**Abstract**

Trong bài báo này, chúng tôi giới thiệu một **kiến trúc mã hóa-giải mã (encoder-decoder)** khai thác cả **ngữ nghĩa toàn cục và cục bộ** để giải quyết bài toán **tự động tô màu ảnh**. Đối với ngữ nghĩa toàn cục, các đặc trưng mã hóa cấp thấp được tinh chỉnh bằng cách phân loại ngữ cảnh cảnh quan để tích hợp phong cách tổng thể của hình ảnh. Hơn nữa, kiến trúc này xử lý sự không chắc chắn và mối quan hệ giữa các phong cách cảnh quan dựa trên kỹ thuật làm mịn nhãn (label smoothing) và trọng số được huấn luyện trước từ Places365.

Đối với ngữ nghĩa cục bộ, ba nhánh học cách tận dụng lợi ích chung ở cấp độ điểm ảnh. Cụ thể, các phân phối trung bình và đa mô hình lần lượt được tạo ra từ nhánh hồi quy và mã hóa mềm (soft-encoding), trong khi nhánh phân đoạn xác định điểm ảnh thuộc về đối tượng nào.

Các thí nghiệm của chúng tôi, sử dụng bộ dữ liệu **Coco-Stuff** để huấn luyện và đánh giá trên DIV2K, Places365 và ImageNet, cho thấy kết quả thu được rất khả quan.

INTRODUCTION

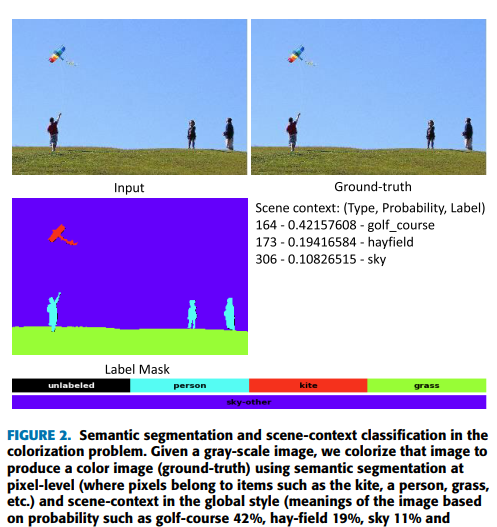
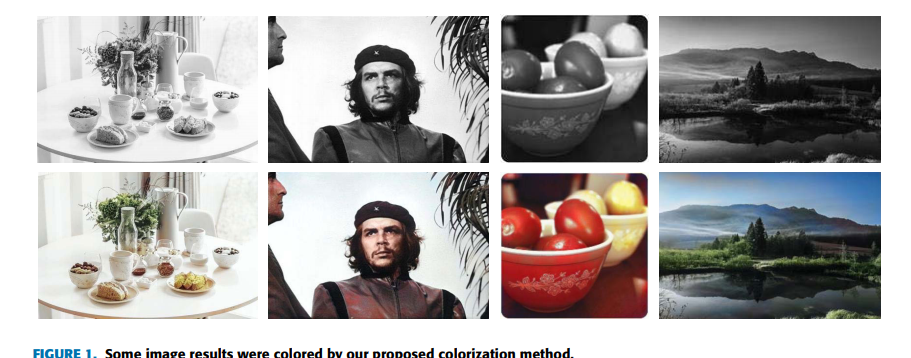
Việc tô màu một hình ảnh xám không chỉ mang lại nhiều ý nghĩa đặc biệt cho hình ảnh đó mà còn giúp nó trở nên sống động và có cảm xúc hơn [1]. Một hình ảnh thường chứa nhiều đối tượng, và con người có thể dễ dàng quyết định màu sắc phù hợp cho từng đối tượng trong ảnh nhờ vào kiến thức về ý nghĩa của đối tượng. Tuy nhiên, hầu hết các đối tượng không chỉ có một màu duy nhất; ví dụ, một chiếc áo có thể có màu đỏ, xanh, vàng hoặc nhiều màu khác. Con người cũng dự đoán màu sắc của đối tượng dựa trên cảm xúc chủ quan của mình, đây là một trong những thách thức lớn nhất đối với một hệ thống máy tính khi cố gắng bắt chước cả kiến thức lẫn cảm nhận của con người.

Vì lý do này, các hệ thống tô màu trước đây thường yêu cầu sự hợp tác với người dùng, chẳng hạn như cung cấp thông tin ban đầu bằng cách nhập các điểm màu [2] hoặc chuyển màu từ các hình ảnh xám tham chiếu. Tuy nhiên, việc chọn một hình ảnh chứa các chi tiết đối tượng chính xác mất quá nhiều thời gian, cũng như việc vẽ các điểm màu.

Do đó, phương pháp tô màu tự động dựa trên mạng nơ-ron tích chập sâu (CNN) mà không cần sự can thiệp của con người đã được đề xuất [3]–[6]. Nhờ vào những lợi ích to lớn của hệ thống tô màu tự động, các nhà nghiên cứu đã tìm ra nhiều cách để giúp máy tính hiểu thông tin ngữ nghĩa của hình ảnh trong quá trình xử lý nhằm đạt được kết quả tô màu hiệu quả.

Một phương pháp điển hình là nghiên cứu của [7], trong đó sử dụng phân loại cảnh từ cơ sở dữ liệu Places365 để huấn luyện mô hình về ngữ nghĩa toàn cục của hình ảnh. Hơn nữa, nghiên cứu [8] tập trung vào cả bố cục ngữ nghĩa và các đối tượng cục bộ của cảnh để tô màu ảnh bằng kiến trúc mạng VGG-16. Trong phương pháp của Zhang et al. [9], họ bổ sung khả năng tổng hợp ngữ nghĩa cấp đối tượng bằng cách mã hóa liên kênh vào mô hình của mình.

Trong các nghiên cứu trên, ngữ nghĩa chỉ được khai thác ở cấp độ hình ảnh. Ngược lại, các nghiên cứu hiện tại về phân đoạn ngữ nghĩa đã đạt đến cấp độ điểm ảnh bằng cách gán nhãn cho từng điểm ảnh [10]. Theo cách tương tự, tô màu cũng là một nhiệm vụ liên quan đến việc gán màu cho từng điểm ảnh dựa trên một phân phối xác suất màu sắc [11]. Để cải thiện phương pháp tô màu bằng cách kết hợp các khái niệm này, ngoài ý nghĩa cảnh quan, chúng tôi nhận thấy rằng nó một cách hợp lý hơn.



Nhiệm vụ phân đoạn ngữ nghĩa đóng vai trò quan trọng trong việc cung cấp thông tin ngữ nghĩa ở cấp độ điểm ảnh.

Kinh nghiệm sống của chúng ta có vai trò thiết yếu trong việc hình thành mô hình nhận thức thông tin. Thông tin, bao gồm cả loại toàn cục và cục bộ, được lưu trữ trong tiềm thức và dần dần hình thành kiến thức theo thời gian. Khi nhìn vào một hình ảnh, chúng ta có thể phân tích loại cảnh, xác định các đối tượng có thể xuất hiện, chất liệu của chúng, và nhiều yếu tố khác. Kiến thức về màu sắc là một dạng thông tin mà con người tích lũy trong suốt cuộc đời. Do đó, việc tô màu cho một bức ảnh xám là một nhiệm vụ dễ dàng đối với con người.

Tuy nhiên, có nhiều thách thức khi để một máy tính tự động chọn màu sắc chính xác để tô màu cho ảnh đen trắng. Máy tính chỉ có thể tính toán dự đoán dựa trên mối quan hệ giữa ánh sáng và màu sắc mà không xét đến ngữ nghĩa của đối tượng, điều này có thể dẫn đến sự nhầm lẫn về màu sắc. Ngoài ra, do các hình ảnh đầu vào có nhiều kích thước khác nhau, không thể tránh khỏi sự thay đổi tỷ lệ, gây ra hiện tượng lem màu giữa các vùng biên và nhiễu màu.

Một ví dụ về quá trình này được minh họa trong Hình 2, trong đó người xem có thể quan sát một hình ảnh xám và dễ dàng suy ra một số thông tin tổng quan, chẳng hạn như đây là một cảnh ngoài trời có bầu trời và cánh đồng—đây được gọi là thông tin ngữ nghĩa cảnh quan hay thông tin toàn cục. Bên cạnh đó, các đối tượng như con người, diều, cỏ và bầu trời được coi là thông tin cục bộ và có thể được xác định thông qua phân đoạn ngữ nghĩa.

Sự kết hợp giữa kiến thức toàn cục (ngữ nghĩa cảnh quan) và kiến thức cục bộ (phân đoạn ngữ nghĩa) sẽ giúp người quan sát xác định đối tượng trong hình ảnh và tham chiếu mô hình đối tượng để xác định màu sắc phù hợp. Vì lý do này, chúng tôi đề xuất một giải pháp sử dụng cả phân đoạn và ngữ nghĩa cảnh quan để tự động tô màu hình ảnh, như minh họa trong Hình 1.

Ngoài ra, chúng tôi cũng quan tâm đến việc lựa chọn một mạng phù hợp để thúc đẩy sự kết hợp giữa các yếu tố này. Trong bài báo của mình, chúng tôi sử dụng kiến trúc encoder-decoder, như được trình bày trong Hình 3, do tính linh hoạt của nó trong học đa nhiệm (multi-task learning). Các đặc trưng mã hóa cấp thấp sẽ được tinh chỉnh thông qua nhánh phân loại ngữ cảnh cảnh quan, nằm ở giữa hai nhánh mã hóa (encoder) và giải mã (decoder). Điều này có nghĩa là chúng không chỉ chứa dữ liệu mã hóa của hình ảnh đầu vào mà còn tích hợp phong cách tổng thể của hình ảnh, giúp nhánh giải mã thực hiện tô màu một cách hợp lý hơn.

Đề xuất một kiến trúc encoder-decoder sử dụng phân loại ngữ cảnh cảnh quan và phân đoạn cấp điểm ảnh để tô màu hình ảnh.

Hình ảnh sẽ được tô màu chính xác hơn dựa trên ngữ nghĩa toàn cục của ngữ cảnh cảnh quan. Đối với ngữ nghĩa cấp điểm ảnh, đầu ra của nhánh giải mã (decoder) dưới dạng các đặc trưng ánh xạ điểm ảnh 2D sẽ được tinh chỉnh thông qua ba nhánh: phân đoạn, hồi quy (regression), và mã hóa mềm (soft-encoding).

**Ba nhánh này học cách tận dụng lợi ích chung, trong đó:**

**Phân đoạn** giúp xác định ý nghĩa của từng điểm ảnh,  
 **Hồi quy** tạo mô hình màu trung bình,

**Mã hóa mềm** sử dụng mô hình màu có độ lệch để xây dựng các đặc trưng ánh xạ điểm ảnh 2D mạnh mẽ hơn.

Trong quá trình triển khai, chúng tôi sử dụng mô hình U-Net [12] với cơ chế kết nối bỏ qua (skip connection) để giúp mô hình hội tụ tốt hơn và tránh vấn đề mất dần thông tin (vanishing problem) do kích hoạt chết và độ sâu mạng quá lớn.

