đơn giản, chúng có thể áp dụng cho các lớp phức tạp. Phương trình $F(x, W_i)$ có thể đại diện cho nhiều lớp phức tạp. Việc bổ sung yếu tố khôn ngoan được thực hiện trên hai bản đồ tính năng, kênh theo kênh.



Hình 13
Trái: Mô hình VGG-19
Giữa: Plain Network.
Phải: Residual Network

Plain Network. Các đường cơ sở đơn giản (Hình 14, giữa) được lấy cảm hứng từ lưới VGG [42] (Hình 14, bên trái). Các lớp phức tạp chủ yếu có lọc 3×3 và tuân theo hai quy tắc thiết kế đơn giản: đối với cùng kích thước bản đồ tính năng đầu ra, các lớp có cùng số lọc. Nếu kích thước bản đồ tính năng giảm một nửa, số lượng lọc được tăng gấp đôi để bảo tồn độ phức tạp thời gian trên mỗi lớp. Thực hiện downsampling trực tiếp bởi các lớp phức tạp có độ dài là 2. Đối với cùng một kích thước bản đồ tính năng đầu ra, các lớp có cùng số lượng lọc. Mạng kết thúc với một lớp gộp trung bình toàn cầu và một lớp kết nối 1000 chiều. Tổng số lớp có trọng số là 34 trong Hình 14 (giữa). Điều đáng chú ý là mô hình của có ít lọc hơn và độ phức tạp thấp hơn so với lưới VGG (Hình 14, bên trái). Đường cơ sở 34 lớp của có 3,6 tỷ FLOP (nhân-thêm), chỉ bằng 18% VGG-19 (19,6 tỷ FLOP).

Kiến trúc Mạng: Dựa trên plain network ở trên, chèn các kết nối phím tắt (Hình 14, bên phải) biến mạng thành phiên bản đối chiếu còn lại. Các phím tắt nhận dạng (Phương trình (1)) có thể được sử dụng trực tiếp khi đầu vào và đầu ra có cùng kích thước (lối tắt đường rắn trong Hình 14). Khi kích thước tăng lên (các phím tắt đường chấm chấm trong Hình 14), xem xét hai tùy chọn: (A) Phím tắt vẫn thực hiện ánh xạ nhận dạng, với các mục không có thêm được đệm để tăng kích thước. Tùy chọn này không giới thiệu tham số bổ sung; (B) Phím tắt chiếu trong Phương trình (2) được sử dụng để khớp với kích thước (được thực hiện bởi 1×1 phức tạp). Đối với cả hai tùy chọn, khi các phím tắt đi qua bản đồ tính năng của hai kích cỡ, chúng được thực hiện với stride là 2.

Layer name	Output size	ResNet-101
Conv1	256x256	7x7, 64 stride 2
		3x3 max pool, stride 2
Conv2_x	128x128	$\begin{bmatrix} 1\ x\ 1.64 \\ 3\ x\ 3.64 \\ 1\ x\ 1.256 \end{bmatrix}$ x3
Conv3_x	64x64	$\begin{bmatrix} 1\ x\ 1.128 \\ 3\ x\ 3.128 \\ 1\ x\ 1.512 \end{bmatrix}$ x4
Conv4_x	32x32	$\begin{bmatrix} 1\ x\ 1.256 \\ 3\ x\ 3.256 \\ 1\ x\ 1.1024 \end{bmatrix}$ x23
Conv5_x	16x16	$\begin{bmatrix} 1\ x\ 1.512 \\ 3\ x\ 3.512 \\ 1\ x\ 1.2048 \end{bmatrix}$ x3
Fc	1x1	Average pool
		1000-d, softmax

C. Datasets

ADE20K *[31]* cung cấp một nền tảng đào tạo và đánh giá tiêu chuẩn cho các thuật toán phân tích cảnh. Dataset ADE20K chứa hơn 20K hình ảnh tập trung vào cảnh được chú thích đầy đủ với các đối tượng và các phận đối tượng

Method	Crop Size	Lr schd	Mem (GB)	Inf time (fps)	mIoU	mIoU (ms+flip)
DeepLabV3	512x512	80000	8.9	14.76	42.42	43.28
DeepLabV3	512x512	80000	12.4	10.14	44.08	45.19
DeepLabV3	512x512	160000	-	-	42.66	44.09
DeepLabV3	512x512	160000	-	-	45	46.66
DeepLabV3+	512x512	80000	10.6	21.01	42.72	43.75
DeepLabV3+	512x512	80000	14.1	14.16	44.6	46.06
DeepLabV3+	512x512	160000	-	-	43.95	44.93
DeepLabV3+	512x512	160000	-	-	45.47	46.35

