**Crafting a Toolchain for Image Restoration by Deep Reinforcement Learning**

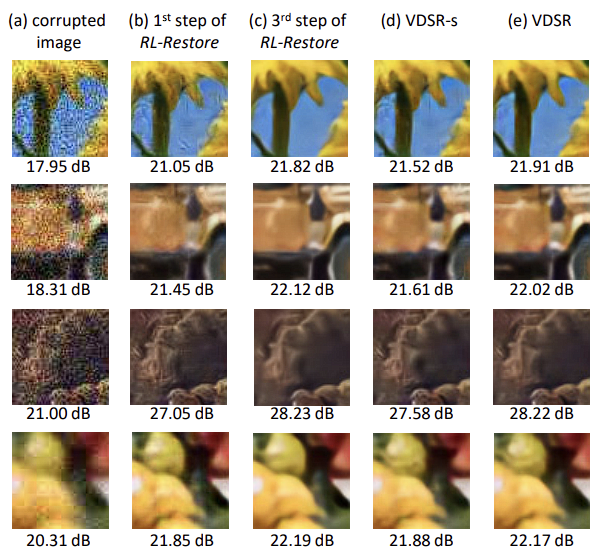
**Abstract**Chúng tôi điều tra một phương pháp mới cho việc khôi phục hình ảnh thông qua học tăng cường. Khác với các nghiên cứu hiện có chủ yếu huấn luyện một mạng lớn duy nhất cho một nhiệm vụ chuyên biệt, chúng tôi chuẩn bị một bộ công cụ bao gồm các mạng tích chập quy mô nhỏ với các độ phức tạp khác nhau và chuyên về các nhiệm vụ khác nhau. Phương pháp của chúng tôi, ***RL-Restore***, sau đó học một chính sách để chọn các công cụ phù hợp từ bộ công cụ để từ từ khôi phục chất lượng của một hình ảnh bị hỏng. Chúng tôi đề xuất một hàm phần thưởng từng bước tỷ lệ với việc hình ảnh được khôi phục tốt như thế nào ở mỗi bước để học chính sách hành động. Chúng tôi cũng đề xuất một kế hoạch học kết hợp để huấn luyện tác nhân và công cụ để có hiệu suất tốt hơn trong việc xử lý không chắc chắn. So với các mạng được thiết kế bởi con người theo cách truyền thống, ***RL-Restore*** có khả năng khôi phục hình ảnh bị hỏng với các biến đổi phức tạp và không biết trước một cách hiệu quả về tham số hơn bằng cách sử dụng chuỗi công cụ được hình thành động này.

1. **Introduction**

Mạng nơ-ron tích chập sâu (CNN) đã đạt được thành công to lớn, không chỉ trong các nhiệm vụ thị giác cấp cao, mà còn trong các nhiệm vụ thị giác cấp thấp như làm mờ, làm sạch nhiễu, giảm nhiễu nghệ thuật JPEG và siêu phân giải. Đặc biệt, hiệu suất tốt và tốc độ kiểm tra nhanh được chứng minh qua các phương pháp tối ưu dựa trên mô hình truyền thống.

Do tính phân biệt của CNN, hầu hết các mô hình này được huấn luyện để xử lý một nhiệm vụ thị giác cấp thấp cụ thể. Trong việc giảm nhiễu nghệ thuật JPEG, ví dụ, đã thiết kế các mạng khác nhau cho các chất lượng nén khác nhau để đạt được sự khôi phục đáng tin cậy. Trong trường hợp siêu phân giải, việc có các mạng khác nhau để xử lý các yếu tố tỉ lệ khác nhau là phổ biến. Một số nghiên cứu gần đây đã chỉ ra khả năng xử lý nhiều loại biến dạng hoặc đối phó với các cấp độ suy giảm khác nhau cùng một lúc bằng cách sử dụng CNN. Tuy nhiên, điều này thường đi kèm với chi phí sử dụng mạng sâu hơn nhiều. Ngoài ra, các mạng như vậy xử lý tất cả các hình ảnh với cùng một cấu trúc, mặc dù một số trong số đó ít khó khăn hơn mặc dù có thể được khôi phục một cách rẻ hơn.