Đối với ngữ nghĩa ngữ cảnh cảnh quan, chúng tôi xử lý sự không chắc chắn và mối quan hệ giữa các cảnh bằng kỹ thuật làm mịn nhãn (label smoothing), sử dụng xác suất tiên nghiệm từ trọng số được huấn luyện trước trong Places365.

Đối với ngữ nghĩa cấp điểm ảnh, chúng tôi tích hợp học lợi ích chung từ mô hình phân đoạn cấp điểm ảnh, mô hình màu trung bình và mô hình màu có độ lệch.

Chúng tôi huấn luyện kiến trúc này trên bộ dữ liệu Coco-Stuff [13] và kiểm tra trên DIV2K [14], Places365 [15], ImageNet [16], và các hình ảnh thu thập từ Internet để đảm bảo kết quả đạt chất lượng tốt.

Phần còn lại của bài báo bao gồm năm phần:

**Phần 2**: Trình bày các nghiên cứu trước liên quan đến bài toán tô màu ảnh.

**Phần 3**: Mô tả kiến trúc mạng, quy trình xử lý dữ liệu đầu vào, đầu ra và thông tin ngữ nghĩa, cũng như hàm mất mát đa nhiệm.

**Phần 4**: Mô tả chi tiết quá trình triển khai phương pháp đề xuất.

**Phần 5**: Trình bày chi tiết các thí nghiệm và thảo luận kết quả.

**Phần 6**: Kết luận bài báo.

**RELATED WORKS**

Việc tô màu cho những bức ảnh đen trắng cũ nhằm tái tạo lại các cảnh lịch sử có thể giúp lưu giữ nhiều giá trị văn hóa và kỷ niệm trước sự tàn phá của thời gian, đồng thời mang những người hiện đại đến gần hơn với quá khứ. Tô màu ảnh vẫn là một trong những chủ đề hấp dẫn đối với người dùng phần mềm chỉnh sửa ảnh, điều này được thể hiện qua nhiều hướng dẫn tô màu có trên Internet. Tuy nhiên, tô màu bằng phần mềm chỉnh sửa ảnh rất tốn kém và mất nhiều thời gian. Một nghệ sĩ kỹ thuật số phải bỏ ra nhiều công sức để tô màu ảnh đen trắng, bắt đầu bằng việc tạo nhiều lớp cho các vùng khác nhau của hình ảnh, sau đó gán và điều chỉnh màu sắc cho các vùng này sao cho phù hợp với kiến thức và trí tưởng tượng của họ.

Các nhà nghiên cứu máy tính đã phát triển nhiều ứng dụng tô màu nhằm giải quyết những khó khăn khác nhau liên quan đến quá trình tô màu này.

Các nghiên cứu trước đây thường đề cập đến ba phương pháp tô màu truyền thống. Loại thứ nhất là dựa trên phác thảo màu (scribble-based), trong đó người dùng chú thích màu lên ảnh; loại thứ hai là sử dụng màu từ ảnh tham chiếu để áp dụng vào ảnh mục tiêu; và loại cuối cùng là tô màu tự động dựa trên học sâu (deep learning).

**TÔ MÀU DỰA TRÊN PHÁC THẢO MÀU (Scribble-Based)**

Đây là phương pháp tô màu sử dụng các nét phác thảo màu ban đầu và tự động lan truyền màu sắc đến các pixel lân cận có màu tương tự. Levin et al. [2] đã đề xuất phương pháp này vào năm 2004 dựa trên giả thuyết rằng các pixel xung quanh có cùng độ sáng sẽ có màu tương tự. Tuy nhiên, kết quả của họ cho thấy tình trạng màu bị lan tràn giữa các đối tượng. Sau đó, phương pháp được mô tả trong [17] đã khắc phục tình trạng này bằng cách khai thác thông tin về cạnh ở mức cục bộ để ngăn màu bị lan qua ranh giới giữa các đối tượng. Một số phương pháp như [18] và [19] đã cải thiện hiệu quả bằng cách truyền màu không chỉ đến các pixel lân cận mà còn đến các pixel có cùng mẫu. Mặc dù hình ảnh được tô màu bằng các phương pháp này trông tự nhiên và có màu sắc chi tiết rõ ràng, nhưng chúng đòi hỏi nhiều thời gian và công sức.

**TÔ MÀU DỰA TRÊN ẢNH THAM CHIẾU (Example-Based)**

Thay vì vẽ nhiều nét phác thảo, phương pháp tô màu dựa trên ảnh tham chiếu liên quan đến việc chuyển màu từ các hình ảnh tương tự sang ảnh mục tiêu. Một số phương pháp này đo lường sự tương đồng giữa ảnh tham chiếu và ảnh mục tiêu ở mức pixel. Cụ thể, Charpiat et al. [3] đã chọn một số hình ảnh màu liên quan làm đầu vào, sau đó xây dựng xác suất phân bố của các màu hợp lý cho từng pixel ở mức cục bộ, **cuối cùng tối ưu hóa xác suất của ảnh tô màu bằng thuật toán cắt đồ thị (graph-cut).** Với cùng mục tiêu, Welsh et al. [20] đã chuyển màu từ ảnh tham chiếu sang cả ảnh tĩnh và video.

Khác với hai phương pháp trên, Liu et al. [4] đã chuẩn bị ảnh tham chiếu bằng cách tự động tìm kiếm các bức ảnh phù hợp trên Internet. Ở mức đoạn ảnh (segment level), Irony et al. [21] đã tính toán màu sắc trên từng đoạn của ảnh tham chiếu có thể được chuyển sang pixel của ảnh mục tiêu. Hơn nữa, phương pháp trong [22] đã thêm chú thích bằng tay vào quá trình phân đoạn để lọc ra các mẫu tham chiếu phù hợp.

Mặc dù các phương pháp này tạo ra kết quả đẹp và tự nhiên, nhưng chúng vẫn mất nhiều thời gian để tìm kiếm hình ảnh tham chiếu phù hợp, ngay cả khi sử dụng các công cụ tìm kiếm tự động trên web.

**TÔ MÀU DỰA TRÊN HỌC SÂU**

Gần đây, với sự phát triển của các ứng dụng thị giác máy tính, tô màu ảnh ngày càng đóng vai trò quan trọng và liên tục được cải tiến để đáp ứng nhu cầu của người dùng. Nó đã được áp dụng để hỗ trợ các nhiệm vụ nâng cao, chẳng hạn như nghiên cứu của [23], **sử dụng thuộc tính màu sắc để cải thiện hiệu suất phát hiện đối tượng**, và nghiên cứu của [24], **phát triển ứng dụng chỉnh sửa màu để tạo ra kết quả tối ưu hơn.**

Gần đây nhất, trong ngành công nghiệp truyện tranh, **các phương pháp tô màu đã được phát triển mạnh mẽ nhằm giảm đáng kể chi phí và công sức** [9], [25]–[27].

Để giảm bớt công việc thủ công của các phương pháp trước đó, quá trình tô màu đã tận dụng học sâu để học cách dự đoán màu sắc. Ví dụ, nghiên cứu của [6] đã đề xuất một kỹ thuật phân cụm hình ảnh cho một tập dữ liệu lớn và xây dựng một kiến trúc mạng nơ-ron sâu để đào tạo với các bộ mô tả đặc trưng của hình ảnh.

Gần đây, việc đào tạo mạng dựa trên ngữ nghĩa (semantics training) đã được khai thác để cải thiện hiệu suất tốt hơn. Ví dụ, Iizuka et al. [7] đã xây dựng hai nhánh mạng dựa trên mạng nơ-ron tích chập sâu (deep convolutional neural networks) để kết hợp các đặc trưng toàn cục và cục bộ.

Để hỗ trợ ngữ nghĩa cho mạng, Zhang et al. [11] đã sử dụng phương pháp mã hóa chéo kênh (cross-channel encoding), trong khi Larsson et al. [8] đã thiết kế một hệ thống có thể dự đoán biểu đồ màu (color histogram) cho từng pixel và được tiền huấn luyện (pre-trained) cho các nhiệm vụ phân loại. Các phương pháp này đã đạt được kết quả ấn tượng khi tô màu những hình ảnh phức tạp.

Tóm lại, có ba phương pháp chính để tô màu ảnh:

**Tô màu dựa trên phác thảo màu** – Người dùng vẽ lên ảnh dựa vào độ sáng để ánh xạ màu sắc tương tự.

Truyền màu từ các nét vẽ ban đầu đến các pixel lân cận có độ sáng tương tự [2].

Cải thiện bằng cách sử dụng thông tin cạnh để ngăn màu lan giữa các đối tượng [17].

Mở rộng bằng cách truyền màu không chỉ theo pixel lân cận mà còn theo các mẫu tương tự [18], [19].

**Tô màu dựa trên ảnh tham chiếu** – Chuyển màu từ ảnh tham chiếu có sẵn sang ảnh mục tiêu.

Truyền màu từ ảnh tham chiếu sang ảnh mục tiêu bằng cách đo độ tương đồng giữa chúng.

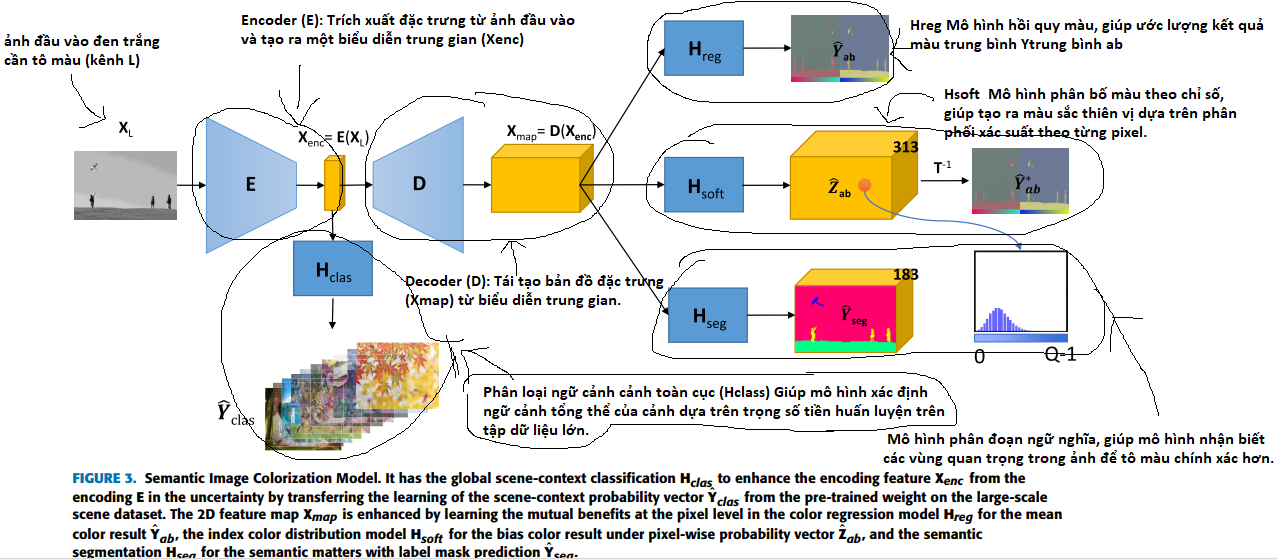
Phương pháp trong [3] sử dụng tối ưu hóa đồ thị (graph-cut) dựa trên xác suất phân phối màu của từng pixel cục bộ.