Bảng 14 Kết quả hiệu suất của DeepLabv3 và DeepLabv3+ sử dụng dataset ADE20K

***Cityscapes** [30]* tập trung vào sự hiểu biết ngữ nghĩa về cảnh đường phố đô thị. Cityscapes chứa một tập hợp đa dạng các chuỗi video stereo được ghi lại trong các cảnh đường phố với chú thích pixel chất lượng cao. Cityscapes bao gồm các chú thích pixel ngữ nghĩa và dày đặc gồm 30 lớp, được nhóm thành 8 loại - bề mặt phẳng, con người, phương tiện, công trình, vật thể, thiên nhiên, bầu trời và khoảng trống.

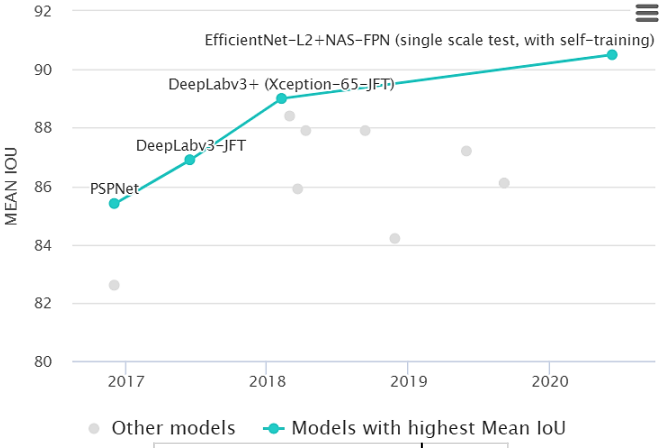
Method	Crop Size	Lr schd	Mem (GB)	Inf time (fps)	mIoU	mIoU (ms+flip)
DeepLabV3	512x1024	40000	6.1	2.57	79.09	80.45
DeepLabV3	512x1024	40000	9.6	1.92	77.12	79.61
DeepLabV3	769x769	40000	6.9	1.11	78.58	79.89
DeepLabV3	769x769	40000	10.9	0.83	79.27	80.11
DeepLabV3	512x1024	80000	1.7	13.78	76.7	78.27
DeepLabV3	512x1024	80000	-	-	79.32	80.57
DeepLabV3	512x1024	80000	-	-	80.2	81.21
DeepLabV3 (FP16)	512x1024	80000	5.75	3.86	80.48	-
DeepLabV3	769x769	80000	1.9	5.55	76.6	78.26
DeepLabV3	769x769	80000	-	-	79.89	81.06
DeepLabV3	769x769	80000	-	-	79.67	80.81
DeepLabV3	512x1024	40000	4.7	-6.96	76.71	78.63
DeepLabV3	512x1024	80000	-	-	78.36	79.84
DeepLabV3	512x1024	80000	1.6	13.93	76.26	77.88
DeepLabV3	512x1024	80000	6	2.74	79.63	80.98
DeepLabV3	512x1024	80000	9.5	1.81	80.01	81.21
DeepLabV3	769x769	80000	1.8	5.79	76.63	77.51
DeepLabV3	769x769	80000	6.8	1.16	78.8	80.27
DeepLabV3	769x769	80000	10.7	0.82	79.41	80.73
DeepLabV3+	512x1024	40000	7.5	3.94	79.61	81.01
DeepLabV3+	512x1024	40000	11	2.6	80.21	81.82
DeepLabV3+	769x769	40000	8.5	1.72	78.97	80.46
DeepLabV3+	769x769	40000	12.5	1.15	79.46	80.5
DeepLabV3+	512x1024	80000	2.2	14.27	76.89	78.76
DeepLabV3+	512x1024	80000	-	-	80.09	81.13
DeepLabV3+	512x1024	80000	-	-	80.97	82.03
DeepLabV3+ (FP16)	512x1024	80000	6.35	7.87	80.46	-
DeepLabV3+	769x769	80000	2.5	5.74	76.26	77.91
DeepLabV3+	769x769	80000	-	-	79.83	81.48
DeepLabV3+	769x769	80000	-	-	80.98	82.18
DeepLabV3+	512x1024	40000	5.8	7.48	79.09	80.36
DeepLabV3+	512x1024	80000	9.9	-	79.9	81.33
DeepLabV3+	512x1024	80000	2.1	14.95	75.87	77.52
DeepLabV3+	512x1024	80000	7.4	3.94	80.28	81.44
DeepLabV3+	512x1024	80000	10.9	2.6	80.16	81.41
DeepLabV3+	769x769	80000	2.4	5.96	76.36	78.24
DeepLabV3+	769x769	80000	8.4	1.72	79.41	80.56
DeepLabV3+	769x769	80000	12.3	1.1	79.88	81.46