Trong bài báo này, chúng tôi khám phá khả năng có một số CNN quy mô nhỏ nhưng chuyên biệt để giải quyết một nhiệm vụ khôi phục khó hơn một cách hợp tác. Ý tưởng của chúng tôi bắt đầu từ triết lý hiện tại rằng người ta cần một CNN có dung lượng lớn để giải quyết một nhiệm vụ khôi phục phức tạp. Thay vào đó, chúng tôi muốn có một bộ công cụ (dựa trên các CNN nhỏ) và học cách sử dụng chúng một cách linh hoạt để giải quyết nhiệm vụ hiện tại. Ý tưởng đã đề cập có thể mang lại cái nhìn mới về cách CNN có thể được sử dụng để giải quyết các nhiệm vụ khôi phục thực tế, trong đó hình ảnh có thể bị nhiễm bẩn bởi một hỗn hợp các biến dạng, chẳng hạn như làm mờ, nhiễu và khối sau một số giai đoạn xử lý. Hơn nữa, phương pháp mới có thể dẫn đến khôi phục hiệu quả về tham số so với các mô hình dựa trên CNN hiện có. Đặc biệt, các công cụ có độ phức tạp khác nhau có thể được chọn dựa trên mức độ biến dạng.



Hình 1. (a) hiển thị các hình ảnh bị nhiễm bẩn bởi các biến dạng phức tạp. (b-c) miêu tả một số bước quyết định được chọn để khôi phục một hình ảnh bằng RL-Restore. Tại mỗi bước, một công cụ cụ thể được chọn bởi tác nhân để cải thiện chất lượng hình ảnh. (d-e) là kết quả dựa trên CNN, trong đó (d) có số lượng tham số tương đương với RL-Restore trong khi (e) có gấp đôi. Giá trị PSNR được thể hiện để so sánh tốt hơn.

Để đạt được mục tiêu này, chúng tôi trình bày một khung công việc xem xét việc khôi phục hình ảnh như là một quá trình ra quyết định, trong đó một tác nhân sẽ chọn một chuỗi công cụ một cách linh hoạt để từ từ tinh chỉnh hình ảnh, và tác nhân có thể chọn dừng lại nếu chất lượng được khôi phục được đánh giá là đủ đáng. Trong khung công việc của chúng tôi, chúng tôi chuẩn bị một số CNN nhẹ với các độ phức tạp khác nhau. Chúng được đặc thù cho một nhiệm vụ cụ thể nhằm xử lý các loại bài tập khác nhau bao gồm làm mờ, làm sạch nhiễu hoặc giảm nhiễu nghệ thuật JPEG. Việc chọn thứ tự của các công cụ được công thức hóa trong một framework học tăng cường (RL). Một tác nhân học cách quyết định công cụ tiếp theo tốt nhất để chọn bằng cách phân tích nội dung của hình ảnh đã được khôi phục trong bước hiện tại và quan sát hành động cuối cùng được chọn. Phần thưởng được tích lũy khi tác nhân cải thiện chất lượng của hình ảnh đầu vào.

Chúng tôi đề cập đến framework được đề xuất là ***RL-Restore***. Chúng tôi tóm tắt đóng góp của mình như sau:

1) Chúng tôi trình bày một cố gắng mới để giải quyết việc khôi phục hình ảnh trong một framework học tăng cường. Khác với các phương pháp hiện có mà triển khai một cấu trúc mạng duy nhất và có thể lớn, RL-Restore tận hưởng tính linh hoạt của việc sử dụng các công cụ có khả năng khác nhau để đạt được sự khôi phục mong muốn.

2) Chúng tôi đề xuất một kế hoạch học kết hợp để huấn luyện tác nhân và công cụ đồng thời để framework có khả năng tốt hơn trong việc đối phó với các hiện tượng nghệ thuật mới và không biết trước xuất hiện giữa quá trình.

3) Chúng tôi chỉ ra rằng chuỗi công cụ được hình thành động thực hiện cạnh tranh so với các mạng được thiết kế bởi con người mạnh mẽ với độ phức tạp tính toán ít hơn. Phương pháp của chúng tôi có thể đối phó được với các biến dạng chưa được nhìn thấy đến một mức độ nhất định. Thú vị là, phương pháp của chúng tôi thậm chí còn minh bạch hơn so với các phương pháp hiện có vì nó có thể tiết lộ cách các biến dạng phức tạp có thể được loại bỏ từng bước một bằng cách sử dụng các công cụ khác nhau.