**Tô màu dựa trên học sâu** – Dùng mạng nơ-ron học cách tô màu tự động mà không cần sự can thiệp của con người.

Mạng hyper-column VGG với hàm mất mát chưa cân bằng [8].

Mạng hai luồng kết hợp đặc trưng toàn cục và cục bộ với phân loại ngữ cảnh cảnh quan [7].

Phương pháp mã hóa kênh chéo để học màu sắc hiệu quả hơn [11].

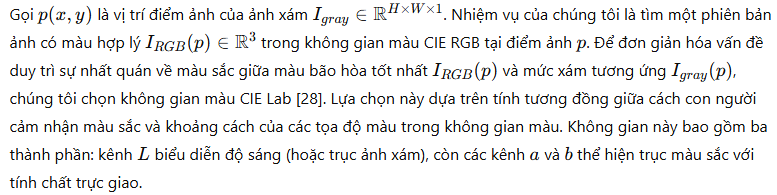


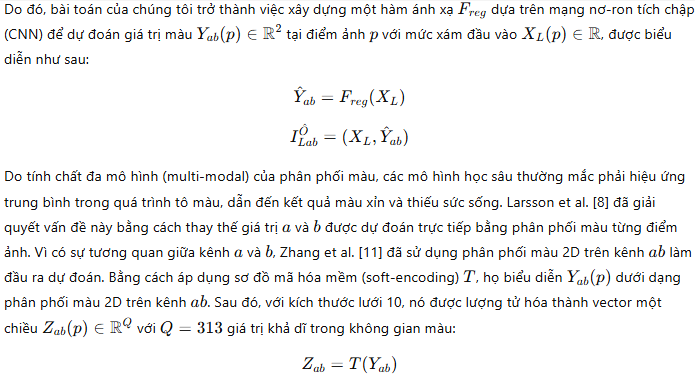
Kết quả cuối cùng là ảnh tô màu được tạo ra thông qua kết hợp các thông tin từ các mô-đun trên.( 3 cái H là 3 cái modun)

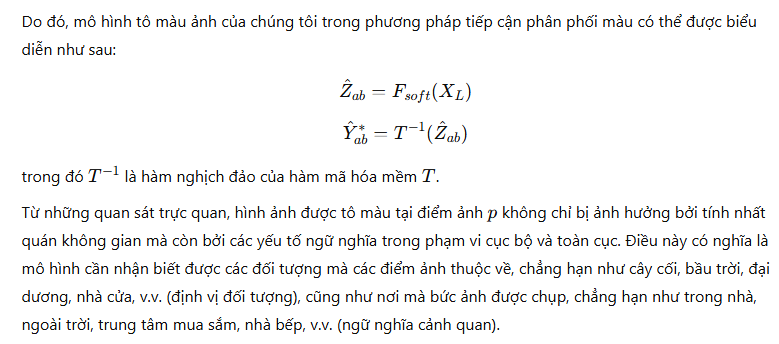
PROPOSED METHOD

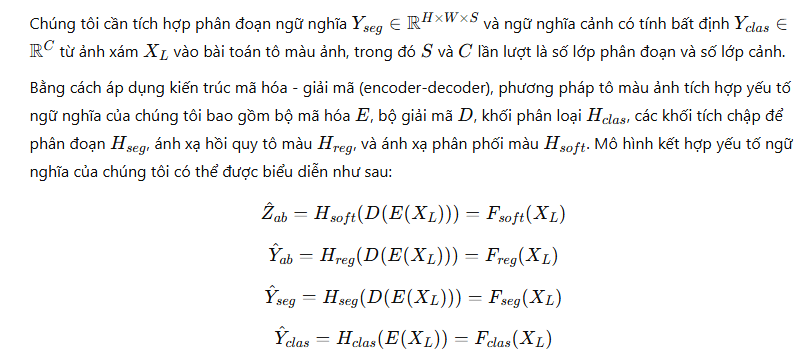
Trong phần này, chúng tôi mô tả mô hình đề xuất của mình, được minh họa trong Hình 3, nhằm giải quyết bài toán tô màu bằng cách sử dụng phân phối màu đa mô hình để lựa chọn màu thiên lệch (bias selection) và hồi quy màu để dự đoán màu trung bình. Đồng thời, mô hình cũng khai thác thông tin ngữ nghĩa bằng cách phân loại ngữ cảnh cảnh quan toàn cục và phân đoạn ngữ nghĩa ở cấp độ điểm ảnh. Sau đó, chúng tôi sẽ mô tả chi tiết phương pháp tô màu ngữ nghĩa, bao gồm: vector phân phối màu chỉ mục với cơ chế mã hóa và giải mã mềm (soft-encoding và decoding), dữ liệu ngữ cảnh cảnh quan có tính bất định, và phân đoạn ngữ nghĩa cấp độ điểm ảnh. Cuối cùng, chúng tôi đề cập đến hàm mất mát đa thành phần trong mô hình đề xuất.

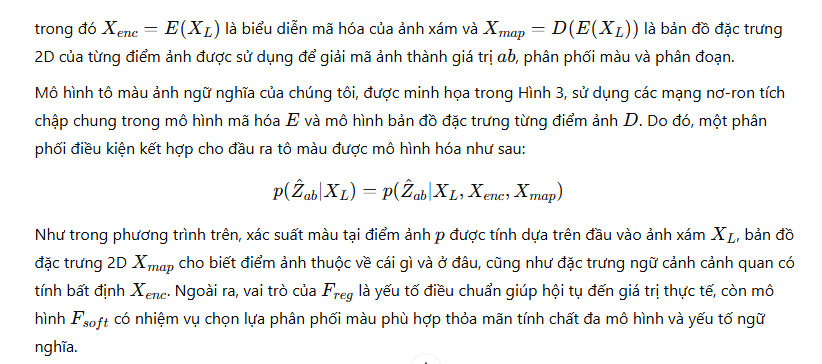
**MÔ HÌNH TÔ MÀU ẢNH NGỮ NGHĨA**

****

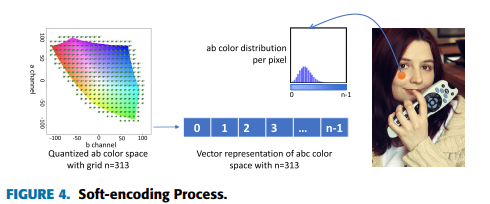
****

****

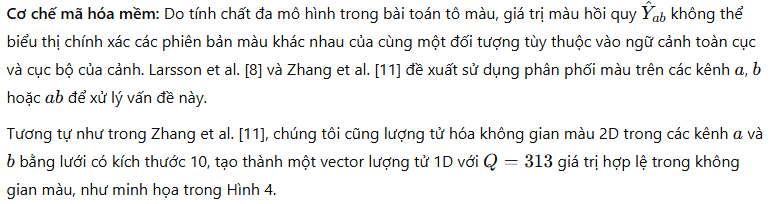
****

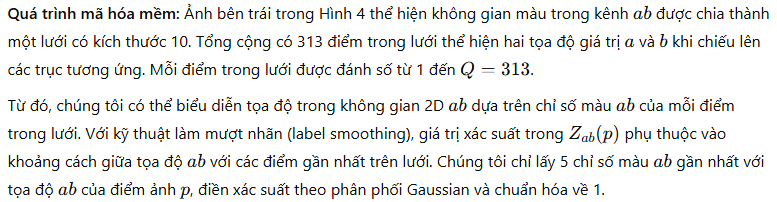


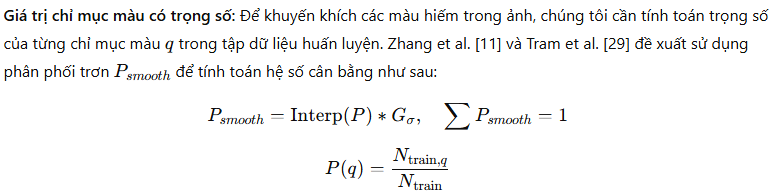
**DỮ LIỆU TÔ MÀU NGỮ NGHĨA**

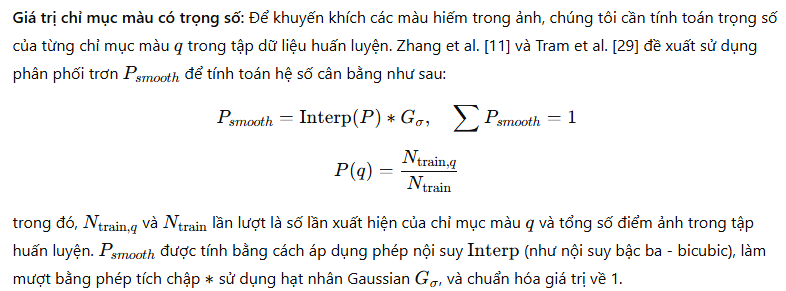


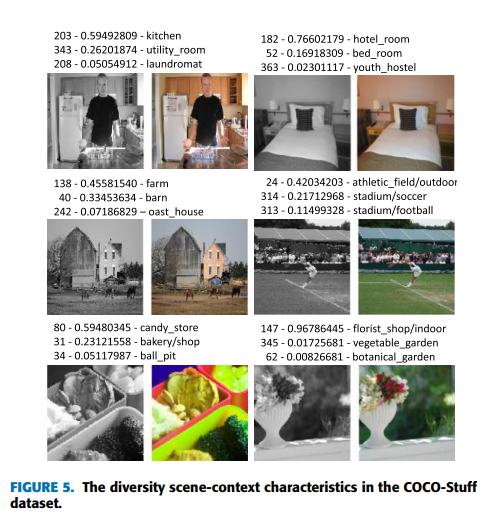
1) Dữ liệu phân phối màu với cơ chế mã hóa mềm