Bảng 15 Kết quả thực hiện của DeepLabv3 và DeepLabv3+ sử dụng dataset Cityscapes

***Pascal Visual Object Classes (VOC)** [32]* là một trong những dataset phổ biến nhất trong thị giác máy tính, với hình ảnh được chú thích có sẵn cho 5 tác vụ - phân loại, phân đoạn, phát hiện, nhận dạng hành động và bố trí người. Đối với nhiệm vụ phân đoạn, có 21 lớp nhãn đối tượng - xe, hộ gia đình, động vật, máy bay, xe đạp, thuyền, xe buýt, xe hơi, xe máy, xe lửa, chai, ghế, bàn ăn, chậu cây, ghế sofa, TV / màn hình, chim, mèo, bò, chó, ngựa, cừu và người.

Method	Crop Size	Lr schd	Mem (GB)	Inf time (fps)	mIoU	mIoU (ms+flip)
DeepLabV3	512x512	20000	6.1	13.88	76.17	77.42
DeepLabV3	512x512	20000	9.6	9.81	78.7	79.95
DeepLabV3	512x512	40000	-	-	77.68	78.78
DeepLabV3	512x512	40000	-	-	77.92	79.18
DeepLabV3+	512x512	20000	7.6	21	75.93	77.5
DeepLabV3+	512x512	20000	11	13.88	77.22	78.59
DeepLabV3+	512x512	40000	-	-	76.81	77.57
DeepLabV3+	512x512	40000	-	-	78.62	79.53

Bảng 16 Kết quả thực hiện của DeepLabv3 và DeepLabv3+ sử dụng Resnet101 trên Pascal VOC 2012 + Aug dataset



Method	mIOU
DeepLabv3	85.7
DeepLabv3-JFT	86.9
DeepLabv3+ (Xception)	87.8
DeepLabv3+ (Xception-JFT)	89

Bảng 17 Kết quả hiệu suất của DeepLabv3 và DeepLabv3+ sử dụng các backbone khác trên dataset Pascal VOC 2012

***Pascal Context** [33]* là một phần mở rộng của thử thách phát hiện PASCAL VOC 2010, và chứa các nhãn pixel khôn ngoan cho tất cả các hình ảnh đào tạo. Pascal Context chứa hơn 400 lớp (bao gồm 20 lớp ban đầu cộng với nền từ phân đoạn PASCAL VOC), được chia thành ba loại (đối tượng, công cụ và hybrid).

Method	Crop Size	Lr schd	Mem (GB)	Inf time (fps)	mIoU	mIoU (ms+flip)
DeepLabV3	480x480	40000	9.2	7.09	46.55	47.81
DeepLabV3	480x480	80000	-	-	46.42	47.53
DeepLabV3+	480x480	40000	-	9.09	47.3	48.47
DeepLabV3+	480x480	80000	-	-	47.23	48.26

Bảng 18 Kết quả hiệu suất của DeepLabv3 và DeepLabv3+ sử dụng dataset Pascal Context

Method	Crop Size	Lr schd	Mem (GB)	Inf time (fps)	mIoU	mIoU (ms+flip)
DeepLabV3	480x480	40000	-	-	52.61	54.28
DeepLabV3	480x480	80000	-	-	52.46	54.09
DeepLabV3+	480x480	40000	-	-	52.86	54.54
DeepLabV3+	480x480	80000	-	-	53.2	54.67

Bảng 19 Kết quả hiệu suất của DeepLabv3 và DeepLabv3+ sử dụng dataset Pascal Context 59

D. Metrics

Một mô hình nên được đánh giá ở nhiều khía cạnh, chẳng hạn như độ chính xác, tốc độ và yêu cầu lưu trữ là lý tưởng nhất.

Pixel Accuracy chỉ đơn giản là tìm thấy tỷ lệ pixel được phân loại đúng cách, chia cho tổng số pixel. Đối với các lớp $K + 1$ (lớp tiền cảnh K và nền) độ chính xác pixel được định nghĩa là

$$A = \frac{\Sigma_{i=0}^{K} p_{ii}}{\Sigma_{i=0}^{K} \Sigma_{j=0}^{K} pij}$$

với p_{ij} là số pixel với lớp i dự đoán thuộc về lớp j.

