**CHÚ THÍCH CÁC ĐỐI TƯỢNG VỚI POLYGON RNN**

**Tóm tắt**

Chúng tôi đề xuất một cách tiếp cận cho chú thích bán tự động của các đối tượng. Trong khi hầu hết các phương thức hiện tại coi phân đoạn đối tượng là một vấn đề gắn nhãn pixel, ở đây chúng tôi đặt nó như là một nhiệm vụ dự đoán đa giác, bắt chước làm thế nào hầu hết các bộ dữ liệu hiện tại đã được chú thích. Cụ thể, cách tiếp cận của chúng tôi lấy làm đầu vào một ảnh crop và tuần tự tạo ra các đỉnh của đa giác phác thảo đối tượng. Điều này cho phép một người chú thích can thiệp bất cứ lúc nào và sửa một đỉnh nếu cần, tạo ra phân đoạn chính xác như mong muốn bởi người chú thích. Chúng tôi cho thấy rằng phương pháp của chúng tôi tăng tốc quá trình chú thích theo hệ số 4.7 trên tất cả các lớp trong Cityscapes, trong khi đạt được thỏa thuận 78.4% trong IoU với ground-truth ban đầu, phù hợp với thỏa thuận điển hình giữa các người chú thích. Đối với ô tô, hệ số tăng tốc của chúng tôi là 7,3 cho thỏa thuận là 82: 2%. Chúng tôi tiếp tục cho thấy khả năng khái quát hóa của phương pháp tiếp cận các bộ dữ liệu chưa thấy.

1. **Giới thiệu**

Phân đoạn hình ảnh ngữ nghĩa đã nhận được sự chú ý đáng kể trong cộng đồng [5, 17]. Với các điểm chuẩn mới như Cityscapes [6], phân đoạn đối tượng cũng thu được hơi nước [14, 24, 34, 21, 29]. Hầu hết các phương pháp tiếp cận dựa trên các mạng thần kinh, đạt được hiệu suất ấn tượng cho các nhiệm vụ này [5, 17, 10, 21]. Tuy nhiên, phương pháp học tập sâu là dữ liệu đói và hiệu suất của chúng có mối tương quan chặt chẽ với lượng dữ liệu đào tạo có sẵn. Điều này đòi hỏi cộng đồng phải chú thích các bộ dữ liệu quy mô lớn, vừa tốn thời gian vừa tốn kém. Mục tiêu của chúng tôi trong bài viết này là làm cho quá trình này nhanh hơn, đồng thời mang lại ground-truth chính xác như chính xác có sẵn trong các bộ dữ liệu hiện tại.

Đã có một số nỗ lực nhằm giảm sự phụ thuộc vào chú thích rất chi tiết chẳng hạn như mặt nạ phân tách đối tượng. Trong việc thiết lập weakly-supervised, các phương pháp tiếp cận nhằm mục đích học các mô hình phân đoạn từ chú thích yếu như thẻ hình ảnh hoặc hộp giới hạn [13, 31, 11]. Trong [15], các tác giả dựa vào những nét vẽ nguệch ngoạc, một trên mỗi đối tượng, trong khi [1] chỉ yêu cầu một điểm duy nhất trên đối tượng. Mặc dù các phương pháp này đầy hứa hẹn, nhưng hiệu suất của chúng vẫn chưa cạnh tranh với các phương pháp được giám sát đầy đủ. Công việc khác khai thác dễ dàng hơn để có được ground-truth chẳng hạn như hộp giới hạn và tạo nhãn (nhiễu) bên trong mỗi hộp với cách tiếp cận GrabCut [25, 4]. Nó đã được chứng minh rằng một ký hiệu như vậy có thể đóng vai trò là dữ liệu phụ trợ hữu ích để huấn luyện các mạng phân đoạn thần kinh [36, 29]. Tuy nhiên, các phân đoạn này không thể được sử dụng như là ground-truth chính thức cho điểm chuẩn do các quy định vốn có của nó.

Hầu hết các bộ dữ liệu phân đoạn quy mô lớn đã được thu thập bằng cách có các chú thích phác thảo các đối tượng bằng một đa giác [8, 18, 16, 6, 37]. Vì các đối tượng thường được kết nối và không có lỗ, đa giác cung cấp cách chú thích một đối tượng với số lần nhấp tương đối nhỏ, thường khoảng 30 đến 40 mỗi đối tượng. Trong bài báo này, chúng tôi đề xuất một phương pháp phân đoạn tương tác tạo ra các chú thích đối tượng kết hợp chính xác và có cấu trúc cao, và giảm thời gian chú thích theo hệ số 4.7.

Đưa ra một hộp giới hạn ground-truth, các phương thức của chúng tôi đã sử dụng một đa giác phác thảo đối tượng đối tượng bằng cách sử dụng Mạng thần kinh hồi quy, mà chúng tôi gọi là Polygon-RNN. Cách tiếp cận của chúng tôi lấy đầu vào là một hình ảnh và tạo ra các đỉnh của đa giác phác thảo đối tượng. Điều này cho phép một người chú thích can thiệp bất cứ lúc nào và chỉnh lại một đỉnh nếu cần, tạo ra các phân đoạn chính xác như mong muốn của bộ chú thích. Chúng tôi cho thấy rằng phương pháp chú thích của chúng tôi tăng tốc quá trình chú thích theo hệ số 4.7, trong khi đạt được thỏa thuận 78. 4% với ground-truth ban đầu, phù hợp với thỏa thuận điển hình của chú thích con người.

1. **Công việc liên quan**

Cách tiếp cận của chúng tôi liên quan đến công việc chú thích hình ảnh bán tự động và phân đoạn đối tượng.