Hình 1(b-c) minh họa một chính sách học được để khôi phục một hình ảnh bị nhiễm bẩn bởi nhiều biến dạng, nơi chất lượng hình ảnh được tinh chỉnh từng bước một. Kết quả của hai mô hình CNN cơ sở được mô tả trong Hình 1(d-e), trong đó (d) có số lượng tham số tương tự như của chúng tôi (tác nhân + công cụ được áp dụng), trong khi (e) có gấp đôi. Như chúng tôi sẽ tiếp tục trình bày trong phần thực nghiệm, ***RL-Restore*** vượt trội hơn so với các phương pháp CNN khi có độ phức tạp tương tự và yêu cầu 82,2% lượng tính toán ít hơn để đạt được cùng hiệu suất với một mạng CNN lớn duy nhất.

1. **Related Work**

**CNN cho Việc Khôi Phục Hình Ảnh.** Việc khôi phục hình ảnh là một chủ đề được nghiên cứu một cách rộng rãi nhằm ước lượng hình ảnh rõ ràng/ban đầu từ một quan sát bị nhiễm bẩn/nhiễu. Các phương pháp dựa trên mạng nơ-ron tích chập (CNN) đã thể hiện hiệu suất xuất sắc trong các nhiệm vụ khôi phục hình ảnh khác nhau. Đa số các nghiên cứu này huấn luyện một mạng duy nhất chuyên biệt cho nhiệm vụ đang xử lý, ví dụ như làm mờ, làm sạch nhiễu, giảm nhiễu nghệ thuật JPEG và siêu phân giải. Công việc của chúng tôi cung cấp một phương pháp thay thế có hiệu quả về tham số hơn nhưng vẫn linh hoạt đối với hình thức của các biến dạng.

Có một số nghiên cứu tiên phong xử lý đồng thời nhiều suy giảm. Bằng cách phát triển một CNN sâu 20 tầng, Kim và cộng sự sử dụng một mô hình duy nhất để xử lý siêu phân giải hình ảnh đa quy mô. Guo và cộng sự xây dựng một mạng một-đến-nhiều có thể xử lý các hình ảnh với các mức độ nghệ thuật nén khác nhau. Zhang và cộng sự đề xuất một CNN sâu 20 tầng để đối phó đồng thời với nhiều nhiệm vụ khôi phục, bao gồm làm sạch nhiễu hình ảnh, giảm nhiễu nghệ thuật JPEG và siêu phân giải. Không có nghiên cứu nào trong số này xem xét các biến dạng kết hợp, trong đó một hình ảnh duy nhất bị ảnh hưởng bởi nhiều biến dạng. Khác biệt với các công trình đã đề cập, chúng tôi quan tâm đến việc khám phá xem liệu các CNN quy mô nhỏ từ 3 đến 8 tầng có thể được sử dụng để đồng thời khôi phục hình ảnh bị nhiễm bẩn bởi các biến dạng kết hợp.

Có các phương pháp có thể được sử dụng để nén một mạng lớn thành một mạng nhỏ hơn để tăng hiệu quả tính toán. Trong lĩnh vực của việc khôi phục hình ảnh, các mạng nơ-ron đệ quy được nghiên cứu để giảm số lượng tham số của mạng. Tuy nhiên, chi phí tính toán vẫn cao do số lượng lặp lớn. Mục tiêu của công việc của chúng tôi là đối giao với các nghiên cứu đã đề cập ở trên - framework của chúng tôi tiết kiệm tham số và tính toán thông qua việc học một chính sách để đưa ra quyết định trong việc chọn các mạng CNN phù hợp cho một nhiệm vụ thay vì nén một mạng hiện có.

**Học Tăng Cường Sâu.** Học tăng cường là một công cụ mạnh mẽ để học một tác nhân đưa ra quyết định tuần tự để tối đa hóa phần thưởng tích lũy. Các công trình đầu tiên của RL chủ yếu tập trung vào điều khiển robot. Gần đây, các thuật toán RL truyền thống được tích hợp vào các framework học sâu và được áp dụng thành công trong các lĩnh vực khác nhau như các tác nhân trò chơi và thiết kế kiến trúc mạng nơ-ron. Sự chú ý cũng được dành cho RL sâu trong lĩnh vực thị giác máy tính. Ví dụ, Huang và cộng sự sử dụng RL để học một chính sách quyết định sớm để tăng tốc việc theo dõi đối tượng bằng CNN. Cao và cộng sự khám phá các thuật toán RL sâu trong thị giác cấp thấp và áp dụng cơ chế chú ý vào việc phục hình. Trong nghiên cứu này, chúng tôi điều tra việc lựa chọn công cụ khôi phục trong một framework RL. Vấn đề này là mới mẻ trong văn bản.