Mean Pixel Accuracy (MPA) là phiên bản mở rộng của PA, trong đó tỷ lệ pixel chính xác được tính theo cách mỗi lớp và sau đó tính trung bình trên tổng số lớp.

$$MPA = \frac{1}{K+1} \Sigma_{i=0}^{K} \frac{p_{ii}}{\Sigma_{j=0}^{K} p_{ij}}$$

Intersection over Union (IoU) hoặc Chỉ số Jaccard là một trong những số liệu được sử dụng nhiều nhất trong Semantic Segmentation. IoU được định nghĩa là khu vực giao điểm giữa bản đồ phân đoạn dự đoán và thực địa, được chia cho khu vực liên minh giữa bản đồ phân đoạn dự đoán và thực địa:

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

trong đó A và B biểu thị thực địa và các bản đồ phân đoạn dự đoán tương ứng. IoU dao động trong khoảng từ 0 đến 1.

Mean-IoU là IoU trung bình trên tất cả các lớp, được sử dụng rộng rãi trong việc báo cáo hiệu suất của các thuật toán phân đoạn hiện đại.

Precision / Recall / F1 score là các chỉ số để báo cáo độ chính xác của nhiều mô hình phân đoạn hình ảnh cổ điển., có thể được xác định cho từng lớp, cũng như ở cấp tổng hợp, như sau:

$$Precision = \frac{TP}{TP + FP}, \qquad Recall = \frac{TP}{TP + FN}$$

trong đó TP đề cập đến true positive fraction, FP đề cập đến false positive fraction và FN đề cập đến false negative fraction. Thông thường, quan tâm đến phiên bản kết hợp của độ chính xác (precision) và tỷ lệ thu hồi (recall). Một số liệu phổ biến như vậy được gọi là điểm F1, được định nghĩa là trung bình hài hòa của độ chính xác và thu hồi:

$$F1score = \frac{2PrecRec}{Prec + Rec}$$

Phân loại và Đánh giá mất mát [52]

Softmax: sử dụng để chuyển đổi output của CNN thành xác suất. Softmax nhận một vector n-chiều của các số thực và biến thành một vector số thực trong phạm vi (0,1) có cộng tối đa 1.

$$p_i = \frac{e^{a_i}}{\sum_{k=1}^{N} e_k^a}$$

Như tên cho thấy, hàm softmax là phiên bản "soft (mềm)" của hàm max. Thay vì chọn một giá trị lớn nhất, softmax phá vỡ toàn bộ với phần tử tối đa nhận được phần lớn nhất của phân phối, nhưng các phần tử nhỏ hơn khác cũng nhận được một số giá trị đó. Thuộc tính này của hàm softmax tạo ra một phân phối xác suất phù hợp để giải thích xác suất trong các nhiệm vụ phân loại.

Gradient hay **Đạo hàm** của softmax được tính toán và chuyển lại các lớp trước trong quá trình backpropagation vì do thuộc tính nêu trên của softmax, softmax được sử dụng làm lớp cuối cùng trong mạng neural:

$$\frac{\partial p_i}{\partial a_j} = \begin{cases} p_i(1 - p_j) & if\ i = j \\ -p_j * p_i & if\ i \neq j \end{cases}$$

Cross-Entropy cho biết sự khác biệt giữa những gì mô hình tin rằng phân phối đầu ra phải là và phân phối đầu ra thực sự là gì. Cross-Entropy là một phương pháp thay thế được sử dụng rộng rãi cho sai số bình phương:

$$H(y,p) = -\sum_i y_i \log(p_i)$$

Đạo hàm của **Cross-entropy** với **Softmax** được sử dụng như lớp output:

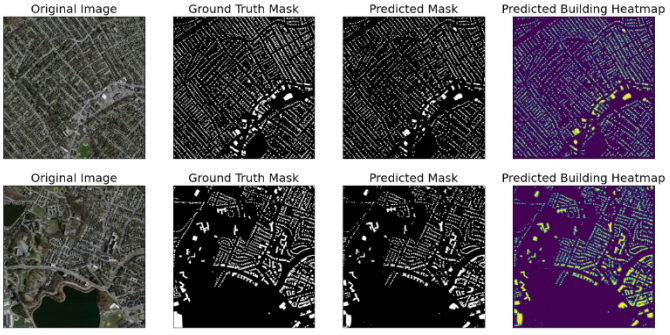
$$\frac{\partial H(y,p)}{\partial p_i} = p_i - y_i$$