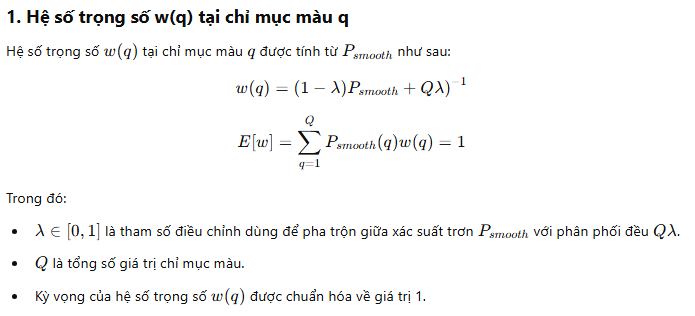


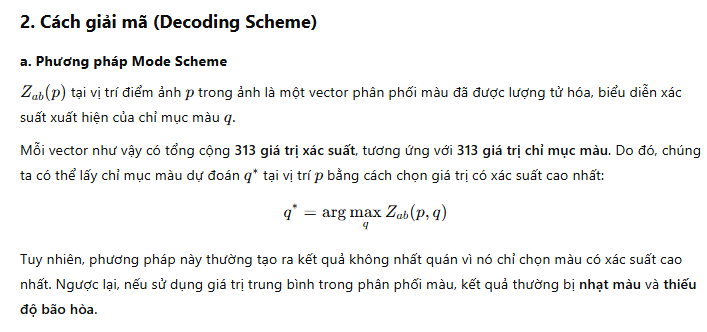


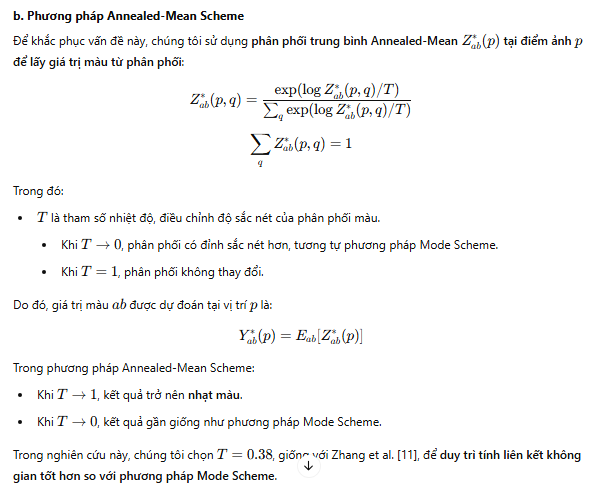


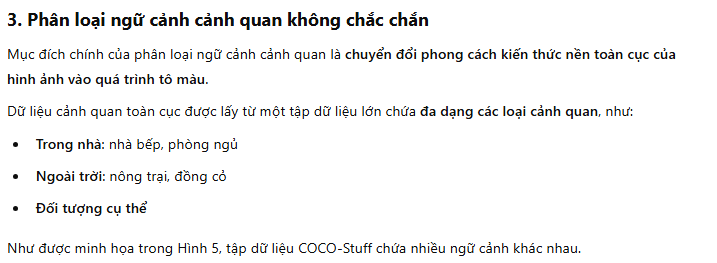


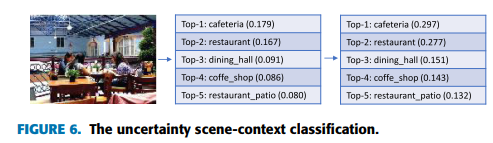


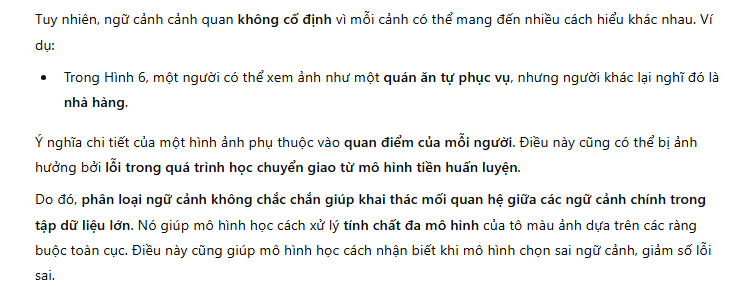


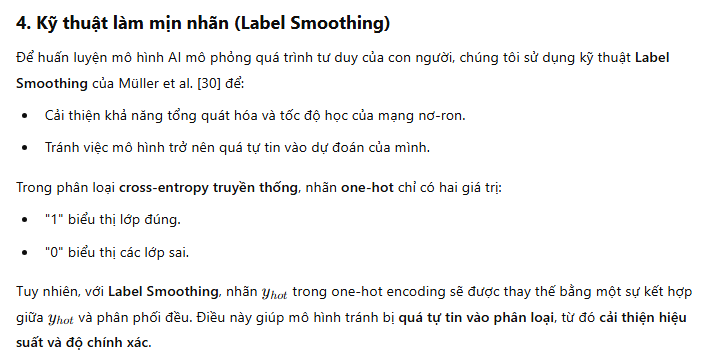






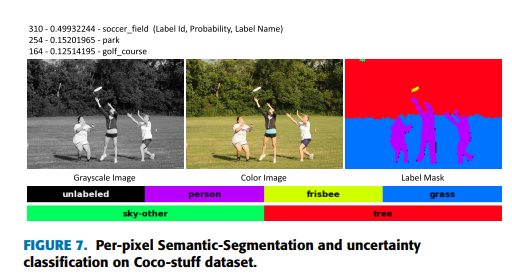






Trong bài báo này, thay vì sử dụng sự kết hợp giữa mã hóa one-hot và phân phối đồng đều như mô tả ở trên, chúng tôi điền vào năm xác suất cao nhất từ mô hình VGG16 đã được huấn luyện trước [31] trên tập Places [15] và chuẩn hóa các giá trị này thành 1, như thể hiện trong Hình 6. Điều này giúp chúng tôi chuyển giao kiến thức từ mạng miền cảnh (scene domain network) sang mô hình tô màu của mình trong điều kiện không chắc chắn, từ đó cải thiện khả năng tổng quát hóa của mô hình.

3) PHÂN ĐOẠN NGỮ NGHĨA Ở CẤP ĐỘ ĐIỂM ẢNH

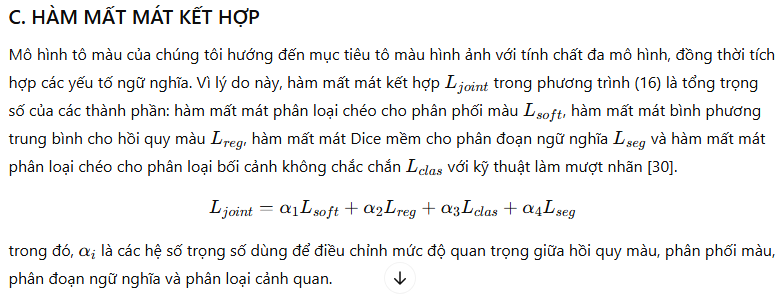


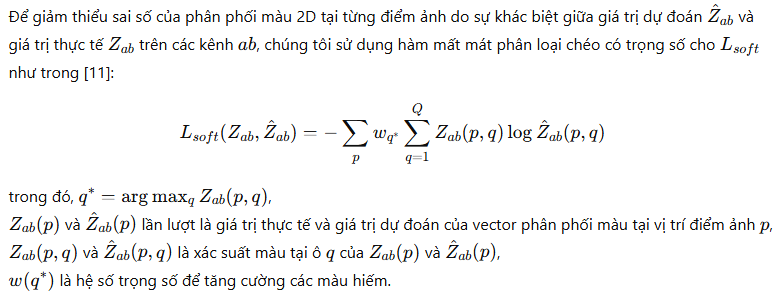
Ngoài các đặc trưng ngữ nghĩa tổng thể giúp chuyển đổi phong cách trong bối cảnh không chắc chắn, quá trình tô màu còn bị ảnh hưởng bởi yếu tố ngữ nghĩa trong phạm vi cục bộ ở cấp độ điểm ảnh. Nghĩa là mỗi điểm ảnh thuộc về một đối tượng hoặc một loại vật thể nhất định. Trong tầm nhìn của con người, chúng ta sử dụng kinh nghiệm với các đối tượng trong ảnh để tạo ra một hình ảnh được tô màu phù hợp với bối cảnh tổng thể của cảnh.

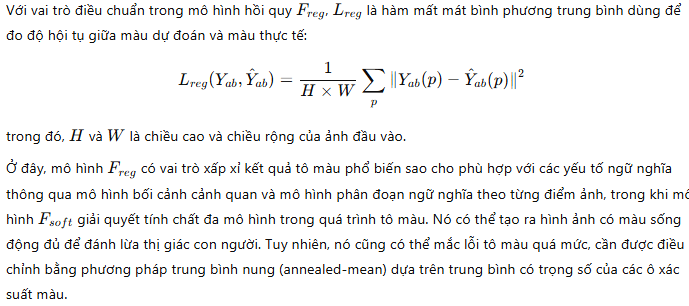
Trong Hình 7, với việc phân loại theo ngữ cảnh cảnh quan, bối cảnh không chắc chắn được tính toán từ trọng số đã được huấn luyện trước trên Places365 [15] cho một hình ảnh trong tập Coco-Stuff [13]. Hình ảnh này cho thấy ba bối cảnh có thể xảy ra với xác suất cao nhất. Sử dụng kỹ thuật làm mượt nhãn (label smoothing), chúng tôi ngăn chặn lỗi trong quá trình chuyển giao học tập và thiết lập mối quan hệ giữa các loại cảnh như sân bóng đá, công viên và sân golf.

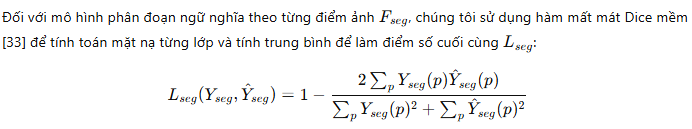
Ngoài ra, trong chi tiết hình ảnh, một mô hình đa phong cách được thể hiện với ba phong cách tô màu khác nhau cho nhãn "người". Đối với các nhãn "vật thể" (stuff), có các nhãn như "cây", "cỏ" và "bầu trời khác" ảnh hưởng đến ngữ cảnh cảnh quan. Cũng có một nhãn đối tượng nhỏ là "đĩa ném (frisbee)". Mặt nạ nhãn nội dung giúp mô hình học được mô tả cục bộ và tăng cường bản đồ đặc trưng giải mã XmapX\_{map}Xmap​ để nâng cao độ chính xác trong các cảnh phức tạp.

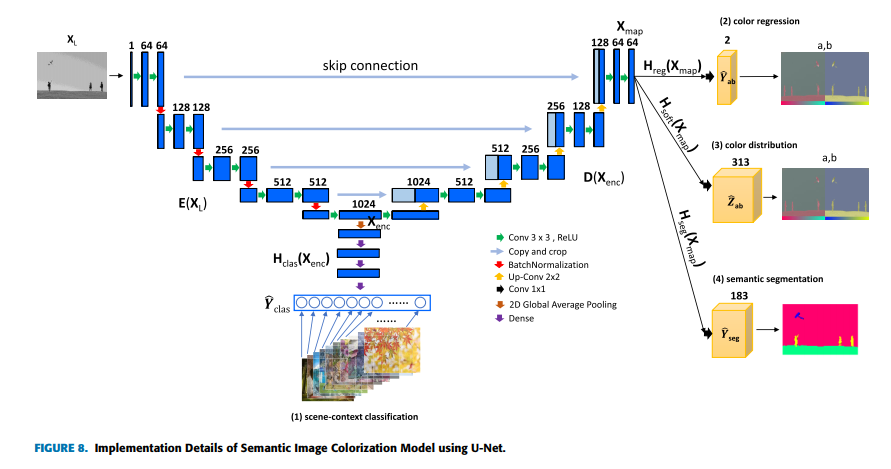
Trên thực tế, một hình ảnh có màu có thể được phân đoạn tốt hơn so với một hình ảnh xám [32]. Mặc dù hệ thống đầu vào của chúng tôi chỉ nhận ảnh xám, nhưng mô hình phân đoạn ngữ nghĩa tận dụng mô hình tô màu để cải thiện kết quả phân đoạn. Mặt khác, nó cũng giúp các mô hình khác khai thác các điểm ảnh theo ngữ nghĩa, xác định chúng thuộc về đâu và chúng là đối tượng hoặc vật thể gì.