**Chú thích bán tự động.** Đã có nỗ lực đáng kể trong việc làm cho nhãn hình ảnh ở mức pixel nhanh hơn cho các chú thích. Trong [2], các tác giả đã sử dụng những nét vẽ nguệch ngoạc như nguồn gốc để mô hình hóa sự xuất hiện của tiền cảnh và hậu cảnh,và thực hiện phân đoạn thông qua việc cắt đồ thị bằng cách kết hợp các dấu hiệu độ trễ và thuật ngữ độ mịn [3]. [19] sử dụng nhiều nét vẽ nguệch ngoạc trên đối tượng và nền và khai thác tín hiệu chuyển động để chú thích một đối tượng trong video. Những nét vẽ nguệch ngoạc gần đây cũng được sử dụng trong [15] để đào tạo CNN cho phân đoạn hình ảnh ngữ nghĩa. GrabCut [25] khai thác các chú thích dưới dạng hộp giới hạn 2D và thực hiện ghi nhãn theo pixel với các mô hình nền trước / nền sau bằng EM. Xây dựng dựa trên ý tưởng này, [23] đã kết hợp GrabCut với CNN để phân đoạn hình ảnh y tế. Trong [4], các tác giả đã khai thác các hộp giới hạn 3D và một đám mây điểm để tạo điều kiện ghi nhãn. Một cách tiếp cận khác là khai thác nhiều hộp giới hạn và thực hiện đồng phân đoạn [13, 11].

Vì hầu hết các cách tiếp cận này xác định một mô hình đồ họa ở cấp độ pixel, với thuật ngữ độ mịn là mối quan hệ chính giữa các pixel, rất khó để kết hợp các hình dạng nguyên mẫu. Chúng đặc biệt quan trọng trong các vùng mơ hồ gây ra bởi bóng, độ bão hòa hình ảnh hoặc độ phân giải thấp của đối tượng. Hơn nữa, không có gì ngăn cản các mô hình này cung cấp nhãn lủng lỗ. Nếu phương pháp phạm sai lầm trong việc phác thảo đối tượng, thì người chú thích sẽ có một công việc khó khăn và tẻ nhạt để sửa chữa cho những sai lầm đó. Do đó, các phương pháp này chủ yếu được sử dụng để tạo ra các ví dụ đào tạo bổ sung nhưng không gây nhiễu, nhưng đầu ra của chúng thường không đủ chính xác để đóng vai trò là ground-truth chính thức của điểm chuẩn.

**Phân đoạn đối tượng.** Công việc của chúng tôi cũng được đưa vào phân khúc đối tượng. Hầu hết các cách tiếp cận này [14, 24, 36, 34, 20, 21] hoạt động ở cấp độ pixel, thường khai thác CNN bên trong hộp hoặc miếng vá để thực hiện ghi nhãn. Công việc liên quan nhiều nhất đến chúng ta là [35, 28] nhằm mục đích tạo ra một đa giác xung quanh một đối tượng. Các phương pháp này bắt đầu bằng cách phát hiện các mảnh vỡ cạnh và tìm một chu kỳ tối ưu liên kết các cạnh thành một khu vực kết hợp. Trong [7], các tác giả đề xuất một phương pháp tạo ra các pixel siêu lớn từ các đa giác nhỏ mà chúng kết hợp thành các vùng đối tượng với mục đích gắn nhãn ảnh trên không. Trong công việc của chúng tôi, chúng tôi dự đoán trực tiếp đa giác xung quanh đối tượng, sử dụng RNN được thiết kế cẩn thận.

1. **Đa giác-RNN**

Mục tiêu của chúng tôi là tạo ra một công cụ chú thích hiệu quả cho các trường hợp đối tượng ghi nhãn với đa giác. Như thông thường trong thiết lập chú thích, chúng tôi giả sử rằng người dùng cung cấp hộp giới hạn xung quanh đối tượng. Đưa ra bản vá hình ảnh bên trong hộp, phương pháp của chúng tôi dự đoán một đa giác (đóng) phác thảo đối tượng bằng cách sử dụng Mạng thần kinh hồi quy. Chúng tôi cho phép người dùng sửa một đỉnh dự đoán của đa giác bất cứ lúc nào nếu cần, chúng tôi tích hợp trong nhiệm vụ dự đoán của mình.

Chúng tôi tham số hóa đa giác dưới dạng một chuỗi các đỉnh 2D (ct) tN, c R2. Chúng tôi giả sử đa giác được đóng, tức là, có một cạnh giữa bất kỳ hai đỉnh liên tiếp, cũng như các đỉnh cuối cùng và đầu tiên. Lưu ý rằngmột đa giác khép kín là một chu kỳ và do đó có nhiều tham số tương đương thu được bằng cách chọn bất kỳ đỉnh nào làm điểm bắt đầu của chuỗi, cũng như chọn hướng của trình tự. Ở đây, chúng tôi sửa đa giác luôn theo hướng kim đồng hồ, nhưng điểm bắt đầu của chuỗi có thể là bất kỳ đỉnh nào.

Mô hình của chúng tôi là một RNN, dự đoán một đỉnh ở mọi thời điểm. Là đầu vào trong mỗi bước của RNN, chúng tôi sử dụng đại diện CNN của ảnh cắt, cũng như các đỉnh được đặt trước một và hai bước thời gian trước, cộng với điểm đầu tiên. Bằng cách cung cấp thông tin rõ ràng về hai điểm trước đây, chúng tôi giúp RNN tuân theo một định hướng cụ thể của đa giác. Mặt khác, đỉnh đầu tiên giúp RNN quyết định khi nào đóng (kết thúc) đa giác. Chúng tôi đào tạo mô hình RNN + CNN từ đầu đến cuối. Điều này về cơ bản giúp CNN được điều chỉnh phù hợp với các ranh giới đối tượng, trong khi RNN học cách tuân theo các ranh giới này và khai thác tính chất lặp lại của nó để mã hóa ưu tiên trên các hình dạng đối tượng. Mô hình của chúng tôi do đó trả về một đại diện mạch lạc về cấu trúc của đối tượng. Chúng tôi đặt tên cho mô hình của chúng tôi Polygon-RNN.

Hình 2 cho thấy tổng quan của mô hình. Chúng tôi tiếp theo ghi chép lại từng thành phần của mô hình chi tiết hơn.

* 1. ***Kiến trúc mô hình***

Chúng tôi bắt đầu bằng cách cung cấp chi tiết về biểu diễn hình ảnh qua CNN và sau đó giải thích thiết kế của RNN.