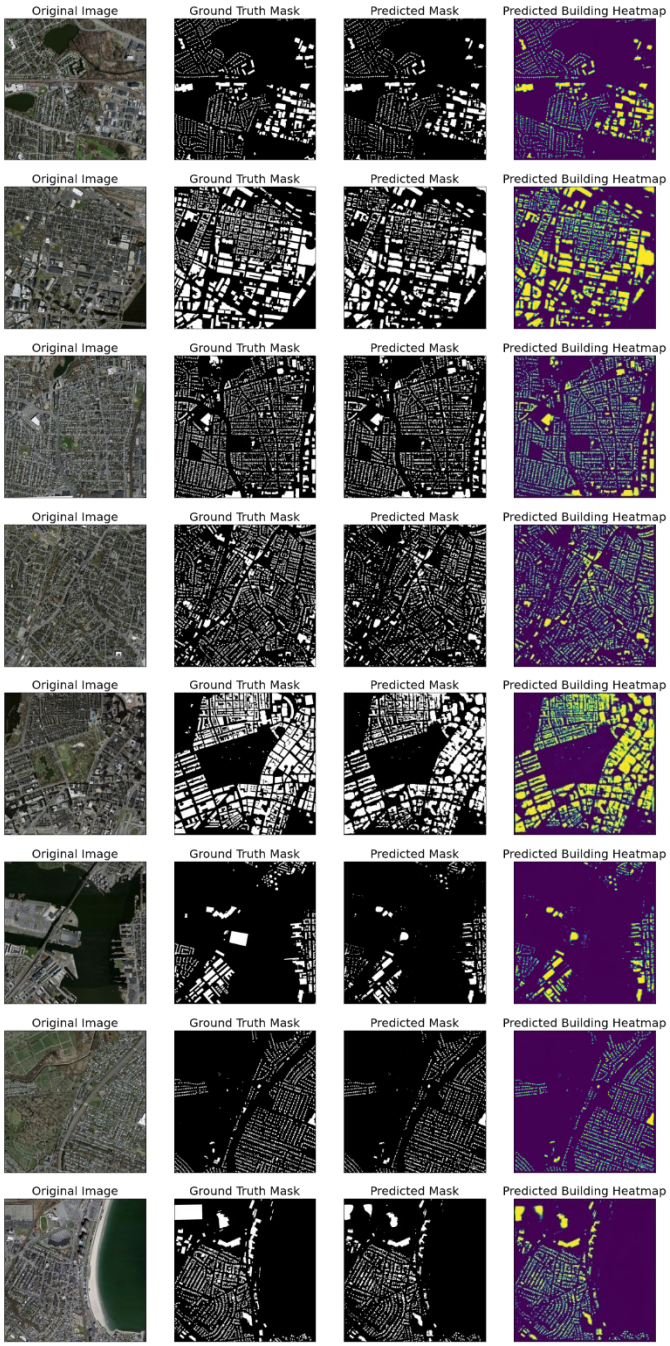
E. Phân tích và Đánh giá dataset

Massachusetts Buildings Dataset [44]:

Trực quan hóa dữ liệu:

Hình 20 Trực quan hóa dữ liệu trên một hình / mặt nạ bất kì chưa qua các phép biến đổi

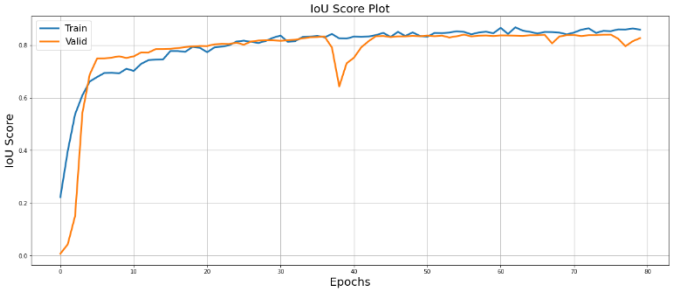




Hình 21 Trực quan hóa dữ liệu qua các phép biến đổi trên DeepLab

Đánh giá mô hình trên tập dữ liệu:

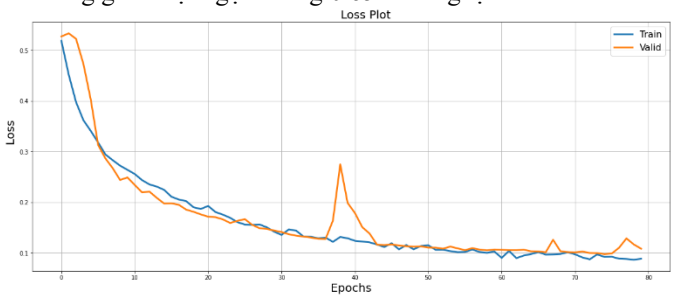
- Cross-entropy: 0.1433
- Mean IoU Score: 0.7765
- Mean Loss: 0.1433



