Report paper: VECTOR-QUANTIZED IMAGE MODELING WITH IM- PROVED VQGAN

Giáp Ngọc Hiệu

**Tóm tắt nội dung**

Bài report này là phần tóm tắt lại ý tưởng, nội dung chính, những điều làm được của paper

VECTOR-QUANTIZED IMAGE MODELING WITH IM- PROVED VQGAN

1. **Ý tưởng**

Vector-quantized Image Modeling (VIM), được lấy cảm hứng từ thành công của việc tiền huấn luyện các mô hình ngôn ngữ để dự đoán từ tiếp theo trên các tập văn bản lớn. Trong phương pháp này, một mô hình Transformer được tiền huấn luyện để dự đoán các token hình ảnh được raster hóa theo cách tự hồi quy. Các token hình ảnh rời rạc này được xuất phát từ một mạng ViT-VQGAN (Vision-Transformer-based Vector Quantized Generative Adversarial Network).

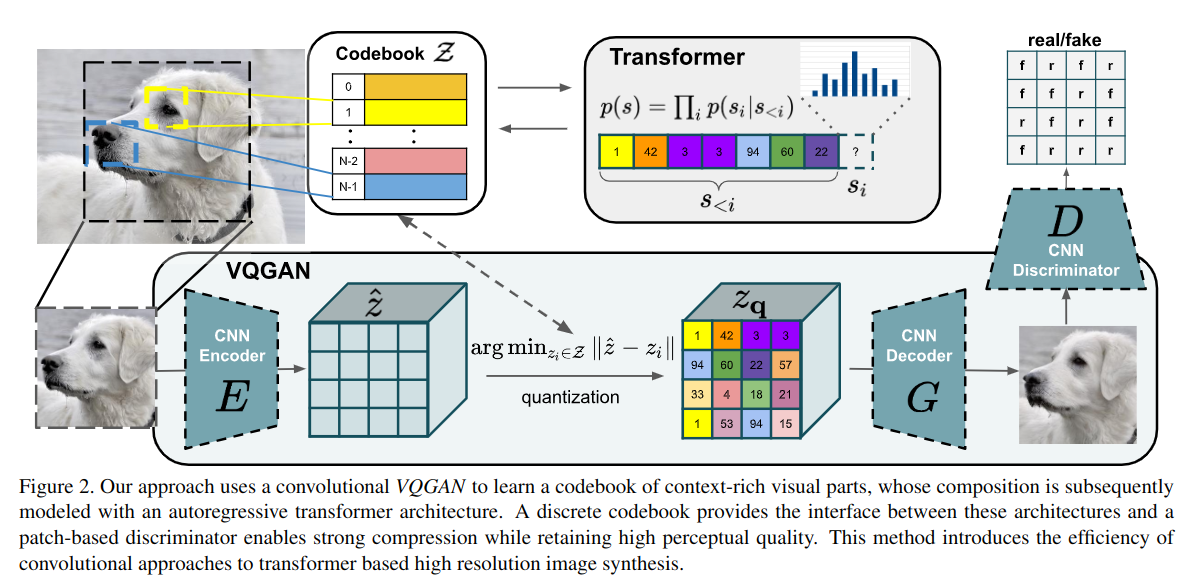
Tác giả đề xuất một số cải tiến cho mô hình VQGAN truyền thống, cải thiện kiến ​​trúc và quá trình học của bảng mã. Những cải tiến này dẫn đến hiệu suất tốt hơn và độ tin cậy cao hơn trong việc tái tạo hình ảnh. ViT-VQGAN được cải thiện được sử dụng cho các nhiệm vụ liên quan đến việc mô hình hóa hình ảnh theo phương pháp vector-quantized, như tạo hình ảnh không điều kiện và có điều kiện theo lớp, cũng như học biểu diễn không giám sát.