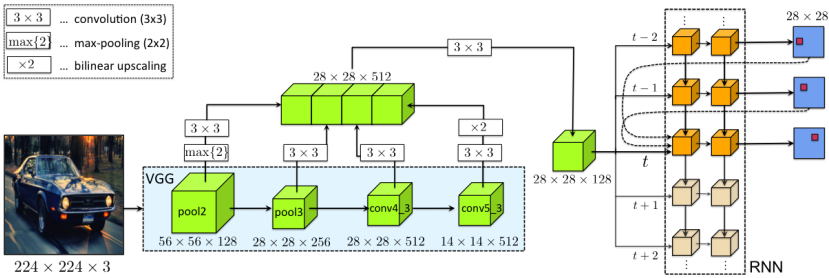


Figure 2. Mô hình đa giác-RNN của chúng tôi. Ở mỗi bước của bộ giải mã RNN (phải), chúng tôi cung cấp một biểu diễn hình ảnh bằng kiến ​​trúc VGG đã sửa đổi. RNN của chúng tôi là một LSTM tích chập hai lớp với kết nối bỏ qua từ một và hai bước thời gian trước. Ở đầu ra ở mỗi bước thời gian, chúng tôi dự đoán vị trí không gian của đỉnh mới của đa giác.

* + 1. *Đại diện hình ảnh qua CNN với Bỏ qua kết nối*

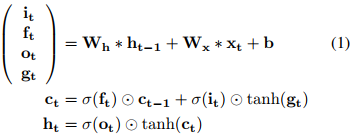
Chúng tôi áp dụng kiến ​​trúc VGG-16 [27] và sửa đổi nó cho mục đích nhiệm vụ của chúng tôi. Trước tiên, chúng tôi loại bỏ các lớp được kết nối đầy đủ cũng như lớp tổng hợp tối đa cuối cùng, pool5.Việc đặt ra của mạng được sửa đổi này có hệ số downsampling là 16. Sau đó, chúng tôi thêm các lớp chập bổ sung với các kết nối bỏ qua kết hợp thông tin từ các lớp trước và nâng cao đầu ra theo hệ số 2 (hệ số lấy mẫu xuống là 8 wrt đến kích thước ban đầu của ảnh cắt, đó là các cách được chia tỷ lệ thành 224 × 224). Điều này cho phép CNN trích xuất các tính năng chứa cả thông tin cấp thấp về các cạnh và góc, cũng như thông tin ngữ nghĩa về đối tượng. Cái sau giúp mô hình nhìn thấy đối tượng, trong khi cái trước giúp nó đi theo ranh giới của đối tượng.

Chúng tôi sử dụng một kiến ​​trúc tương tự cho các kết nối bỏ qua như kết nối trong [21]. Nguyên tắc thiết kế là trước tiên xử lý các tính năng trong các lớp bỏ qua bằng cách sử dụng một lớp chập khác, sau đó nối tất cả các đầu ra,và thường xử lý tính năng nối này bằng cách sử dụng một lớp chập khác. Chúng tôi sử dụng các bộ lọc tích chập với kích thước kernel là 3 × 3, theo sau là phi tuyến tính ReLU. Các lớp catenation nối các kênh của các đầu ra khác nhau thành một tensor duy nhất. Vì chúng tôi sử dụng các tính năng từ nhiều lớp bỏ qua có kích thước không gian khác nhau, chúng tôi sử dụng phương pháp ghép mẫu song song hoặc gộp nhóm tối đa để có được kết quả đầu ra mà tất cả đều có cùng độ phân giải không gian. Chúng tôi giới thiệu người đọc đến hình 2 để trực quan hóa và biết thêm chi tiết về kiến ​​trúc (CNN được tô sáng màu xanh lá cây).

* + 1. *RNN cho Dự đoán đỉnh*

Một RNN là một đại diện mạnh mẽ của dữ liệu chuỗi thời gian, vì nó mang thông tin phức tạp hơn về lịch sử bằng cách sử dụng các hàm tuyến tính và phi tuyến tính. Trong trường hợp của chúng tôi, chúng tôi hy vọng RNN sẽ nắm bắt được hình dạng của đối tượng và do đó đưa ra dự đoán mạch lạc ngay cả trong các trường hợp mơ hồ, chẳng hạn như bóng tối và bão hòa.

Cụ thể, chúng tôi sử dụng LSTM Convolutional [30] trong mô hình của chúng tôi và sử dụng nó làm bộ giải mã.ConvLSTM hoạt động ở dạng 2D, cho phép chúng tôi lưu giữ thông tin không gian nhận được từ CNN. Hơn nữa, một ConvLSTM sử dụng các cấu trúc, điều này làm giảm đáng kể số lượng tham số được học so với sử dụng RNN được kết nối đầy đủ. Ở dạng đơn giản nhất, một ConvLSTM (lớp đơn) tính toán trạng thái ẩn ht cho đầu vào xt theo các phương trình sau:



Ở đây *i, f, o* biểu thị cổng đầu vào, quên và cổng đầu ra, *h* là trạng thái ẩn và *c* là trạng thái ô. biểu thị hàm sigmoid, 🞊 biểu thị một sản phẩm có yếu tố thông minh và tích chập. *Wh* biểu thị kernel tích chập ẩn và *Wx*hạt nhân chập đầu vào trạng thái. Cụ thể, chúng tôi mô hình đa giác với ConvLSTM hai lớp với kích thước kernel là 3 × 3 và 16 kênh, tạo ra một đỉnh ở mỗi bước thời gian. Chúng tôi xây dựng dự đoán đỉnh như một nhiệm vụ phân loại. Cụ thể, chúng tôi biểu thị đầu ra của chúng tôi tại thời điểm t là mã hóa one-hot của lưới *D × D + 1*, trong đó kích thước *D × D* đại diện cho các vị trí 2D có thể có của đỉnh, và thứ nguyên cuối cùng tương ứng với mã thông báo cuối chuỗi (nghĩa là, đa giác được đóng). Do đó, vị trí của các đỉnh được lượng tử hóa theo độ phân giải của lưới đầu ra. Gọi *yt* là mã hóa one-hot của một đỉnh, đầu ra ở bước thời gian *t*. ConvLSTM của chúng tôi nhận đầu vào là một tensor *xt* tại bước *t*, kết hợp nhiều tính năng: đại diện tính năng CNN của hình ảnh, *yt − 1* và *yt − 2*, tức là, một mã hóa one-hot của đỉnh được dự đoán trước đó và đỉnh được dự đoán từ hai bước thời gian trước, cũng như mã hóa one-hot của đỉnh *y1* được tiên đoán đầu tiên.