1. **Learning a restroration toolchain**

**Định nghĩa Vấn đề.** Cho một hình ảnh bị nhiễm bẩn **I**dis, mục tiêu của chúng tôi là khôi phục một hình ảnh rõ ràng **I**res gần giống với hình ảnh đúng nền **I**gt. Quá trình biến dạng có thể được công thức hóa như sau:

**I**dis = D(**I**gt); D = Dn ◦ · · · ◦ D1, (1)

trong đó ◦ biểu thị sự kết hợp chức năng và mỗi D1, . . . , Dn đại diện cho một loại biến dạng cụ thể. Khác với các phương pháp hiện có [6, 8, 31, 37, 41] tập trung vào một loại biến dạng duy nhất, chúng tôi dự định xử lý một hỗn hợp của nhiều biến dạng (tức là, n > 1). Ví dụ, hình ảnh đầu ra cuối cùng có thể bị ảnh hưởng tuần tự bởi việc làm mờ ngoài tiêu điểm, nhiễu và nén JPEG. Trong trường hợp như vậy, số lượng biến dạng n là 3, và D1, D2, D3 lần lượt đại diện cho mờ, nhiễu và nén. Để giải quyết các biến dạng kết hợp, chúng tôi đề xuất khôi phục hình ảnh bị hỏng từng bước một với một chuỗi công cụ khôi phục.

A screenshot of a computer program

Description automatically generated

Hình 2. Các chuỗi công cụ khác nhau cho việc khôi phục hình ảnh. Chúng tôi tiến hành một thử nghiệm sơ bộ ở đây. Cho hai hình ảnh bị nhiễm bẩn và các chuỗi công cụ phù hợp tương ứng như (c) và (d), chúng tôi xây dựng các chuỗi công cụ khác nhau bằng cách sắp xếp lại thứ tự (được biểu thị bằng hình dạng) hoặc điều chỉnh cấp độ (được biểu thị bằng màu sắc) của các công cụ được chọn. Các kết quả khôi phục cho thấy rằng những thay đổi nhỏ của một chuỗi công cụ có thể dẫn đến hiệu suất rất khác nhau.

**Thách Thức.** *Nhiệm vụ của việc lựa chọn công cụ không phải là dễ dàng và đặt ra những thách thức đặc biệt đối với RL. Đầu tiên, sự lựa chọn của loại khôi phục, cấp độ và thứ tự xử lý đều ảnh hưởng đến hiệu suất cuối cùng*. Một ví dụ được thể hiện trong Hình 2, trong đó các hình ảnh bị nhiễm bẩn bởi hai kết hợp khác nhau của các biến dạng. Với một chuỗi công cụ phù hợp, như trong Hình 2 (c, d), chất lượng hình ảnh và giá trị Peak Signalto-Noise Ratio (PSNR) được cải thiện một cách tuần tự. Sau đó, chúng tôi sắp xếp lại thứ tự công cụ một chút như trong Hình 2(b, e) hoặc điều chỉnh cấp độ khôi phục của các công cụ như trong Hình 2(a, f). Kết quả chỉ ra rằng các thay đổi nhỏ trong một chuỗi công cụ có thể ảnh hưởng nghiêm trọng đến hiệu suất khôi phục. Cụ thể, việc sử dụng các công cụ không phù hợp có thể dẫn đến các đầu ra không tự nhiên, như là quá sắc nét trong Hình 2(a) và làm mờ trong Hình 2(f). Ngay cả khi các công cụ được lựa chọn tốt, một thứ tự không phù hợp có thể giảm hiệu suất (Hình 2(b, e)). Vì chuỗi công cụ có ảnh hưởng rất lớn đến kết quả, việc lựa chọn công cụ nào để sử dụng ở mỗi bước trở nên quan trọng.