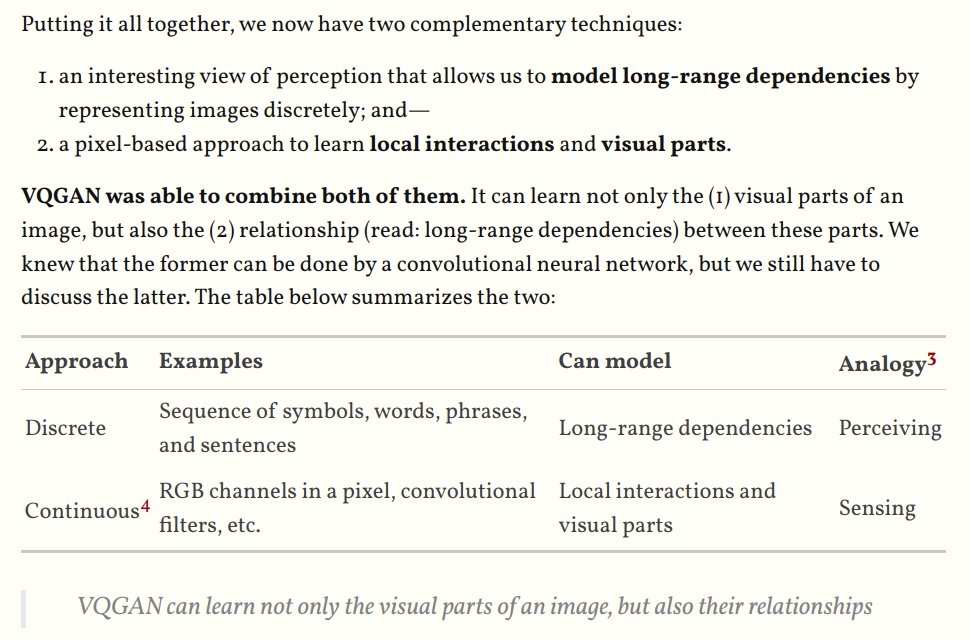
Khi được huấn luyện trên tập dữ liệu ImageNet với độ phân giải 256 × 256, tác giả báo cáo các cải thiện đáng kể về các chỉ số đánh giá so với VQGAN thông thường. Cụ thể, họ đạt được Inception Score (IS) là 175.1 và Fréchet Inception Distance (FID) là 4.17, đây là các điểm số đáng kể hơn so với các điểm số thu được bởi VQGAN thông thường.

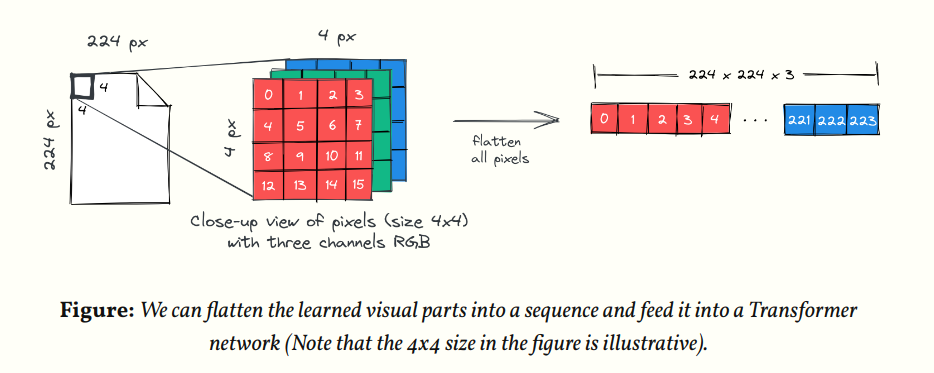
Hơn nữa, mô hình Transformer được tiền huấn luyện, dựa trên ViT-VQGAN và tiền huấn luyện không giám sát, được đánh giá bằng cách lấy trung bình các đặc trưng trung gian, giống như phương pháp được sử dụng trong Image GPT (iGPT). Mô hình VIM-L tiền huấn luyện này vượt xa iGPT-L trong mức độ chính xác của việc sử dụng linear-probe, cải thiện từ 60.3% lên 73.2%, mặc dù có cùng kích thước mô hình. Ngoài ra, VIM-L cũng vượt qua iGPT-XL, mà được huấn luyện trên dữ liệu hình ảnh web bổ sung và có kích thước lớn hơn.