Cho hai đỉnh liên tiếp, đỉnh tiếp theo trên đa giác được xác định duy nhất. Tuy nhiên, đây không phải là trường hợp của đỉnh đầu tiên, vì bất kỳ đỉnh nào của đa giác đều có thể đóng vai trò là điểm bắt đầu (đa giác là một chu kỳ). Do đó, chúng tôi coi điểm bắt đầu là đặc biệt và dự đoán nó theo cách sau. Chúng tôi sử dụng lại kiến ​​trúc tương tự của CNN như trong Sec. 3.1.1, nhưng thêm hai lớp, mỗi chiều *D × D*. Một nhánh dự đoán các ranh giới đối tượng trong khi nhánh kia lấy đầu vào của lớp dự đoán ranh giới cũng như các tính năng hình ảnh và dự đoán các đỉnh của đa giác. Chúng tôi coi cả hai, ranh giới và đỉnh là một vấn đề phân loại nhị phân trong mỗi ô trong lưới đầu ra.

* 1. ***Đào tạo***

Để đào tạo mô hình của chúng tôi, chúng tôi sử dụng entropy chéo tại mỗi bước thời gian của RNN. Để không phạt quá mức các dự đoán kết hợp gần với đỉnh của ground-truth, chúng tôi làm mịn phân phối mục tiêu của mình ở mỗi bước. Chúng tôi chỉ định khối lượng xác suất khác không cho các vị trí nằm trong khoảng cách 2 trong lưới đầu ra D × D của chúng tôi.

Chúng tôi tuân theo chế độ đào tạo điển hình nơi chúng tôi đưa ra dự đoán ở mỗi bước thời gian nhưng cung cấp thông tin về đỉnh của ground-truth tiếp theo. Chúng tôi đào tạo mô hình của mình bằng trình tối ưu hóa Adam [12] với kích thước lô b = 8 và tỷ lệ học ban đầu là λ = 1*e* - 4. Chúng tôi phân rã tỷ lệ học tập sau 10 epoch theo hệ số 10 và sử dụng các giá trị mặc định là β1 = 0.9 và β2 = 0.999.

Đối với nhiệm vụ dự đoán đỉnh đầu tiên, chúng tôi đào tạo một CNN khác bằng cách sử dụng mất mát tuyến tính. Cụ thể, chúng tôi sử dụng tổn thất logistic cho mọi vị trí trong lưới. Là ground-truth cho các ranh giới đối tượng, chúng ta vẽ các cạnh của đa giác ground-truth và sử dụng các đỉnh của đa giác làm ground-truth cho lớp đỉnh. Mô hình đầy đủ của chúng tôi mất khoảng một ngày để đào tạo trên GPU Nvidia Titan-X.

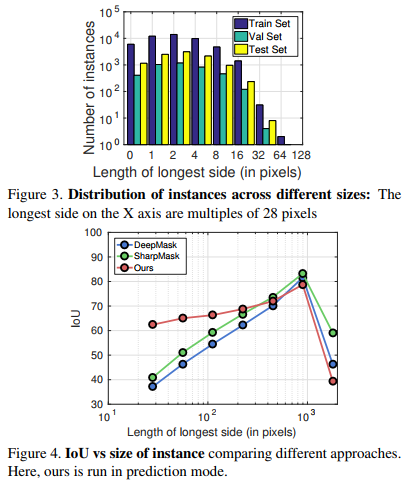
* 1. ***Suy luận và chú thích trong vòng lặp***

Suy luận trong mô hình của chúng tôi được thực hiện bằng cách lấy đỉnh với log-prob cao nhất ở mỗi bước thời gian của RNN. Điều này cho phép một giao diện chú thích đơn giản: bộ chú thích có thể sửa dự đoán bất cứ lúc nào, và chúng tôi cung cấp trong đỉnh đã sửa cho bước thời gian tiếp theo của RNN (thay vì dự đoán). Điều này đặt mô hình trở lại vào đúng hướng. Thời gian suy luận điển hình là 250 ms mỗi đối tượng.

* 1. ***Chi tiết thực hiện***

Chúng tôi dự đoán đa giác ở độ phân giải D × D.Trong các tình huống cũ của chúng tôi, chúng tôi đã sử dụng D = 28, tương ứng với hệ số lấy mẫu giảm xuống 8 lần với độ phân giải đầu vào và khớp với độ phân giải của ConvLSTM. Chúng tôi thực hiện đơn giản hóa đa giác với sai số 0 trong lưới lượng tử hóa để loại bỏ các đỉnh nằm trên một đường thẳng và để loại bỏ nhiều đỉnh sẽ rơi vào cùng một vị trí lưới do kết quả của quá trình lượng tử hóa.

Chúng tôi thực hiện ba loại tăng dữ liệu khác nhau: (1) chúng tôi lật ngẫu nhiên các ảnh cắt và chú thích đa giác tương ứng, (2) chúng tôi chọn ngẫu nhiên số lượng mở rộng bối cảnh (mở rộng hộp) trong khoảng từ 10% đến 20% của khung giới hạn ban đầu và (3) chúng tôi chọn ngẫu nhiên đỉnh bắt đầu của chú thích đa giác của chúng tôi.



1. **Các kết quả**

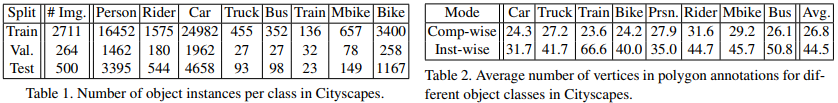
Chúng tôi đánh giá cách tiếp cận của chúng tôi đối với nhiệm vụ chú thích đối tượng trên bộ dữ liệu Cityscapes [6] và cung cấp kết quả bổ sung trên KITTI [9]. Lưu ý rằng trong tất cả các thí nghiệm của chúng tôi, chúng tôi giả sử được cung cấp một hộp ground-truth xung quanh đối tượng. Mục tiêu của chúng tôi sau đó là cung cấp một đa giác phác thảo chính xác đối tượng này càng chính xác càng tốt và với số lần nhấp tối thiểu được yêu cầu từ bộ chú thích. Chúng tôi báo cáo hiệu suất của chúng tôi bằng thước đo IOU tiêu chuẩn, cũng như số lần hiệu chỉnh đỉnh của đa giác dự đoán. Một hộp xung quanh đối tượng về nguyên tắc yêu cầu hai lần nhấp bổ sung. Tuy nhiên, các hộp thường dễ dàng hơn và rẻ hơn nhiều khi sử dụng các dịch vụ tìm nguồn cung ứng đám đông như AMT, trong khi đối với hầu hết các tiêu chuẩn phân khúc chính, đa giác đã được kết nối với các chú thích chất lượng cao (trong nhà).