Hình 22 Biểu đồ chỉ số IoU số liệu Train và Valid

Iou Score Plot

- Xét Epochs từ 0-5: IoU score của cả Train và Valid đều tăng trưởng đáng kể. Cụ thể, IoU score của của Train tăng từ khoảng 0.2 đến xấp xỉ 0.7, còn IoU score của Valid tăng từ 0 đến gần 0.8.
- Xét Epochs từ 5-20: IoU score của Train và Valid tăng tới khoảng mức 0.8.
- Xét Epochs từ 20-80: IoU score của Train và Valid gần bằng nhau và trên 0.8 (IoU score của Train vẫn cao hơn một ít) trừ Epoch 38. Ở Epoch 38, IoU score của Valid có xu hướng giảm đột ngột xuống 0.65 và tăng lại.



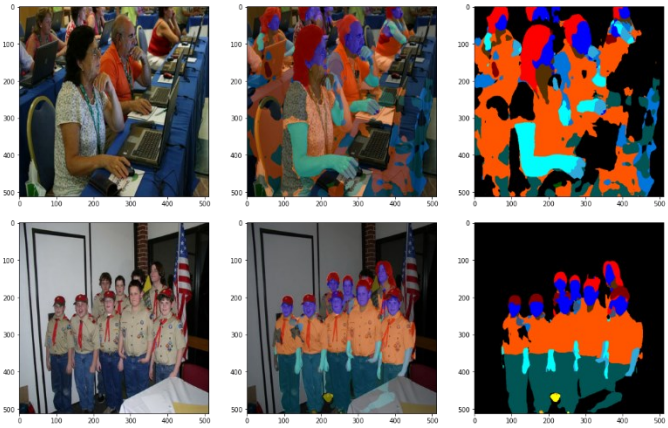
Hình 23 Biểu đồ chỉ số Loss số liệu Train và Valid

Loss plot:

- Xét Epochs từ 0-30: giá trị Loss của cả Train và Valid giảm đáng kể từ 0.5 xuống 0.15.
- Xét Epochs từ 30-80: giá trị Loss của Train và Valid đều giảm liên tục từ 0.15 xuống khoảng 0.1. Riêng ở Epochs 36-43, giá trị Loss của Valid tăng đột ngột từ 0.15 lên gần 0.3 và giảm xuống 0.13.

Crowd Instance-level Human Parsing Dataset [45]:

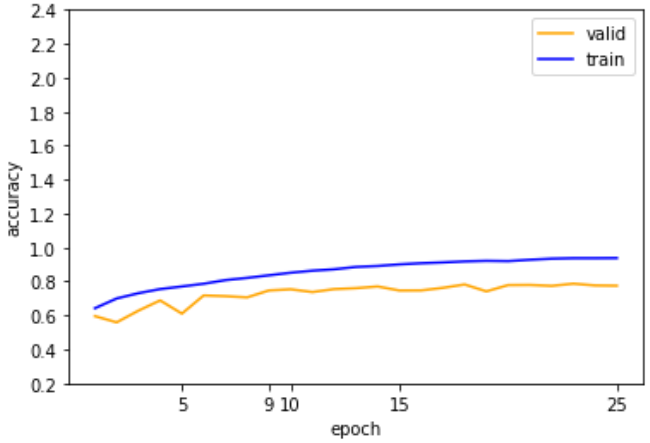
Trực quan hóa dữ liệu:



Hình 24 Trực quan hóa dữ liệu qua các phép biến đổi trên DeepLab

Đánh giá mô hình trên tập dữ liệu:

- Cross-entropy: 0.1433
- Mean IoU Score: 0.7765
- Mean Loss: 0.1433

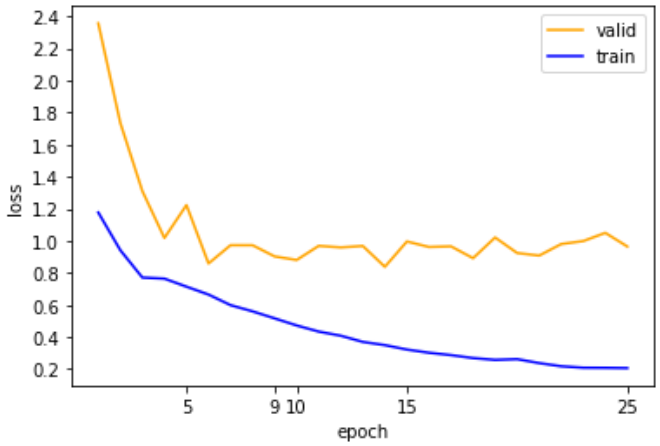


Hình 25 Biểu đồ chỉ số Accuracy số liệu Train và Valid

Nhìn chung, giá trị Accuracy của Train đều cao hơn của Valid trong Epochs từ 0-25

Xét Epochs từ 0-25, cụ thể:

- Giá trị Accuracy của Train tăng từ khoảng 0.6 đến 0.8
- Giá trị Accuracy của Valid có nhiều biến động, nhưng nhìn chung vẫn tăng nhẹ từ 0.6 đến 0.7.