Khi các công cụ được huấn luyện cho các nhiệm vụ cụ thể, chúng ta gặp phải một vấn đề khác là *không có công cụ nào có thể xử lý hoàn hảo 'trạng thái giữa'*, tức là kết quả trung gian sau một số bước xử lý. Bởi vì hầu hết các biến dạng không thể đảo ngược được, việc khôi phục sự kết hợp của chúng không phải là một sự kết hợp đơn giản của các công cụ khôi phục tương ứng. Các hiện tượng nghệ thuật mới có thể được giới thiệu trong các trạng thái trung gian. Ví dụ, việc làm mờ sẽ cũng cải thiện nhiễu, gây ra các hiện tượng nghệ thuật mới không thể loại bỏ bằng cách sử dụng các công cụ làm sạch nhiễu sau đó. Thách thức này là đặc biệt đối với nhiệm vụ của chúng tôi.

Để giải quyết thách thức đầu tiên, chúng tôi xem xét vấn đề lựa chọn công cụ tuần tự như một Markov Decision Process (MDP) và giải quyết nó theo một cách học tăng cường sâu. Để giải quyết thách thức thứ hai, chúng tôi đề xuất một phương pháp huấn luyện để tinh chỉnh đồng thời tác nhân và công cụ sao cho các công cụ được thông tin tốt hơn về các trạng thái giữa được quan sát bởi tác nhân. Trước hết, chúng tôi cung cấp một cái nhìn tổng quan về framework được đề xuất như sau.

**Tổng Quan về RL-Restore**. Framework được đề xuất nhằm mục đích khám phá một chuỗi công cụ cho một hình ảnh đầu vào bị hỏng. Như được thể hiện trong Hình 3, *RL-Restore* bao gồm hai thành phần chính:

1) Một hộp công cụ chứa các công cụ khác nhau cho việc khôi phục hình ảnh

2) Một tác nhân có cấu trúc tái phát động, động viên chọn một công cụ ở mỗi bước hoặc một hành động dừng sớm. Chúng tôi giả định quá trình lựa chọn công cụ là một quy trình học tăng cường - một chuỗi quyết định về lựa chọn công cụ được thực hiện để tối đa hóa một phần thưởng tỷ lệ với chất lượng của hình ảnh đã khôi phục. Tiếp theo, chúng tôi sẽ mô tả một cài đặt hợp lý của hộp công cụ và sau đó giải thích chi tiết về tác nhân.

* 1. **Toolbox**

Hộp công cụ chứa một tập hợp các công cụ có thể được áp dụng vào hình ảnh bị hỏng. Mục tiêu của chúng tôi là thiết kế một hộp công cụ mạnh mẽ và nhẹ, vì vậy chúng tôi hạn chế mỗi công cụ chỉ thành thạo trong một nhiệm vụ cụ thể. Điều này có nghĩa là mỗi công cụ chỉ được huấn luyện trên một phạm vi hẹp của các biến dạng. Để giảm thiểu tổng thể độ phức tạp, chúng tôi sử dụng các mạng nhỏ hơn cho các nhiệm vụ dễ dàng hơn. Đối với mục đích nghiên cứu của chúng tôi, chúng tôi chuẩn bị 12 công cụ như được hiển thị trong Bảng 1, trong đó mỗi công cụ được gán để giải quyết một mức độ nhất định của làm mờ Gaussian, nhiễu Gaussian hoặc nén JPEG. Chúng tôi áp dụng một CNN ba tầng (như trong [8]) cho các biến dạng nhẹ và một CNN tám tầng sâu hơn cho các biến dạng nghiêm trọng. Lưu ý rằng các công cụ không cần bị hạn chế để giải quyết các biến dạng đã đề cập ở trên. Chúng tôi đã thực hiện các lựa chọn này vì chúng thường được xem xét trong văn bản về khôi phục hình ảnh. Trong thực tế, một người có thể thiết kế các công cụ của họ với độ phức tạp phù hợp dựa trên nhiệm vụ cụ thể.

Như đã thảo luận ở đầu *Mục 3*, một tập hợp hữu hạn các công cụ không hoàn hảo để xử lý các hiện tượng nghệ thuật mới xuất hiện trong các trạng thái trung gian. Để giải quyết vấn đề này, chúng tôi đề xuất hai chiến lược:

1) Để tăng tính ổn định của các công cụ, chúng tôi thêm các nhiễu Gaussian nhẹ và nén JPEG cho tất cả dữ liệu huấn luyện.