* 1. ***Bộ dữ liệu Cityscapes***

Chúng tôi đánh giá cách tiếp cận của chúng tôi trên bộ dữ liệu phân đoạn cá thể của Cityscapes [6]. Bộ dữ liệu này có hình ảnh được chụp từ 27 thành phố ở Đức và các nước lân cận. Nó chứa 2975 hình đào tạo, 500 hình xác nhận và 1525 hình thử nghiệm. Vì chúng tôi không có các trường hợp thực tế trên bộ thử nghiệm, chúng tôi sử dụng phân tách thay thế, trong đó 500 hình ảnh xác thực ban đầu tạo thành bộ thử nghiệm của chúng tôi. Sau đó chúng tôi chia tập huấn luyện ban đầu và chọn hình ảnh từ hai thành phố (Weimar và Zurich) làm xác nhận của chúng tôi, trong khi các thành phố còn lại trở thành tập huấn luyện của chúng tôi. Bộ dữ liệu có chú thích cho tám loại đối tượng: người, người lái, xe hơi, xe tải, xe buýt, xe lửa, xe máy và xe đạp. Số lượng phiên bản cho mỗi lớp trong phân chia của chúng tôi được hiển thị trong Bảng 1. Bộ dữ liệu Cityscapes có các phiên bản có sự thay đổi lớn về kích thước của chúng. Chúng tôi hiển thị phân phối các thể hiện cho các độ dài khác nhau của cạnh dài nhất của hộp, trong Hình 3. Chúng tôi quan sát một phương sai lớn, từ 28 pixel đến 1792 pixel.

Cityscapes cung cấp ground truth phân đoạn cá thể cả về nhãn mác pixel cũng như về đa giác. Trước đây, mỗi pixel có thể tương ứng với tối đa một trường hợp, do đó đại diện cho phần hiển thị của đối tượng. Tuy nhiên, đa giác Cityscapes thông thường cũng chụp một số phần bị chặn của một thể hiện, vì công cụ chú thích đã thực hiện thứ tự độ sâu của các đối tượng để di chuyển lại các phần bị chặn một cách hiệu quả [6]. Chúng tôi xử lý các đa giác để tạo lại hiệu ứng phân lớp và thu được các đa giác chỉ đại diện cho các phần có thể nhìn thấy của mỗi đối tượng. Số lượng đỉnh trung bình từ các đa giác kết quả được hiển thị trong Bảng 2. Vì các đối tượng có thể được chia thành nhiều thành phần do tắc, nên các thống kê theo thành phần coi mỗi thành phần là một ví dụ duy nhất, trong khi thống kê khôn ngoan ví dụ coi toàn bộ ví dụ là một ví dụ. Dựa trên thống kê này, chúng tôi chọn giới hạn cứng là 70 bước thời gian cho RNN của chúng tôi, cũng tính đến các yêu cầu bộ nhớ GPU.

**Đánh giá số liệu:** Chúng tôi đo lường hai khía cạnh của các chú thích được chỉ định trước của chúng tôi. Để đánh giá chất lượng của chúng, chúng tôi sử dụng số liệu giao nhau (IoU), được tính toán trên cơ sở từng trường hợp và tính trung bình trên tất cả các trường hợp. Đây là một biện pháp nghiêm ngặt vì các đối tượng nhỏ bị phạt giống như các trường hợp lớn. Để đánh giá số lượng hành động của con người cần thiết để sửa các chú thích của chúng tôi, chúng tôi mô phỏng một chú thích sửa một điểm mỗi khi đỉnh dự đoán lệch khỏi đỉnh GT nhiều hơn một ngưỡng. Sau đó chúng tôi báo cáo số lần hiệu chỉnh (được đo bằng số lần nhấp).



* 1. **Chế độ dự đoán**

Trước tiên, chúng tôi sẽ kiểm tra hiệu suất của mô hình của chúng tôi bằng bất kỳ tương tác nào từ bộ chú thích, tức là, chúng tôi dự đoán đa giác đầy đủ tự động.Chúng tôi sẽ đề cập đến cài đặt này là chế độ dự đoán.

**Baselines:** Chúng tôi sử dụng DeepMask [20] và SharpMask [21] được đề xuất gần đây làm đường cơ sở tiên tiến nhất. Đưa ra một bản vá hình ảnh đầu vào, DeepMask sử dụng CNN để xuất nhãn nhãn pixel của một đối tượng và thực hiện điều đó không rõ ràng đối với lớp. Sharpmask mở rộng Deepmask bằng cách lấy mẫu thông minh đầu ra để có được nhãn ở độ phân giải cao hơn nhiều (160 so với 56). Lưu ý rằng theo cách tiếp cận ban đầu của họ, [20, 21] các bản vá mẫu hoàn toàn ở các tỷ lệ khác nhau trên toàn bộ hình ảnh. Ở đây, chúng tôi sử dụng các hộp ground-truth khi báo cáo hiệu suất cho phương pháp của họ. Hơn nữa, DeepMask và SharpMask sử dụng kiến ​​trúc ResNet [10] 50 lớp, đã được đào tạo trên bộ dữ liệu COCO [16]. Chúng tôi tinh chỉnh mạng này trên Cityscapes được chia thành hai bước. Trong bước đầu tiên, chúng tôi tinh chỉnh kiến ​​trúc ResNet chuyển tiếp nguồn cấp dữ liệu cho 150 epoch, tiếp theo là tinh chỉnh các trọng số cho các lớp nâng cấp của Sharpmask, trong 70 epoch. Quá trình hai bước này là trong tinh thần giống như đề xuất trong bài báo. Lưu ý rằng mặc dù hai cách tiếp cận này thực hiện tốt trong việc ghi nhãn pixel, nhưng đầu ra của chúng có thể dễ dàng bị thay đổi bởi một chú thích trong trường hợp xảy ra lỗi. Điều này trái ngược với cách tiếp cận của chúng tôi, trong đó tích hợp hiệu quả một con người trong vòng lặp để có được các chú thích chất lượng cao.