Hình 26 Biểu đồ chỉ số Loss số liệu Train và Valid

Nhìn chung, giá trị Loss của Valid đều cao hơn của Train trong Epochs từ 0-25

- Xét Epochs từ 0-5: Giá trị Loss của Train giảm từ 1.2 đến khoảng 0.7. Trong khi đó giá trị Loss của Valid cũng giảm đáng kể từ khoảng 2.4 xuống khoảng 1.2.
- Xét Epochs từ 5-25: Giá trị Loss của Valid giảm liên tục từ 0.7 đến 0.2. Còn giá trị Loss của Valid có nhiều biến động hơn, nhưng nhìn chung chỉ giao động từ 1.2 xuống 1.0.

IV. KẾT LUẬN

A. Về mặt lý thuyết

Semantic segmentation là một ứng dụng quan trọng trong xử lý ảnh và thị giác máy tính. Bên cạnh việc xem xét ngắn gọn về semantic segmentation truyền thống, bài báo cáo này viết một cách toàn diện kiến trúc của DeepLabv3 trong semantic segmentation.

Một cấu trúc encoder-decoder mới sử dụng DeepLabv3 làm decoder và một cấu trúc decoder đơn giản nhưng hiệu quả, trong đó tính năng lớp thấp được nối với tính năng lớp cao được đề xuất. Atrous separable convolution để đánh đổi độ chính xác và thời gian chạy và để nhanh hơn, Atrous separable convolution được áp dụng cho cả mô-đun ASPP và mô-đun decoder.

B. Về mặt thực tiễn

So với DeepLabv3, bài báo cáo này thiết kế một decoder đơn giản để hợp nhất các tính năng cấp thấp và cao, tương ứng với chi tiết cấu trúc và thông tin ngữ nghĩa, để trích xuất thêm thông tin từ hình ảnh đầu vào. Sử dụng atrous and depthwise separable convolution để máy tính nhanh hơn trong khi vẫn duy trì độ chính xác khá giống nhau. Mô hình ResNet-101 được sử dụng làm backbone ở đây để có hiệu suất tốt hơn (cả độ chính xác và tốc độ). Tuy nhiên, số lượng tham số không lớn lắm.

C. Giới hạn nghiên cứu

Các chi tiết được khôi phục từ tính năng độ phân giải thấp của đầu ra decoder không đủ tốt khi sử dụng mô hình DeepLab V3. DeepLabv3 không thực hiện nhanh trong quá trình suy luận. Đồng thời, do thời gian làm nghiên cứu còn hạn chế nên có thể đã bỏ qua một số yếu tố có thể ảnh hưởng đến bài nghiên cứu.

V. THAM KHẢO

[1] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(12):2481–2495, 2017.

[2] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Transactions on Pattern Analysis and Machine Intelligence, 40(4):834–848, 2017.

[3] Hieu Le, Tomas F. Yago Vicente, Vu Nguyen, Minh Hoai, and Dimitris Samaras. A+D Net: Training a shadow detector with adversarial shadow attenuation. In Proceedings of the European Conference on Computer Vision, 2018.

[4] Chenxi Liu, Liang-Chieh Chen, Florian Schroff, Hartwig Adam, Wei Hua, Alan L Yuille, and Li Fei-Fei. Autodeeplab: Hierarchical neural architecture search for semantic image segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019.

[5] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015.

[6] Vu Nguyen, Tomas F. Yago Vicente, Maozheng Zhao, Minh Hoai, and Dimitris Samaras. Shadow detection with conditional generative adversarial networks. In Proceedings of the International Conference on Computer Vision, 2017.

[7] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. Unet: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention, 2015.

[8] Hafiz, A. M., & Bhat, G. M. (2020, June 28). A survey on instance segmentation: State of the art.

[9] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In CVPR, 2014.

[10] B. Hariharan, P. Arbelaez, R. Girshick, and J. Malik. Simultaneous detection and segmentation. In ECCV. 2014.

[11] J. Dai, K. He, and J. Sun. Convolutional feature masking for joint object and stuff segmentation. In CVPR, 2015.