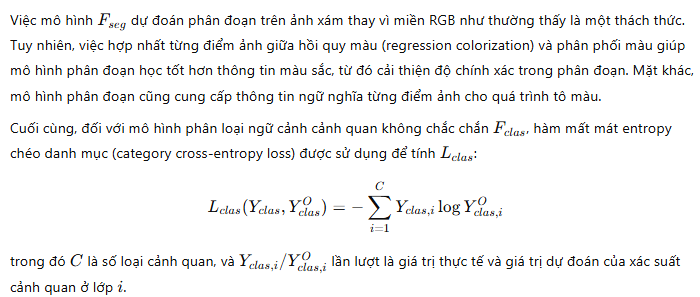


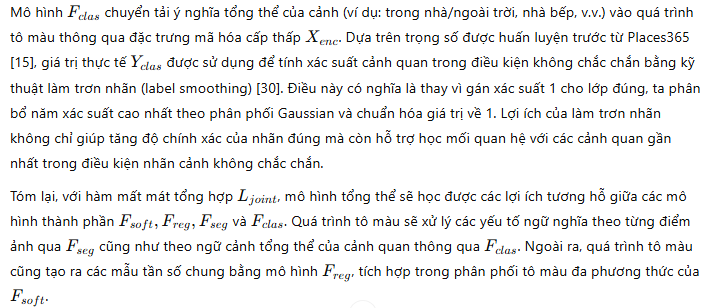












Trong phần này, chúng tôi mô tả chi tiết hơn về kiến trúc mạng của mình. Việc triển khai tự động tô màu hình ảnh dựa trên việc học từ từng pixel của cảnh-semantic và bối cảnh toàn cục của cảnh được mô tả trong Hình 8. **Đầu vào của mô hình là hình ảnh grayscale XL có kích thước H × W × 1**. **Các đầu ra của mô hình là phân loại bối cảnh cảnh YOclas có kích thước H × W × C, trong đó C = 365 là số lượng lớp phân loại cảnh, hồi quy màu sắc YOab trên kênh ab có kích thước H × W × 2( H x W x 2** là dữ liệu màu sắc trên kênh **a** và **b** trong không gian màu **Lab** Mô hình này sẽ dự đoán các giá trị màu sắc cho từng pixel dựa trên các kênh **a** và **b),** phân phối màu sắc ZOab có kích thước H × W × Q, trong đó Q = 313 là số lượng hộp trong hệ thống mã hóa mềm, và mặt nạ phân đoạn semantic YOseg có kích thước H × W × S, trong đó S = 183 là số lượng lớp phân đoạn.

**YOab**: Dự đoán phân phối màu sắc của các pixel trong ảnh, sử dụng hệ thống mã hóa mềm với 313 hộp trong không gian màu.

**YOseg**: Mặt nạ phân đoạn ngữ nghĩa, phân loại từng pixel của ảnh vào một trong 183 lớp phân đoạn (đối tượng hoặc phần của cảnh).

Vấn đề tô màu có thể được coi là một quá trình học bán giám sát, liên quan đến việc học các đặc trưng tô màu bằng cách sử dụng phiên bản grayscale của một hình ảnh màu làm dữ liệu đầu vào và sau đó học phiên bản màu của chính hình ảnh đó. Trong giai đoạn kiểm tra, chúng tôi sử dụng mô hình học để dự đoán hình ảnh grayscale mà không có phiên bản màu.

Do đó, trong bước đầu tiên, chúng tôi xây dựng dữ liệu tô màu bằng cách chuyển đổi hình ảnh màu IRGB thành ILab trong không gian màu CIE Lab**. Không gian này có ba tọa độ: L đại diện cho trục grayscale, trục a đại diện cho sắc xanh-lục và đỏ**. **trục b đại diện cho thành phần xanh dương và vàng.**

Chúng tôi chọn không gian màu này vì chỉ số đo của nó thuận tiện cho việc đo lường sự tương quan giữa kênh độ sáng L và kênh màu ab. Với bối cảnh này, hình ảnh đầu vào XL có kích thước RH×W×1 là kênh độ sáng IL. Nhiệm vụ tô màu chỉ cần học các giá trị màu hồi quy **Yab** trong các kênh **a** và **b** của **Iab**. Dựa trên phong cách ngữ cảnh toàn cục và các yếu tố ngữ nghĩa ở mức điểm ảnh, chúng tôi đề xuất **vector phân phối màu chỉ mục Zab** cho giải pháp đa mô hình, **vector ngữ cảnh không chắc chắn Yclas** bằng cách chuyển giao học để có phong cách toàn cục, và **phân đoạn ngữ nghĩa Yseg** cho các yếu tố ngữ nghĩa ở mức điểm ảnh. Chi tiết được mô tả trong **Phần III-B**.

Chúng tôi chọn cấu trúc **U-net** [12] làm nền tảng để xây dựng mạng của mình vì nó có nhiều ưu điểm. Đầu tiên, mạng này phù hợp với nhiệm vụ tô màu hoàn toàn tự động vì nó có thể được huấn luyện theo phương pháp end-to-end. Thêm vào đó, chúng tôi cũng sử dụng kiến trúc bao gồm **đường đi co lại E(XL)** để mã hóa kênh độ sáng thành các đặc trưng mã hóa cấp thấp **Xenc**, và **đường đi mở rộng D(Xenc)**, có khả năng giải mã các đặc trưng mã hóa thành bản đồ đặc trưng 2D **Xmap**.

**Đặc trưng mã hóa Eenc(XL):** Sử dụng **VGG16** [31], gồm bốn khối convolution với hai lớp convolution dùng kernel 3x3 và một lớp chuẩn hóa batch, cùng một khối convolution ở giữa cho đặc trưng mã hóa. Sau mỗi khối mã hóa, độ phân giải của hình ảnh giảm một nửa nhờ vào việc sử dụng striding 2 trong lớp convolution cuối cùng trong khối và số lượng bộ lọc tăng gấp đôi.

**Đặc trưng giải mã D(Xenc):** Bao gồm bốn khối giải mã, mỗi khối có một lớp up-sampling và ba lớp convolution. Sau mỗi lần giải mã, lớp up-sampling làm tăng gấp đôi độ phân giải và số lượng bộ lọc giảm một nửa. Cũng có các kết nối bỏ qua giữa lớp convolution cuối cùng trong khối mã hóa và lớp convolution đầu tiên trong khối giải mã ở cùng cấp độ độ sâu. Điều này giúp mô hình tránh tình trạng mất kết nối khi các đặc trưng từ khối giải mã dưới được chuyển lên khối giải mã trên do vấn đề vanishing hoặc dying-ReLU.

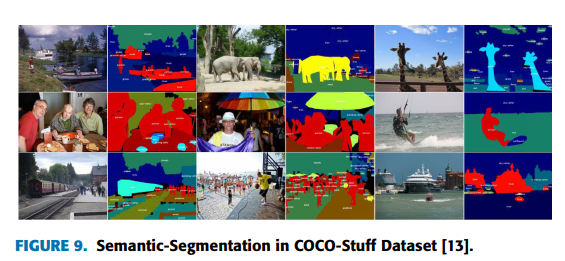
**Phân loại ngữ cảnh không chắc chắn Hclas(Xenc):**  
Phân loại ngữ cảnh không chắc chắn được đặt ở giữa mô hình **U-Net** để tăng cường **Xenc**. Nhánh phân loại này bao gồm lớp **global average pooling 2D**, hai lớp **dense**, và lớp **soft-max** để xuất ra vector one-hot chứa xác suất xuất hiện của ngữ cảnh cảnh vật.

Nhánh phân phối màu **Hsoft(Xdec)**, nhánh hồi quy màu **Hreg(Xdec)** và nhánh phân đoạn ngữ nghĩa **Hseg(Xdec)** được đặt sau mô hình giải mã **D(Xenc)**, với đầu vào là bản đồ đặc trưng 2D **Xdec**.

Bằng cách sử dụng lớp **convolution 1x1** với số bộ lọc giống như số giá trị đầu ra (hai cho hồi quy màu, 313 cho phân phối màu, và 183 cho phân đoạn ngữ nghĩa), chúng tạo ra đầu ra ở mức điểm ảnh với cùng kích thước như đầu vào dạng xám. Chúng chỉ khác nhau ở lớp kích hoạt với hàm **tanh** cho hồi quy màu và **softmax** cho phân phối màu và phân đoạn.

EXPERIMENTS AND DISCUSSION ( Thí nghiệm và thảo luận)

DATASET





Các thí nghiệm của chúng tôi được thực hiện trên tập dữ liệu **COCO-Stuff** [13] như hình 9. **COCO-Stuff** là một phân tập của tập dữ liệu **COCO** dành cho phát hiện đối tượng quy mô lớn, phân đoạn và chú thích, với hơn **118.000 hình ảnh** cho quá trình huấn luyện và **5.000 hình ảnh** cho bộ kiểm tra. Tập dữ liệu này gồm **172 lớp** bao gồm **80 lớp vật thể (things)**, **91 lớp vật liệu (stuff)** và một lớp không nhãn (unlabeled). Chúng tôi cũng chuyển mỗi hình ảnh về kích thước **224×224** trước khi bắt đầu quá trình huấn luyện.

**Hình 9:** **Phân đoạn ngữ nghĩa trong tập dữ liệu COCO-Stuff** [13].

Đối với **ground-truth cảnh vật**, chúng tôi sử dụng trọng số đã được huấn luyện trước trên mô hình **VGG16** của tập dữ liệu **Places365-Standard** [15]. Chúng tôi dự đoán xác suất cảnh vật của **5.000 hình ảnh** từ bộ kiểm tra **COCO-Stuff**, lấy **5 xác suất cao nhất** và chuẩn hóa giá trị về 1. Hình 10 cho thấy một số hình ảnh và lớp trong **Places365-Standard**, với **1.8 triệu hình ảnh** trong bộ huấn luyện, cùng với **365 lớp cảnh/vị trí** khác nhau. Mỗi lớp có từ **3.068 đến 5.000 hình ảnh**.

**Hình 10:** **Phân loại ngữ cảnh cảnh vật trong tập dữ liệu Places365** [15].

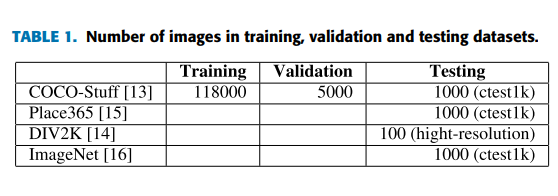
Để kiểm tra tô màu, chúng tôi xây dựng **1000 hình ảnh đầu tiên** từ các bộ kiểm tra của bốn tập dữ liệu: **ImageNet** [16], **Places365** [15], và **COCO-Stuff** [13], cùng với **100 hình ảnh** từ bộ kiểm tra độ phân giải cao trong tập dữ liệu **DIV2K** [14] như được hiển thị trong **Hình 11**. Ngoài ra, chúng tôi cũng thu thập một số hình ảnh từ Internet để xác nhận kết quả tô màu.



**Hình 11:** Hình ảnh từ bộ kiểm tra DIV2K và các bộ kiểm tra khác.

Các hình ảnh được xuất bản trên Internet bao gồm nhiều loại hình ảnh về cuộc sống và di sản.

**Bảng 1** liệt kê số lượng hình ảnh được sử dụng cho huấn luyện, kiểm tra và kiểm tra trong nghiên cứu này.



Ý nghĩa bức ảnh:

COCO : Training (Huấn luyện): 118,000 ảnh. Đây là số lượng ảnh được sử dụng để huấn luyện mô hình. Trong học máy (machine learning), tập huấn luyện là tập dữ liệu chính mà mô hình sẽ "học" từ đó. Mô hình sẽ điều chỉnh các tham số của nó (như trọng số trong mạng nơ-ron) dựa trên 118,000 ảnh này để học cách tô màu ảnh một cách chính xác. Số lượng lớn (118,000) đảm bảo mô hình có đủ dữ liệu để học các đặc trưng và mẫu hình đa dạng từ ảnh.