Chúng tôi sử dụng hai đường cơ sở bổ sung là SquareBox và Dilation10. SquareBox là một đường cơ sở đơn giản trong đó hộp đầy đủ được gắn nhãn là đối tượng. Thay vì lấy hộp vừa khít, chúng tôi giảm kích thước của hộp, giữ nguyên tỷ lệ khung hình. Dựa trên bộ xác thực, chúng tôi nhận được kết quả tốt nhất bằng cách chọn 80% hộp ban đầu. Nếu một cá thể có nhiều thành phần, chúng tôi sẽ lắp một hộp cho từng thành phần riêng lẻ thay vì sử dụng hộp đầy đủ. Đường cơ sở này bắt chước kịch bản, trong đó đối tượng được mô hình hóa đơn giản là một hộp chứ không phải là đa giác. Đối với đường cơ sở Dilation10, chúng tôi sử dụng kết quả phân đoạn từ [33], được đào tạo trên bộ dữ liệu phân đoạn Cityscapes. Đối với mỗi hộp giới hạn, chúng tôi coi các pixel thuộc danh mục đối tượng tương ứng làm mặt nạ thể hiện.

**Kết quả định lượng:** Chúng tôi báo cáo số liệu IoU trong Bảng 3. Chúng tôi vượt trội so với đường cơ sở ở 6 trên 8 loại, cũng như trung bình trên tất cả các lớp. Chúng tôi hoạt động đặc biệt tốt trong xe hơi, người và người lái, vượt trội so với Sharpmask lần lượt là 12%, 7% và 6%. Điều này đặc biệt ấn tượng vì Sharpmask sử dụng mạnh hơn Kiến trúc ResNet (chúng tôi sử dụng VGG).

**Ảnh hưởng của kích thước đối tượng:** Trong hình 4, chúng ta thấy mô hình của chúng ta thực hiện các đường cơ sở w.r.t trên các kích thước cá thể khác nhau. Đối với các trường hợp nhỏ, mô hình của chúng tôi thực hiện tốt hơn đáng kể so với đường cơ sở. Đối với các đối tượng lớn hơn, đường cơ sở có lợi thế do độ phân giải đầu ra lớn hơn. Hiệu ứng này là đáng chú ý nhất đối với các lớp như xe buýt và xe lửa, trong đó mô hình của chúng tôi có IOU thấp hơn so với đường cơ sở.

* 1. ***Chú thích trong vòng lặp***

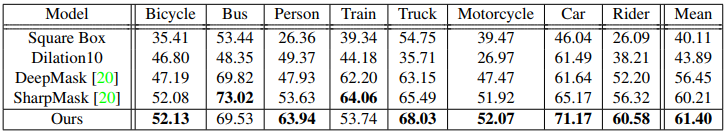
Ưu điểm chính của mô hình của chúng tôi là nó cho phép người chú thích dễ dàng can thiệp nếu xảy ra lỗi. Đặc biệt, ở mỗi bước thời gian RNN, bộ chú thích có khả năng sở hữu để điều chỉnh một đỉnh bị đặt sai vị trí. Việc hiệu chỉnh được đưa đến mô hình ở bước tiếp theo thay thế mô hình dự đoán mô hình, giúp mô hình trở lại đúng đường. Mục tiêu của chúng tôi là có được các chú thích chất lượng cao trong khi giảm thiểu thời gian chú thích.

Chúng tôi phân tích có bao nhiêu lần nhấp là cần thiết để có được mức độ chính xác của phân khúc. Chúng tôi thực hiện phân tích như vậy bằng cách mô phỏng một chú thích: chúng tôi sửa một dự đoán nếu nó lệch khỏi đỉnh sự thật mặt đất theo một khoảng cách nhất định. Khoảng cách được tính toán ở độ phân giải đầu ra mô hình bằng cách sử dụng số liệu bàn cờ. Trong các thử nghiệm của chúng tôi, chúng tôi so sánh các dự đoán đã sửa bằng các ngưỡng khoảng cách T [1; 2; 3; 4]. Trong Bảng 4, chúng tôi hiển thị kết quả IoU đưa ra các ngưỡng khác nhau về khoảng cách. Chúng ta có thể quan sát sự đánh đổi giữa số lần hiệu chỉnh và các số liệu này.

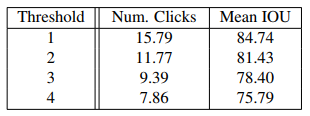
để so sánh kết quả với những thứ khác, chúng tôi đã thuê một người chú thích có chất lượng cao. Chúng tôi đã yêu cầu người chú thích chú thích tất cả các trường hợp xe (bao gồm cả xe tải) trong 10 hình ảnh Cityscapes được chọn ngẫu nhiên từ phần tách xác thực của chúng tôi. Chúng tôi thực hiện hai thí nghiệm: trong thử nghiệm đầu tiên, người chú thích được yêu cầu chú thích các đối tượng bằng cách xem miễn phí toàn bộ hình ảnh.Trong thử nghiệm thứ hai, chúng tôi cắt các bản vá hình ảnh bằng cách sử dụng các hộp Cityscapes và đặt một dấu chấm màu xanh trên ví dụ để định hướng chú thích. Chúng tôi lấy một bản cắt với 15% bối cảnh xung quanh hộp và mở rộng nó thành kích thước 224x224. Người chú thích đã sử dụng công cụ LabelMe [26] để chú thích.