[12] P. Arbelaez, J. Pont-Tuset, J. T. Barron, F. Marques, and J. Malik. Multiscale combinatorial grouping. In CVPR, 2014.

[13] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. IJCV, 2013.

[14] P. O. Pinheiro, R. Collobert, and P. Dollar. Learning to segment object candidates. In NIPS, 2015.

[15] P. O. Pinheiro, T.-Y. Lin, R. Collobert, and P. Dollar. Learning to refine object segments. In ECCV, 2016.

[16] J. Dai, K. He, Y. Li, S. Ren, and J. Sun. Instance-sensitive fully convolutional networks. In ECCV, 2016.

[17] R. Girshick. Fast R-CNN. In ICCV, 2015.

[18] Elharroussa, O., Himeurb, Y., Almaadeeda, N., Ottakatha, N., Subramaniana, N., & Maadeeda, S. (2021). Panoptic Segmentation: A Review

[19] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3431–3440.

[20] Mohan, R., & Valada, A. (2021). EfficientPS: Efficient Panoptic Segmentation. Trong International Journal of Computer Vision (Vol 129, Issue 5, tr 1551–1579). Springer Science and Business Media LLC.

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In NIPS, 2012.

[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in ICLR, 2015.

[23] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," arXiv:1409.4842, 2014.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[25] Le, J. (2021, May 20). How to do semantic segmentation using Deep Learning, from https://nanonets.com/blog/how-to-do-semantic-segmentation-using-deep-learning/

[26] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," arXiv preprint arXiv:1412.7062, 2014

[27] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," IEEE transactions on pattern analysis and machine intelligence, vol. 40, no. 4, pp. 834–848, 2017.

[28] Semantic segmentation tutorial: Semantic segmentation model. Analytics Vidhya. (2020, May 24).

[29] Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., & Terzopoulos, D. (2020, November 15). Image segmentation using Deep Learning: A Survey.

[30] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 3213–3223

[31] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, "Scene parsing through ade20k dataset," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[32] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," International journal of computer vision, vol. 88, pp. 303–338, 2010.

[33] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, and A. Yuille, "The role of context for object detection and semantic segmentation in the wild," in IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 891–898.

[34] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Doll´ar, and C. L. Zitnick, "Microsoft coco: Common objects in context," in European conference on computer vision. Springer, 2014.

[35] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in Proc. IEEE, 1998.

[36] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," arXiv:1312.6229, 2013.

[37] G. Papandreou, I. Kokkinos, and P.-A. Savalle, "Modeling local and global deformations in deep learning: Epitomic convolution, multiple instance learning, and sliding window detection," in CVPR, 2015.

[38] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov, "Scalable object detection using deep neural networks," in CVPR, 2014.

[39] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in CVPR, 2006.

[40] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in ECCV, 2014.

[41] B. Hariharan, P. Arbelaez, R. Girshick, and J. Malik, "Hyper- ´ columns for object segmentation and fine-grained localization," in CVPR, 2015.

[42] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In ICLR, 2015.

[43] Lin, T., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017, April 19). Feature Pyramid Networks for Object Detection, arXiv: 1612.03144v2.

[44] V. Mnih, "Machine learning for aerial image labeling," Ph.D. dissertation, University of Toronto, 2013.

[45] Yang, L., Song, Q., Wang, Z., & Jiang, M. (2018, November 30). Parsing R-CNN for instance-level human analysis, from https://arxiv.org/abs/1811.12596

[46] Newell, A., Yang, K., Deng, J.: Stacked hourglass networks for human pose estimation. In: ECCV. (2016)

[47] Shrivastava, A., Sukthankar, R., Malik, J., Gupta, A.: Beyond skip connections: Top-down modulation for object detection. arXiv:1612.06851 (2016)

[48] Fu, C.Y., Liu, W., Ranga, A., Tyagi, A., Berg, A.C.: Dssd: Deconvolutional single shot detector. arXiv:1701.06659 (2017)

[49] Abadi, M., Agarwal, A., et al.: Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv:1603.04467 (2016)

[50] Liu, W., Rabinovich, A., Berg, A.C.: Parsenet: Looking wider to see better. arXiv:1506.04579 (2015)

[51] Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., & Adam, H. (2018, August 22). Encoder-decoder with atrous separable convolution for Semantic Image segmentation. , from https://arxiv.org/abs/1802.02611v3

[52] Dahal, P. (2017, May 28). Classification and loss evaluation - softmax and cross entropy loss. DeepNotes. Retrieved April 20, 2022, from https://deepnotes.io/softmax-crossentropy