Validation (Xác nhận): 5,000 ảnh

Tập xác nhận được sử dụng trong quá trình huấn luyện để đánh giá hiệu suất của mô hình, nhưng không dùng để điều chỉnh trực tiếp các tham số của mô hình. Sau mỗi vòng huấn luyện (epoch), mô hình sẽ được kiểm tra trên tập xác nhận này để xem nó hoạt động tốt như thế nào trên dữ liệu mà nó chưa được huấn luyện trực tiếp. Điều này giúp:

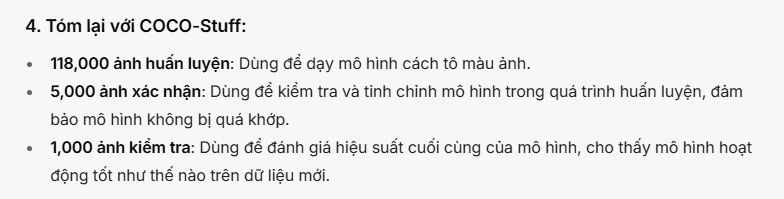
Theo dõi xem mô hình có đang học tốt hay không.

Phát hiện hiện tượng quá khớp (overfitting), tức là khi mô hình học quá tốt trên tập huấn luyện nhưng không tổng quát hóa tốt trên dữ liệu mới.

Điều chỉnh các siêu tham số (hyperparameters) như tốc độ học (learning rate) hoặc cấu trúc mô hình.

Testing (Kiểm tra): 1,000 ảnh (test1K)

Tập kiểm tra là tập dữ liệu hoàn toàn tách biệt, được sử dụng sau khi quá trình huấn luyện và xác nhận hoàn tất. Mục đích của tập kiểm tra là để đánh giá hiệu suất cuối cùng của mô hình trên dữ liệu mà nó chưa từng thấy trước đó. 1,000 ảnh trong tập test1K của COCO-Stuff được dùng để đo lường khả năng tổng quát hóa của mô hình, tức là xem mô hình có thể áp dụng những gì đã học được từ tập huấn luyện và xác nhận để tô màu ảnh mới hay không.



**CẤU HÌNH MÔI TRƯỜNG, CHI TIẾT HUẤN LUYỆN**

Môi trường phát triển của chúng tôi là **Python 3.7** sử dụng **TensorFlow Keras 2**. Chúng tôi sử dụng một máy tính để bàn với **Intel Core i5 8400**, **32 GB RAM** và **card đồ họa GeForce GTX 1080 8GB RAM**.

**Sơ đồ huấn luyện:**  
Trong nghiên cứu này, chúng tôi không sử dụng **cross-validation** vì huấn luyện trên tập dữ liệu lớn **COCO-Stuff** và kiểm tra trên các bộ kiểm tra khác nhau từ **DIV2K**, **Place365**, **ImageNet**, và **COCO-Stuff**.

**Dữ liệu huấn luyện:**  
Chúng tôi đã huấn luyện mô hình trên **118.000 hình ảnh** của bộ huấn luyện **COCO-Stuff**, và xác nhận mô hình trên **5.000 hình ảnh**. Mỗi đầu vào được thay đổi kích thước tối đa là **256 pixels**. Một số kỹ thuật được sử dụng cho **tăng cường dữ liệu** bao gồm **tăng độ tương phản ngẫu nhiên**, **độ sáng ngẫu nhiên**, **lật ngang ngẫu nhiên**, **xoay**, **thay đổi tỷ lệ** và **dịch chuyển**.

**Chi tiết huấn luyện:**

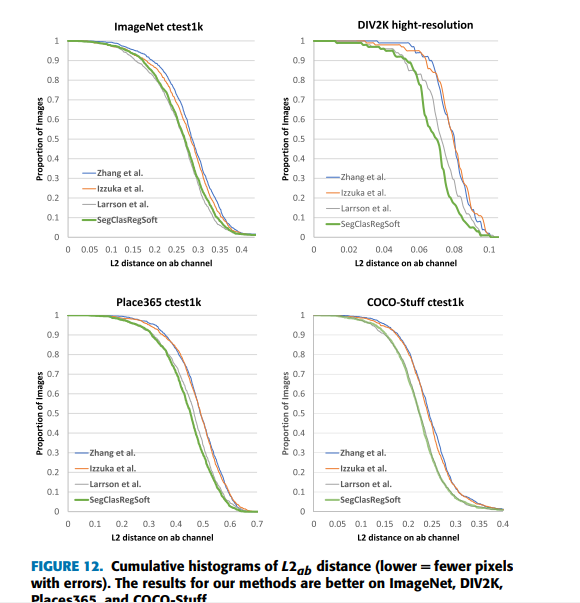
Để tránh hiện tượng **overfitting** khi học đa nhiệm, quá trình huấn luyện mô hình bao gồm hai giai đoạn:

**Giai đoạn đầu tiên:** Huấn luyện bắt đầu với **tốc độ học ban đầu là 0.0004** sử dụng **giảm tốc độ học khi đạt đỉnh (reduce learning rate on plateau)** mỗi **5 epoch** với tỷ lệ giảm là **0.95**. Sau đó, mô hình được tinh chỉnh lại bằng cách sử dụng **Cycle Learning Rate** với tốc độ học ban đầu là **0.0008**, và các giá trị của tốc độ học thay đổi tuần hoàn trong khoảng [0.0008, 0.00001] trong **8 epoch** để tối ưu hóa việc tô màu, phân loại cảnh ngữ nghĩa và phân đoạn ngữ nghĩa.

Mỗi cấu hình yêu cầu **ba ngày** huấn luyện.

Một số hàm tối ưu hóa đã được áp dụng trong quá trình học như **Adam**, **SGD** và **RMSProp**. Trong đó, **Adam** và **RMSProp** mang lại kết quả tốt hơn so với **SGD**.

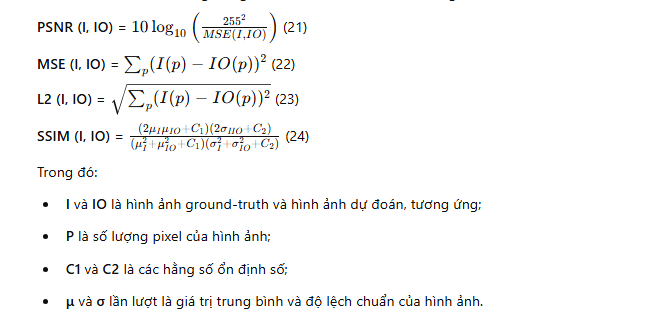
**CÁC CHỈ SỐ ĐÁNH GIÁ VÀ PHƯƠNG PHÁP SO SÁNH**

****

**Chỉ số Đánh giá:**  
Theo các nghiên cứu trước, chúng tôi sử dụng hai loại chỉ số đánh giá: **chỉ số định lượng** và **chỉ số chất lượng**.

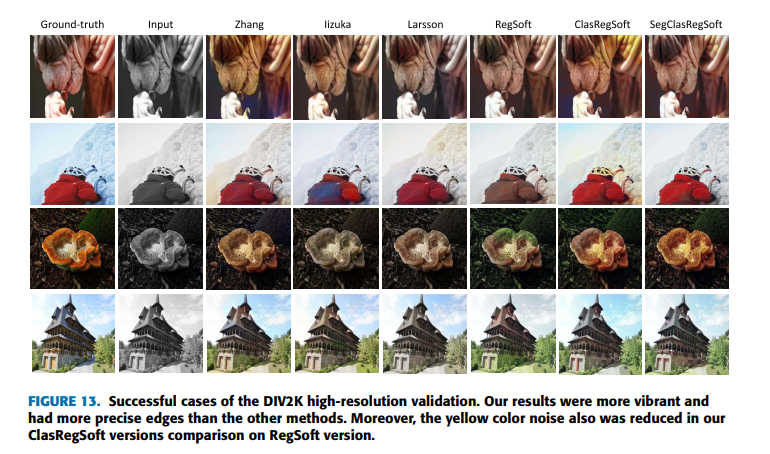
**Hình 12:** Biểu đồ tích lũy khoảng cách L2ab (giá trị nhỏ hơn = ít pixel có lỗi). Kết quả của phương pháp chúng tôi tốt hơn trên **ImageNet**, **DIV2K**, **Places365**, và **COCO-Stuff**.

Đối với các chỉ số định lượng, chúng tôi chia chúng thành hai nhóm nhỏ: **chỉ số tương đồng** và **phương pháp cảm nhận**. Có hai chỉ số tương đồng được sử dụng trong nghiên cứu này: **Tỷ số Độ nhiễu tín hiệu đỉnh (PSNR)**, **Chỉ số Tương đồng Cấu trúc (SSIM)** [34], và **Khoảng cách L2 của kênh ab**, như dưới đây:



Đối với một hình ảnh màu, chúng tôi tính toán các chỉ số này trên mỗi kênh và lấy kết quả trung bình. Các đánh giá **PSNR** và **SSIM** được thực hiện trên hình ảnh màu **RGB**. **L2** được đánh giá dựa trên kênh màu **ab**.

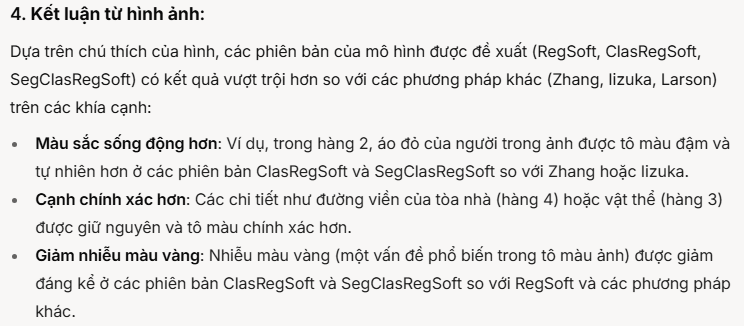
Các chỉ số tương đồng này định lượng chất lượng tái tạo và sự tương đồng cấu trúc giữa **ground-truth** và các hình ảnh đã được tô màu.



 **RegSoft**: Kết quả từ phiên bản RegSoft của mô hình được đề xuất trong bài báo.

 **ClasRegSoft**: Kết quả từ phiên bản ClasRegSoft của mô hình được đề xuất.

 **SegClasRegSoft**: Kết quả từ phiên bản SegClasRegSoft của mô hình được đề xuất.



Với các màu được dự đoán, tuy nhiên, theo quan sát của chúng tôi, chúng không phản ánh đủ chất lượng thị giác. Do đó, chúng tôi đã thêm một hình thức đánh giá khác liên quan đến phương pháp cảm nhận trong các chỉ số định lượng như sau:

Đầu tiên, chúng tôi sử dụng **Khả năng giải thích ngữ nghĩa** [7], [11] qua độ chính xác phân loại trong các bộ dữ liệu **ImageNet** và **Places365** đối với các hình ảnh đã tô màu. Chúng tôi sử dụng **Top1-Acc** với trọng số đã được huấn luyện của **VGG16** trên **ImageNet** và **Places365**.