2) Sau khi huấn luyện tác nhân, tất cả các công cụ được điều chỉnh chung dựa trên các chuỗi công cụ được huấn luyện tốt. Sau đó, các công cụ sẽ linh hoạt hơn đối với nhiệm vụ của tác nhân, và có thể xử lý các trạng thái trung gian một cách mạnh mẽ hơn. Chúng tôi thảo luận về các bước huấn luyện trong *Mục 3.3*. Các thử nghiệm trong *Mục 4* xác nhận tính hiệu quả của các chiến lược được đề xuất.

A diagram of a computer program

Description automatically generated

Hình 3. Minh họa về framework RL-Restore của chúng tôi. Tại mỗi bước t, tác nhân fag quan sát trạng thái hiện tại St, bao gồm hình ảnh đã khôi phục hiện tại It và vector giá trị đầu vào v˜t, đó là đầu ra của tác nhân tại bước trước đó. Lưu ý rằng I1 đại diện cho hình ảnh đầu vào và v˜1 là một vector không. Dựa trên giá trị tối đa của đầu ra vt của tác nhân, một hành động at được lựa chọn và công cụ tương ứng được sử dụng để khôi phục hình ảnh hiện tại. Sau quá trình khôi phục fr, với hình ảnh mới được khôi phục It+1 và vector giá trị v˜t+1 = vt, RL-Restore tiến hành một bước khôi phục khác một cách lặp lại cho đến khi hành động dừng được chọn.

A table of numbers and letters

Description automatically generated

Bảng 1. Công cụ trong hộp công cụ. Chúng tôi xem xét ba loại biến dạng và các cấp độ suy giảm khác nhau. Mỗi công cụ là một CNN ba tầng hoặc một CNN tám tầng tùy thuộc vào biến dạng mà nó nhắm vào để giải quyết.

* 1. **Agent**

Hệ thống xử lý của ***RL-Restore*** được hiển thị trong Hình 3. Cho một hình ảnh đầu vào, tác nhân đầu tiên chọn một công cụ từ hộp công cụ và sử dụng nó để khôi phục hình ảnh, sau đó tác nhân chọn một công cụ khác dựa trên kết quả trước đó và lặp lại quá trình khôi phục cho đến khi nó quyết định dừng lại. Trước hết, chúng tôi sẽ làm rõ một số thuật ngữ như hành động, trạng thái và phần thưởng, sau đó đi vào chi tiết về cấu trúc của tác nhân và quy trình khôi phục.

**Action.** Không gian hành động, được ký hiệu là *A*, là một tập hợp của tất cả các hành động có thể tác nhân có thể thực hiện. Tại mỗi bước *t*, một hành động at được chọn và áp dụng vào hình ảnh đầu vào hiện tại. Mỗi hành động đại diện cho một công cụ trong hộp công cụ và có một hành động bổ sung đại diện cho việc dừng lại. Nếu có *N* công cụ trong hộp công cụ, thì độ lớn của *A* là *N + 1*. Do đó, đầu ra, ***v****t*, của tác nhân là một vector *(N + 1)-chiều* mô tả giá trị của mỗi hành động. Một khi hành động dừng lại được chọn, quy trình khôi phục sẽ kết thúc và hình ảnh đầu vào hiện tại sẽ trở thành kết quả cuối cùng.

**State.** Trạng thái chứa thông tin mà tác nhân có thể quan sát được. Trong công thức của chúng tôi, trạng thái được công thức hóa như sau: *St = {****I****t,* ***v****’t}*, trong đó ***I****t*là hình ảnh đầu vào hiện tại, và ***v˜****t* là vector hành động lịch sử trước đó. Tại bước 1, ***I****1* là hình ảnh đầu vào và***v˜****1* là một vector không. Trạng thái cung cấp kiến thức ngữ cảnh phong phú cho tác nhân.

1) Hình ảnh đầu vào hiện tại ***I****t* quan trọng vì hành động được chọn sẽ được áp dụng trực tiếp vào hình ảnh này để tạo ra một kết quả khôi phục tốt hơn.

2) Thông tin về vector hành động trước đó ***v˜****t*, là vector giá trị đầu ra của tác nhân ở bước *t - 1*, tức là ***v˜****t =* ***v****t−1*, cũng quan trọng. Kiến thức về quyết định trước đó có thể giúp việc lựa chọn hành động tại bước hiện tại. Điều này được thấy là hoạt động tốt hơn theo kinh nghiệm học thực tế so với việc chỉ sử dụng ***I****t*.