Trong Bảng 5, chúng tôi báo cáo IoU đạt được bởi bộ chú thích con người cũng như số lần nhấp trung bình trên mỗi phiên bản trong mỗi thử nghiệm. Chúng ta có thể quan sát rằng thỏa thuận đạt được trong IoU là 69,5% trong chế độ xem miễn phí và 78,60% khi hiển thị các loại cây trồng (chế độ của chúng tôi). Con số này làm sáng tỏ những gì chúng ta mong đợi từ các phương thức tự động nói chung và chỉ ra một số sự mơ hồ trong nhiệm vụ. Nó cũng chỉ ra rằng điểm chuẩn sẽ thu thập nhiều chú thích của hình ảnh để giảm các biến thể và thành kiến trên các chú thích. Chúng tôi hy vọng phương pháp của chúng tôi sẽ làm cho việc thu thập dữ liệu đó khả thi và giá cả phải chăng.

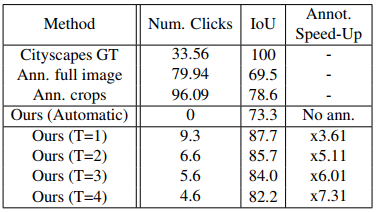
Lưu ý rằng mô hình của chúng tôi đạt được thỏa thuận cao hơn (82%) bằng cách chỉ yêu cầu trung bình 4.6 lần nhấp, đây là một yếu tố của 7.3 tốc độ tăng tốc trong thời gian chú thích. Ngay cả khi đạt được thỏa thuận cao tới 87.7, hệ số tăng tốc chú thích vẫn là 3.6. Điều này cho thấy hiệu quả của mô hình của chúng tôi như là một công cụ chú thích. Đối với tất cả các danh mục trong Cityscapes và theo cùng một quy trình, chúng tôi chỉ cần trung bình 9.39 lần nhấp để đạt được thỏa thuận 78.40% IoU, đạt được hệ số tăng tốc 4.74.



Bảng 3: Hiệu suất (IoU tính theo%) trên tất cả các lớp Cityscapes mà không có chú thích trong vòng lặp.

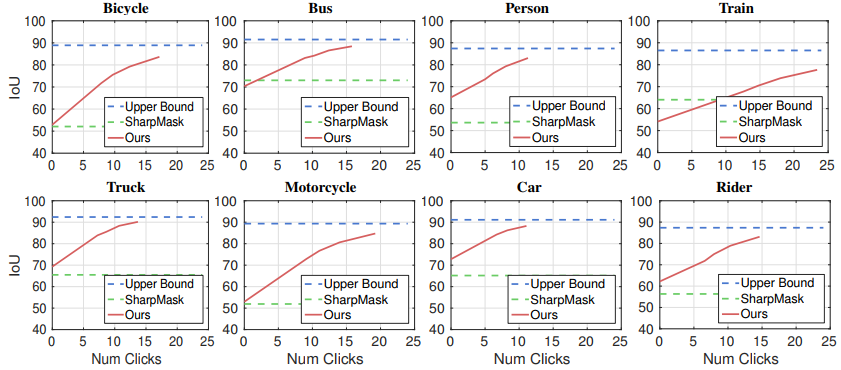


Bảng 4: Chú thích trong vòng lặp: Số lần sửa trung bình cho mỗi phiên bản và IoU, được tính trên tất cả các lớp. Ngưỡng cho biết khoảng cách bàn cờ đến đỉnh GT gần nhất.



Bảng 5: Mô hình của chúng tôi và Thỏa thuận chú thích: Chúng tôi đã thuê một người chú thích được đào tạo chuyên sâu để dán nhãn cho các trường hợp xe hơi trên 10 hình ảnh bổ sung (101 trường hợp). Chúng tôi báo cáo thỏa thuận IoU với Cityscapes GT và báo cáo thống kê đa giác. Chúng tôi so sánh cách tiếp cận của chúng tôi với thỏa thuận giữa các chú thích con người.

**So sánh với Grabcut.** Chúng tôi cũng so sánh hiệu suất của phương pháp của chúng tôi với một phương pháp bán tự động khác trên một bộ gồm 54 trường hợp được chọn ngẫu nhiên. Chúng tôi đã sử dụng triển khai OpenCV của Grabcut [25] cho thử nghiệm này. Trung bình, bằng cách sử dụng Grabcut, các trình chú thích cần 42.2 giây và 17.5 lần nhấp mỗi lần và đạt được mức trung bình 70.7% thỏa thuận IoU với Cityscapes GT. Trên cùng một bộ hình ảnh, mô hình của chúng tôi đạt được IoU từ 79.7% đến 85.8%, với 5.0 lần nhấp (T = 4) đến 9.6 lần nhấp (T = 1), tương ứng. Chuyên gia chú thích con người của chúng tôi cần 87,6 lần nhấp để có được IoU là 77,6% (không sử dụng bất kỳ công cụ tự động nào). Vì mô hình của chúng tôi yêu cầu sự can thiệp của con người ít hơn nhiều so với [25] (5 so với 17,5 lần nhấp) và yêu cầu thời gian suy luận tương đương cho mỗi lần nhấp, chúng tôi hy vọng rằng trong kịch bản trong thế giới thực, phương pháp của chúng tôi sẽ nhanh hơn nhiều.



Hình 5: Chú thích trong vòng lặp: Chúng tôi hiển thị IoU là một hàm của số lần nhấp / hiệu chỉnh.

**Kết quả định tính:** Trong hình 6, chúng tôi hiển thị các ví dụ về hình ảnh được chú thích bằng phương pháp của chúng tôi. We remind the reader, that this labeling is obtained by exploiting the GT bounding boxes. In particular, we here show the predictions obtained without any corrections (0 clicks). Mô hình của chúng tôi có thể phân chia chính xác các thể hiện với nhiều hình dạng và kích cỡ khác nhau.Đối với các trường hợp lớn, lỗi lượng tử hóa được giới thiệu bởi độ phân giải đầu ra của mô hình của chúng tôi trở nên rõ ràng. Tăng độ phân giải đầu ra là chủ đề của công việc đang diễn ra.Những thách thức chính là cân nhắc bộ nhớ cũng như những thách thức do chuỗi dài hơn (đa giác có nhiều đỉnh) mà mạng sẽ cần dự đoán.