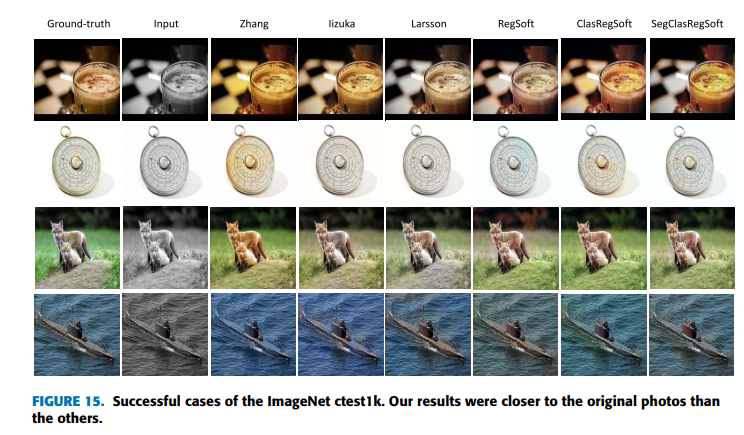
Thứ hai, chúng tôi cũng sử dụng **FrØchet Inception Distance (FID)** [35] để đo khoảng cách ngữ nghĩa giữa đầu ra tô màu và các hình ảnh tự nhiên thực tế.

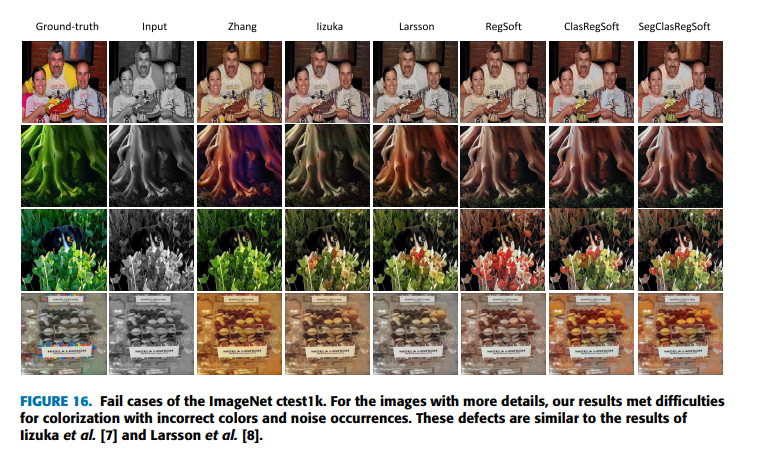
Cuối cùng, phương pháp của chúng tôi cũng được đánh giá bằng chỉ số cảm nhận **Learned Perceptual Image Patch Similarity (LPIPS)** được mô tả bởi **Zhang et al.** [36] (với backbone **AlexNet**, phiên bản 0.1). Chỉ số này đánh giá mức độ tương thích của các chỉ số với phán đoán cảm nhận của con người dưới các biến dạng truyền thống: nhiễu, quang học, mờ, biến dạng và nén bằng cách tính toán khoảng cách cosine từ các đặc trưng vector chuẩn hóa.

Đối với các chỉ số chất lượng, chúng tôi sử dụng **hiệu suất hình ảnh** để trình bày các trường hợp thành công cũng như các trường hợp thất bại để so sánh. Chúng tôi đã đánh giá hiệu suất hình ảnh trên bốn bộ dữ liệu công khai **DIV2K**, **ImageNet**, **COCO-Stuff**, và **Places365**. Chúng tôi cũng trình bày kết quả tô màu trên các hình ảnh di sản và cuộc sống thu thập từ Internet.

Về phương pháp so sánh, phương pháp của chúng tôi được so sánh với ba phương pháp tô màu mạnh mẽ là **Iizuka et al.** [7], **Larsson et al.** [8], và **Zhang et al.** [11], như được trình bày trong **Bảng 2**.

Chúng tôi cũng sử dụng trọng số đã được huấn luyện từ các tác giả này để dự đoán màu cho hình ảnh và so sánh với dự đoán của chúng tôi, như được liệt kê trong **Bảng 2**. Mặc dù có sự khác biệt trong...





Hình 15. Các trường hợp thành công của tập kiểm tra ImageNet test1K. Kết quả của chúng tôi gần với ảnh gốc hơn so với các phương pháp khác.

#### **Cấu trúc của Figure 15**:

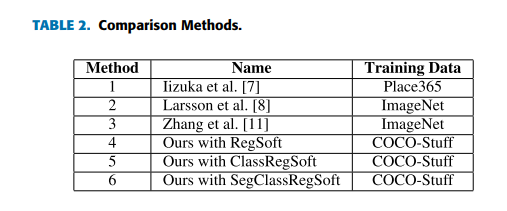
* **Cột**:
  + **Ground-truth**: Ảnh gốc có màu (ảnh tham chiếu).
  + **Input**: Ảnh đầu vào ở dạng thang độ xám (grayscale).
  + **Zhang**: Kết quả tô màu từ phương pháp của Zhang.
  + **Iizuka**: Kết quả tô màu từ phương pháp của Iizuka.
  + **Larson**: Kết quả tô màu từ phương pháp của Larson.
  + **RegSoft**: Kết quả từ phiên bản RegSoft của mô hình được đề xuất.
  + **ClasRegSoft**: Kết quả từ phiên bản ClasRegSoft.
  + **SegClasRegSoft**: Kết quả từ phiên bản SegClasRegSoft.

Figure 15 cho thấy các trường hợp thành công của các phương pháp tô màu ảnh trên tập kiểm tra ImageNet test1K (1,000 ảnh). ImageNet là một tập dữ liệu đa dạng, chứa nhiều loại ảnh khác nhau (đồ vật, động vật, cảnh thiên nhiên), nên việc đánh giá trên tập này giúp kiểm tra khả năng tổng quát hóa của mô hình.

**So sánh**:

* **Ground-truth**: Là ảnh gốc có màu, dùng làm tham chiếu để đánh giá chất lượng tô màu.
* **Input**: Ảnh thang độ xám, là đầu vào cho các phương pháp.
* **Zhang, Iizuka, Larson**: Đây là các phương pháp tô màu ảnh nổi tiếng. Kết quả của chúng thường có màu sắc không chính xác hoặc nhạt (ví dụ, trong hàng 3, con cáo của Zhang và Iizuka có màu nâu nhạt, không giống màu cam tự nhiên của Ground-truth).
* **RegSoft, ClasRegSoft, SegClasRegSoft**: Các phiên bản của mô hình được đề xuất trong bài báo. Kết quả của chúng (đặc biệt là ClasRegSoft và SegClasRegSoft) có màu sắc gần với Ground-truth hơn

**Thông điệp chính**: Các phiên bản của mô hình được đề xuất (đặc biệt là ClasRegSoft và SegClasRegSoft) vượt trội hơn các phương pháp khác (Zhang, Iizuka, Larson) trong việc tái tạo màu sắc gần với ảnh gốc trên tập dữ liệu ImageNet.



 Việc liệt kê tập dữ liệu huấn luyện cho thấy sự khác biệt trong cách các phương pháp được huấn luyện. Ví dụ, COCO-Stuff (dùng cho các mô hình của bài báo) có thể cung cấp thông tin ngữ nghĩa phong phú hơn, giúp mô hình hiểu ngữ cảnh và phân đoạn tốt hơn so với Places365 hoặc ImageNet.

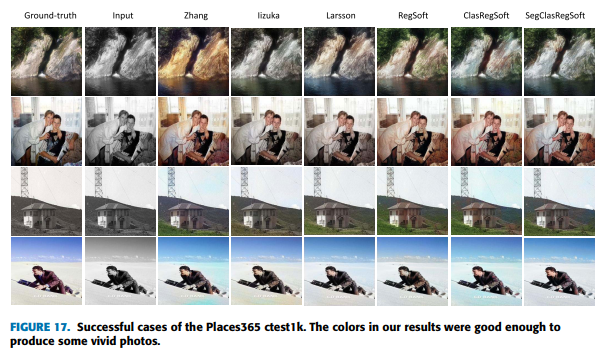
 Điều này cũng giải thích tại sao các mô hình của bài báo (RegSoft, ClasRegSoft, SegClasRegSoft) có thể hoạt động tốt hơn trong một số trường hợp, vì chúng được huấn luyện trên dữ liệu phù hợp hơn cho bài toán tô màu ảnh kết hợp ngữ nghĩa.

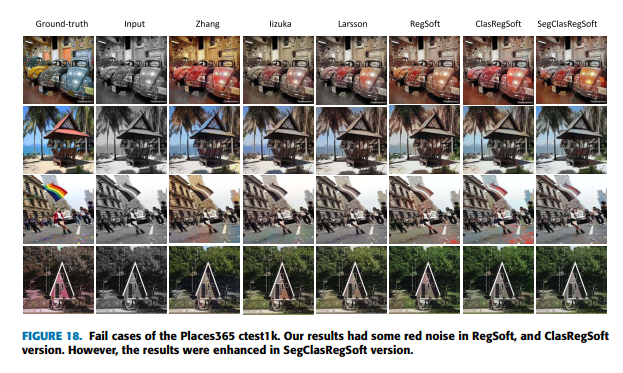
Tập dữ liệu huấn luyện, tất cả các hình ảnh được thu thập từ Internet và thể hiện nhiều hoạt động, đồ vật và cảnh trong tự nhiên. Trong nghiên cứu này, chúng tôi cũng tiến hành một nghiên cứu ablation để đánh giá một số thiết kế mô hình trong phương pháp được đề xuất của chúng tôi, như được liệt kê trong Bảng 2.

Chúng tôi đã khám phá ảnh hưởng của từng nhánh đối với kết quả tô màu. Đối với thiết kế RegSoft, chúng tôi chỉ giữ nhánh hồi quy và nhánh phân phối màu với các nhánh (2) + (3), như được hiển thị trong Hình 8. Đối với ClasRegSoft, chúng tôi đã thêm một nhánh phân loại vào mô hình của mình với các nhánh (1) + (2) + (3). Cuối cùng, thiết kế SegClasRegSoft có tất cả các nhánh (1) + (2) + (3) + (4), bao gồm các nhánh hồi quy, phân phối màu, phân loại và phân đoạn, tương ứng.

Chúng tôi có bốn loại đầu ra trong mô hình của mình. Phân đoạn ngữ nghĩa cung cấp mặt nạ nhãn để xác định ngữ nghĩa của các pixel trong hình ảnh, trong khi ngữ cảnh cảnh phân loại xác định ngữ cảnh toàn cục của một hình ảnh. Những nhánh này đã giúp mô hình của chúng tôi học cách tô màu ngữ nghĩa.

Đối với kết quả tô màu, mô hình của chúng tôi trả về từ nhánh 2 với hồi quy tô màu. Đầu ra của nhánh này cung cấp một kết quả trung bình với hiệu ứng mất bão hòa và hiệu ứng xám. Tuy nhiên, với việc sử dụng phân loại và phân đoạn, nhánh này đã cho kết quả tốt hơn, như được hiển thị trong các trường hợp thành công của ImageNet tại Hình 15 với ít nhiễu nhất và các trường hợp thành công của ImageNet tại Hình 15."