**Reward.** Phần thưởng thúc đẩy việc huấn luyện của tác nhân khi nó học cách tối đa hóa phần thưởng tích lũy. Tác nhân được giả định học một chính sách tốt để hình ảnh khôi phục cuối cùng là đáng tin cậy. Chúng tôi mong muốn đảm bảo rằng chất lượng hình ảnh được cải thiện ở mỗi bước, do đó một phần thưởng theo bước được thiết kế như sau:

*rt=Pt+1* − *Pt,*

Ở đây *rt* là hàm phần thưởng ở bước *t*, *Pt+1*biểu thị PSNR giữa ***I****t+1*và hình ảnh tham chiếu ***I****gt* ở cuối bước *t,* và *Pt* đại diện cho PSNR đầu vào ở bước t. Phần thưởng tích lũy có thể được viết lại như sau , đây là tổng PSNR tổng thể trong quá trình khôi phục và được tối đa hóa để đạt được cải thiện tối ưu. Lưu ý rằng có linh hoạt để sử dụng các tiêu chí chất lượng hình ảnh khác (ví dụ: mất mát cảm giác, mất mát GAN) như phần thưởng trong khung ghi của chúng tôi. Sự điều tra này vượt ra ngoài phạm vi của bài báo này.

**Structure.** Tại mỗi bước *t*, tác nhân đánh giá giá trị của mỗi hành động dựa trên trạng thái đầu vào *St*, có thể được công thức như sau:

ở đây *fag* chỉ ra mạng tác nhân *Wag* là tham số của nó. Vector ***v****t* đại diện cho giá trị của các hành động. Hành động có giá trị tối đa được chọn là *at*, tức là *at = argmaxavt,a*, trong đó *vt,a* chỉ ra phần tử của vector giá trị ***v****t*tương ứng với hành động a.

Tác nhân được tạo thành từ ba mô-đun như được miêu tả trong Hình 3. Mô-đun đầu tiên, được đặt tên là trích xuất đặc trưng, là một CNN bốn tầng tiếp theo bởi một tầng fully-connected (fc) xuất ra một đặc trưng 32 chiều. Mô-đun thứ hai là một bộ mã hóa một-hot với đầu vào *N + 1* chiều và đầu ra N chiều, bảo tồn thông tin về hành động đã chọn trước đó. Lưu ý rằng đầu ra ít một chiều so với đầu vào, vì hành động dừng không thể được áp dụng ở bước trước đó, và do đó chúng ta đơn giản là loại bỏ chiều cuối cùng. Các đầu ra của hai mô-đun đầu tiên được nối với nhau để tạo thành đầu vào của mô-đun thứ ba, một Long Short-Term Memory (LSTM). LSTM không chỉ quan sát trạng thái đầu vào, mà còn lưu trữ các trạng thái lịch sử trong bộ nhớ của nó, cung cấp thông tin ngữ cảnh về các hình ảnh và hành động đã khôi phục lịch sử. Cuối cùng, với một tầng fc tiếp theo LSTM, một vector giá trị **𝑣**𝑡 được tạo ra để lựa chọn công cụ.

**Restoration.** Một khi một hành động **𝑎**𝑡 được thu được dựa trên giá trị tối đa trong **𝑣**𝑡, công cụ tương ứng sẽ được áp dụng vào hình ảnh đầu vào **𝐼**𝑡 để có được một hình ảnh đã được khôi phục mới:

,

ở đây *fr* biểu thị hàm khôi phục và *Wr* chỉ ra các tham số của một công cụ trong hộp công cụ. Nếu một hành động dừng lại được chọn, *fr* biểu thị một ánh xạ đồng nhất. Đặt ***I****dis* và ***I****res* là hình ảnh bị biến dạng đầu vào và kết quả cuối cùng đã được khôi phục tương ứng, quy trình khôi phục tổng thể có thể được biểu diễn như sau:

ở đây *f =[ fag; fr ]* và *W = [Wag; Wr ]. T* là bước khi hành động dừng được chọn. Chúng tôi cũng đặt một bước tối đa *Tmax* để ngăn chặn quá trình khôi phục quá mức. Khi *t = Tmax* và hành động dừng không được chọn, chúng tôi sẽ kết thúc quá trình khôi phục sau bước hiện tại. Nói cách khác, chúng tôi thêm một ràng buộc *T ≤ Tmax*.