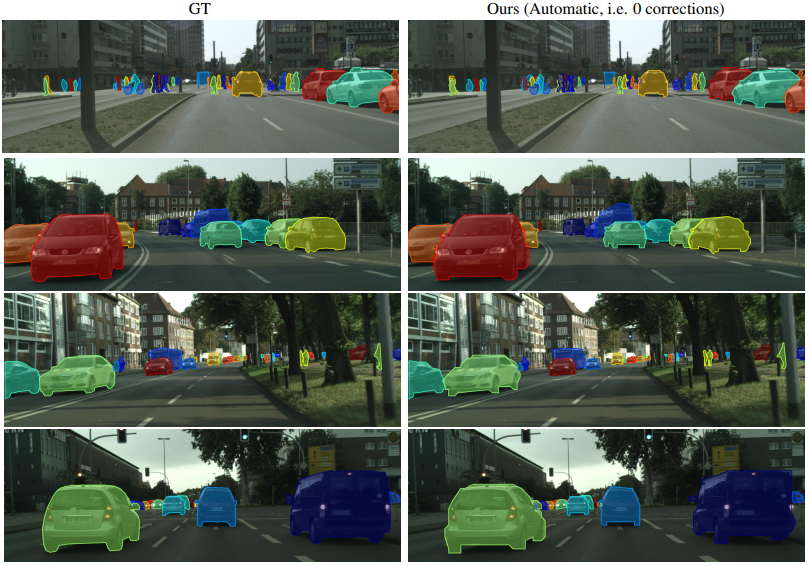
Trong hình 7, chúng tôi so sánh các chú thích của các ví dụ gần hơn bằng cách phóng to từng đối tượng.Chúng tôi có thể kiểm tra thỏa thuận giữa chú thích GT và chú thích trong nhà của chúng tôi, cũng như chất lượng của các dự đoán mà PolygonRNN thu được có và không có sửa chữa.

* 1. ***Chú thích các trường hợp KITTI***

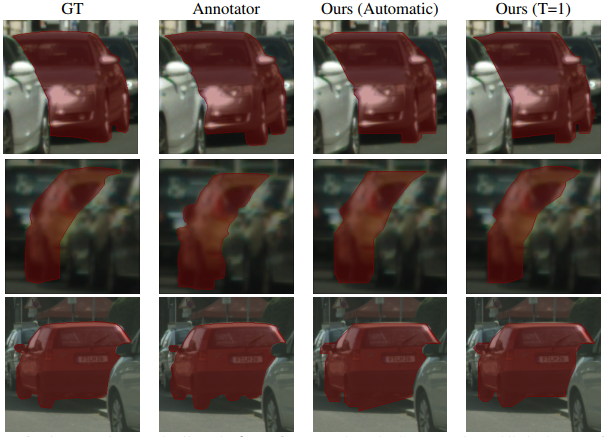
Chúng tôi cũng đánh giá mô hình của chúng tôi đã được đào tạo về Cityscapes tổng quát hóa như thế nào đến một bộ dữ liệu không nhìn thấy. Chúng tôi sử dụng KITTI cho thí nghiệm này, có 741 trường hợp chú thích được cung cấp bởi [4]. Chúng tôi báo cáo kết quả trong Bảng 6. Các thể hiện đối tượng trong KITTI thường lớn hơn các trường hợp được tìm thấy trong Cityscapes, khiến Deepmask và SharpMask hoạt động rất giống nhau. Lưu ý rằng [4], cũng là một phương pháp cho chú thích bán tự động, đã khai thác các đám mây điểm Velodyne để thực hiện ghi nhãn của chúng, điều này đặt nó với một mánh khóe không công bằng. Mô hình của chúng tôi bị phạt thêm bởi đầu ra độ phân giải thấp hơn. Tuy nhiên, hiệu suất của họ thấp hơn so với phương pháp hoàn toàn tự động của chúng tôi. Chỉ với 5.84 lần nhấp vào trung bình mỗi trường hợp, các mô hình của chúng tôi đạt được IOU tương đương với thỏa thuận ký hiệu của con người, do đó giảm chi phí chú thích.

1. **Kết luận**

Trong bài báo này, chúng tôi đã đề xuất một cách tiếp cận để tạo điều kiện chú thích cho các trường hợp đối tượng. Đa giác-RNN của chúng tôi dự đoán một đa giác phác thảo một đối tượng và dễ dàng kết hợp các hiệu chỉnh từ một chú thích trong vòng lặp. Chúng tôi hiển thị tăng tốc chú thích của yếu tố 4.74 trong khi đạt được thỏa thuận chú thích giống như giữa các chú thích con người. Ưu điểm chính của phương pháp của chúng tôi là nó tạo ra các chú thích hợp lý về mặt cấu trúc của các đối tượng và cho phép chúng tôi đạt được độ chính xác chú thích mong muốn bằng cách chỉ cần một vài lần nhấp của chú thích. Các thử nghiệm bổ sung cho thấy phương pháp của chúng tôi tổng quát hóa trên các bộ dữ liệu khác nhau, do đó thể hiện sức mạnh của nó như một công cụ chú thích chung.



Hình 6: Kết quả định tính trong chế độ dự đoán: Chúng tôi hiển thị đa giác cho tất cả các lớp trong ảnh gốc. Lưu ý rằng phương pháp của chúng tôi sử dụng hộp GT làm đầu vào. (trái) chúng tôi hiển thị nhãn GT của hình ảnh, (phải) chúng tôi hiển thị đa giác của chúng tôi mà không có sự can thiệp của con người. Các hình ảnh GT chứa 38, 12, 28 và 16 trường hợp và yêu cầu lần lượt 985, 308, 580 và 338 từ các chú thích Cityscapes của chúng



Hình 7: Chúng tôi xem xét một vài trường hợp chi tiết hơn. Trong cột đầu tiên, chúng tôi hiển thị chú thích GT, trong khi ở cột thứ hai, chúng tôi hiển thị các đa giác từ chú thích trong nhà. Chúng tôi quan sát rằng các phân khúc này có chất lượng cao nhưng khác nhau ở các khu vực không chắc chắn như cơ sở của chiếc xe.Trong cột thứ ba, chúng tôi hiển thị dự đoán PolygonRNN mà không cần sự can thiệp của con người. Cuối cùng, trong cột thứ tư, chúng tôi hiển thị một dự đoán chính xác. Chúng ta có thể quan sát rằng phân khúc được tinh chỉnh để phác thảo tốt hơn gương xe hoặc bánh xe.

.