Hình 17. Các trường hợp thành công của tập kiểm tra PLACES365 test1K. Màu sắc trong kết quả của chúng tôi đủ tốt để tạo ra một số bức ảnh sống động.

**Mục đích**: Figure 17 cho thấy các trường hợp thành công của các phương pháp tô màu ảnh trên tập kiểm tra Places365 test1K (100 ảnh độ phân giải cao). Places365 là tập dữ liệu chuyên về các cảnh quan (scenes), nên việc đánh giá trên tập này giúp kiểm tra khả năng của mô hình trong việc tái tạo màu sắc cho các cảnh tự nhiên.

**So sánh**:

* **Ground-truth**: Là ảnh gốc có màu, dùng làm tham chiếu.
* **Input**: Ảnh thang độ xám, là đầu vào cho các phương pháp.
* **Zhang, Iizuka, Larson**: Các phương pháp này thường tạo ra màu sắc nhạt hoặc không chính xác. Ví dụ:
  + Trong hàng 1, Zhang và Iizuka tạo ra màu xanh lá nhạt cho núi, không giống màu xanh đậm tự nhiên của Ground-truth.
  + Trong hàng 2, Larson tạo ra màu xanh dương không đồng đều cho bầu trời và nước.
* **RegSoft, ClasRegSoft, SegClasRegSoft**: Các phiên bản của mô hình được đề xuất cho kết quả tốt hơn:
  + **RegSoft**: Màu sắc đã cải thiện so với Zhang, Iizuka, Larson, nhưng vẫn có thể nhạt hoặc thiếu độ sống động.
  + **ClasRegSoft và SegClasRegSoft**: Màu sắc sống động hơn, gần với Ground-truth hơn. Ví dụ:
    - Hàng 1: Thác nước và núi có màu xanh lá và xám tự nhiên hơn.
    - Hàng 2: Bầu trời và nước có màu xanh dương đậm, giống Ground-truth.
    - Hàng 4: Tuyết trắng và cây xanh được tái tạo chính xác hơn.

Hình 18. Các trường hợp thất bại của tập kiểm tra PLACES365 test1K. Kết quả của chúng tôi có một số nhiễu màu đỏ trong phiên bản RegSoft và ClasRegSoft. Tuy nhiên, kết quả đã được cải thiện trong phiên bản SegClasRegSoft, với nhiều màu sắc hơn trong ClasRegSoft và SegClasRegSoft so với thiết kế RegSoft.

 Figure 18 tập trung vào các trường hợp thất bại trên tập Places365 test1K, chỉ ra những hạn chế của các phương pháp, bao gồm cả mô hình được đề xuất.

**So sánh**:

* **Ground-truth và Input**: Tương tự như Figure 16.
* **Zhang, Iizuka, Larson**: Các phương pháp này tiếp tục gặp khó khăn trong việc tái tạo màu sắc chính xác. Ví dụ:
  + Trong hàng 1, Zhang và Iizuka tạo ra màu xanh dương nhạt hoặc không đồng đều cho nước và bầu trời.
  + Trong hàng 4, Larson tạo ra màu xanh lá không tự nhiên cho rừng.
* **RegSoft**: Phiên bản này có nhiễu màu đỏ (red noise) ở một số khu vực, ví dụ:
  + Trong hàng 1, nước và bầu trời có một số vùng bị nhuộm đỏ không tự nhiên.
* **ClasRegSoft**: Vẫn có nhiễu màu đỏ, nhưng màu sắc đã cải thiện hơn so với RegSoft, với nhiều màu sắc đa dạng hơn.
* **SegClasRegSoft**: Phiên bản này cải thiện đáng kể, giảm nhiễu màu đỏ và tạo ra nhiều màu sắc hơn, gần với Ground-truth hơn. Ví dụ:
  + Hàng 1: Bầu trời và nước có màu xanh dương tự nhiên hơn, ít nhiễu đỏ hơn.
  + Hàng 4: Cây xanh trong rừng có màu sắc phong phú hơn.

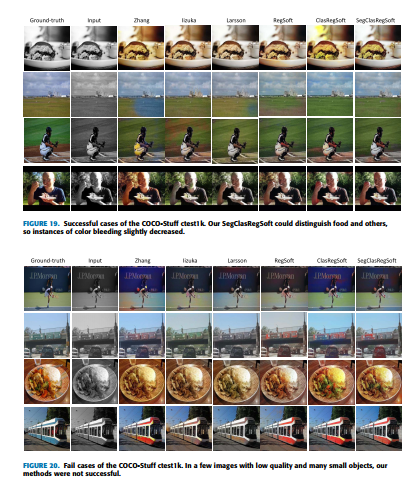
Cuối cùng, kết quả mã hóa mềm (soft-encoding) trong nhánh 3 trả về phân phối màu của các kênh ab. Kết quả tô màu được tính toán dựa trên trung bình có trọng số giữa các giá trị xác suất và cường độ tương ứng trong mỗi kênh a và b theo Phương trình 15. Sử dụng lược đồ trung bình ủ (annealing-mean scheme), chúng tôi điều chỉnh các giá trị xác suất trong vai trò của số hạng có trọng số bằng hàm mũ được kiểm soát bởi số hạng nhiệt độ T theo Phương trình 14. Lợi ích của lược đồ này là nó giữ được các ràng buộc không gian và loại bỏ nhiễu bằng hiệu ứng trung bình có trọng số. Khi T đạt 1, các xác suất màu gần như không đổi, dẫn đến hiệu ứng trung bình. Khi T đạt 0, các giá trị xác suất cao được tăng lên đáng kể hơn so với các giá trị thấp nhờ hiệu ứng của hàm mũ. Kết quả tô màu mang lại kết quả sống động hơn với ít hiện tượng nhiễu (artifacts).

**D. THÍ NGHIỆM 1: SO SÁNH ĐỊNH LƯỢNG TRÊN CÁC CHỈ SỐ TƯƠNG ĐỒNG**

Bảng 3 và Bảng 4 liệt kê kết quả của chúng tôi về các so sánh định lượng trên các Chỉ số Tương đồng. Đối với PSNR và SSIM, giá trị càng cao thì càng tốt. Đối với L2ab, giá trị càng thấp thì càng tốt. Phương pháp của Larsson và cộng sự đã cho kết quả PSNR gần như tốt hơn trên ImageNet, DIV2K và COCO-Stuff (lần lượt là 23.335, 23.49 và 23.773) và kết quả SSIM trên ImageNet và DIV2K (lần lượt là 0.869 và 0.929).

Tuy nhiên, các phương pháp của chúng tôi đã mang lại kết quả tốt nhất trên chỉ số L2ab cho DIV2K, Places365 và COCO-Stuff với các giá trị lần lượt là 0.068, 0.442 và 0.223. Kết quả cho thấy phân đoạn ngữ nghĩa đóng vai trò quan trọng trong việc nâng cao kết quả tô màu, và nó đã giúp phương pháp của chúng tôi cải thiện độ chính xác của các kênh ab.

Hình 12 thể hiện so sánh về phân bố sai số của L2ab trên các tập dữ liệu so sánh. Phương pháp hoàn toàn tự động của chúng tôi đã đạt được kết quả cạnh tranh so với các phương pháp khác, chứng minh hiệu quả của việc tích hợp ngữ cảnh cảnh toàn cục và phân đoạn ngữ nghĩa cấp pixel trong việc cải thiện độ chính xác của tô màu."



**Hình 19. Các trường hợp thành công của tập kiểm tra COCO-Stuff test1K. Phiên bản SegClasRegSoft của chúng tôi có thể phân biệt thực phẩm và các đối tượng khác, do đó các** trường hợp màu bị lem (color bleeding) đã giảm nhẹ.

 **Ground-truth và Input**: Là ảnh gốc và ảnh thang độ xám, dùng làm tham chiếu và đầu vào.

 **Zhang, Iizuka, Larson**: Các phương pháp này thường gặp vấn đề với hiện tượng **color bleeding** (màu bị lem), tức là màu sắc của một đối tượng bị lem sang đối tượng khác. Ví dụ:

* Trong hàng 3 (đĩa thức ăn), Zhang và Iizuka có thể làm cho màu của thức ăn lem sang nền, không phân biệt rõ ranh giới.
* Trong hàng 4 (con chim), Larson có thể làm cho màu của chim và cành cây bị lẫn lộn.

 **RegSoft và ClasRegSoft**: Các phiên bản này cải thiện so với Zhang, Iizuka, Larson, nhưng vẫn có thể gặp một số vấn đề về color bleeding.

 **SegClasRegSoft**: Phiên bản này vượt trội hơn, đặc biệt trong việc phân biệt các đối tượng (như thực phẩm và các đối tượng khác). Ví dụ:

* Hàng 3: Thức ăn được tô màu chính xác hơn, không bị lem sang nền.
* Hàng 4: Con chim và cành cây có màu sắc tách biệt rõ ràng hơn.
* Phiên bản SegClasRegSoft của mô hình được đề xuất có khả năng phân biệt các đối tượng (như thực phẩm và các đối tượng khác), từ đó giảm hiện tượng màu bị lem, dẫn đến kết quả tô màu chính xác và tự nhiên hơn.

**Hình 20. Các trường hợp thất bại của tập kiểm tra COCO-Stuff test1K. Trong một số hình ảnh có chất lượng thấp và chứa nhiều đối tượng nhỏ, các phương pháp của chúng tôi không thành công.**

 **Ground-truth và Input**: Là ảnh gốc và ảnh thang độ xám.

 **Zhang, Iizuka, Larson**: Các phương pháp này thường thất bại trong việc tái tạo màu sắc chính xác, đặc biệt trong các cảnh phức tạp. Ví dụ:

* Trong hàng 1 (cảnh đường phố), Zhang và Iizuka có thể tô màu không đồng đều cho xe hơi và người.
* Trong hàng 3 (đoàn tàu), Larson có thể tạo ra màu sắc sai lệch (như màu xanh không tự nhiên cho tàu).

 **RegSoft, ClasRegSoft, SegClasRegSoft**: Các phiên bản của mô hình được đề xuất cũng gặp khó khăn trong các trường hợp này:

* Hàng 1: Các đối tượng nhỏ như xe hơi và người không được tô màu chính xác, có thể bị lem màu hoặc màu sắc sai.
* Hàng 3: Đoàn tàu có thể bị tô màu không đồng đều hoặc sai (ví dụ, màu xám thay vì màu thực tế của tàu).
* Hàng 4: Các đối tượng nhỏ trong cảnh thiên nhiên (như đá, cây cối) không được phân biệt rõ, dẫn đến màu sắc không chính xác.

**Thông điệp chính**:

* Các phương pháp của bài báo (bao gồm SegClasRegSoft) gặp khó khăn khi xử lý các ảnh có chất lượng thấp và chứa nhiều đối tượng nhỏ. Điều này cho thấy hạn chế của mô hình trong các kịch bản phức tạp, nơi việc nhận diện và phân đoạn các đối tượng nhỏ trở nên khó khăn.
* Đây là một điểm yếu cần cải thiện trong các nghiên cứu tương lai, ví dụ bằng cách tăng cường khả năng xử lý ảnh chất lượng thấp hoặc cải thiện phân đoạn ngữ nghĩa cho các đối tượng nhỏ.