* 1. **Training**

Đào tạo cho các công cụ tuân theo một phương pháp tiêu chuẩn được trình bày trong [19], trong đó giảm thiểu sai số trung bình bình phương (MSE) . Ở đây, ***y*** đại diện cho hình ảnh mục tiêu, ***x*** là hình ảnh đầu vào, và ***h*** là biểu thị công cụ. Về phần tác nhân, việc huấn luyện được thực hiện bằng cách sử dụng học sâu Q (deep Q-learning) vì không có kiến thức trước về các hành động đúng để chọn. Trong cấu trúc đề xuất, mỗi phần tử của vt đại diện cho một giá trị hành động như được định nghĩa trong [30]. Do đó, hàm mất mát có thể được viết , trong đó

Ở đây, = 0.99 là một hệ số giảm. Ngoài ra, chúng tôi sử dụng một mạng mục tiêu *f’ag* để ổn định quá trình đào tạo, đó là một bản sao của *fag* và cập nhật các tham số của nó sau mỗi *C* bước trong quá trình đào tạo. Trong công thức trên, *vt+1,a’* được xuất phát từ *f* và từ *fag*. Trong quá trình đào tạo, các tập tin được lựa chọn ngẫu nhiên từ bộ nhớ phát lại, và có hai chiến lược cập nhật được đề xuất trong [12]: ‘cập nhật ngẫu nhiên’ liên quan đến việc cập nhật từ một điểm ngẫu nhiên trong mỗi tập tin và tiến hành một số bước cố định, trong khi 'cập nhật tuần tự' đề cập đến việc cập nhật tất cả các tham số từ đầu đến cuối mỗi tập tin. Cả hai chiến lược cập nhật đã cho thấy hiệu suất tương tự nhau [12]. Vì chuỗi công cụ của chúng tôi không quá dài, chúng tôi đơn giản chỉ áp dụng 'cập nhật tuần tự', trong đó mỗi chuỗi đào tạo chứa một chuỗi công cụ hoàn chỉnh.

**Joint Training.** Như đã thảo luận ở *Mục 3.1*, không có một công cụ nào có thể hoàn toàn xử lý trạng thái trung gian, nơi các hiện tượng mới và phức tạp có thể được giới thiệu trong các bước khôi phục trước đó. Để giải quyết vấn đề này, chúng tôi đề xuất một thuật toán huấn luyện chung, như được hiển thị trong ***Thuật toán 1***, để huấn luyện các công cụ theo một cách end-to-end để tất cả các công cụ có thể học cách xử lý trạng thái trung gian. Cụ thể, đối với mỗi chuỗi công cụ trong một lô, hình ảnh bị méo ***I****1* được chuyển tiếp để nhận kết quả đã được khôi phục ***I****T +1*. Cho một mất mát MSE cuối cùng, các độ dốc sau đó truyền ngược theo chuỗi công cụ tương tự. Trong khi đó, các độ dốc của mỗi công cụ được tích lũy trong một lô và cuối cùng trung bình của độ dốc được sử dụng để cập nhật công cụ tương ứng. Quá trình cập nhật trên được tiến hành lặp lại trong vài vòng.

**Implementation Details.** Trong triển khai của chúng tôi, việc huấn luyện các công cụ tương tự như trong [19], trong đó tất cả các thí nghiệm chạy qua 80 epoch (3.2 × 105 lần lặp) với kích thước lô là 64. Tốc độ học ban đầu là 0.1 và giảm đi một lần gấp 0.1 sau mỗi 20 epoch. Đối với huấn luyện chung, chúng tôi đặt *M = 64, α = 0.0001* trong ***Thuật toán 1***, đại diện cho kích thước lô và tốc độ học tương ứng. Huấn luyện chung chạy qua 2 × 105 lần lặp. Trong khi huấn luyện tác nhân, chúng tôi sử dụng bộ tối ưu hóa Adam [21] và kích thước lô là 32. Bước tối đa *Tmax* được đặt là 3 một cách kinh nghiệm và kích thước bộ nhớ phát lại được chọn là 5 × 105. Tần suất cập nhật *C = 2,500* để mạng mục tiêu *f’ag* được sao chép từ mạng tác nhân mới nhất *fag* mỗi 2,500 lần lặp. Tốc độ học giảm theo cấp số nhân từ 2.5 × 10-4 đến 2.5 × 10-5 trong 5 × 105 lần lặp.

A white text with black text

Description automatically generated