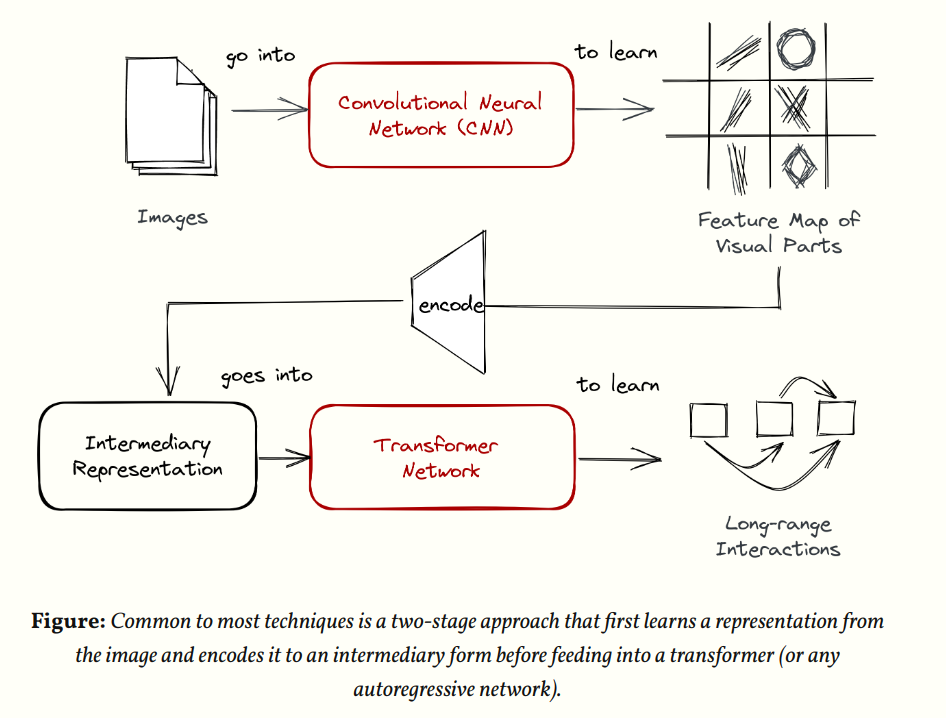
1. **Một số kiến thức cần thiết**

**2.1 VQGAN**

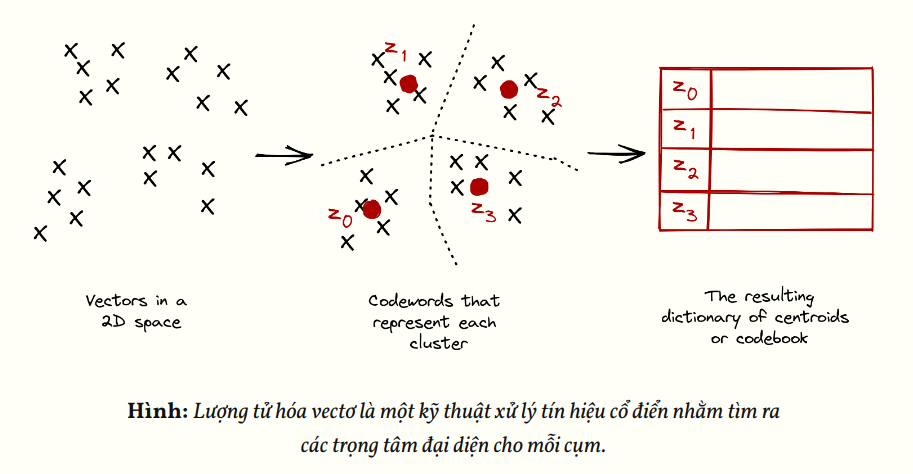
****





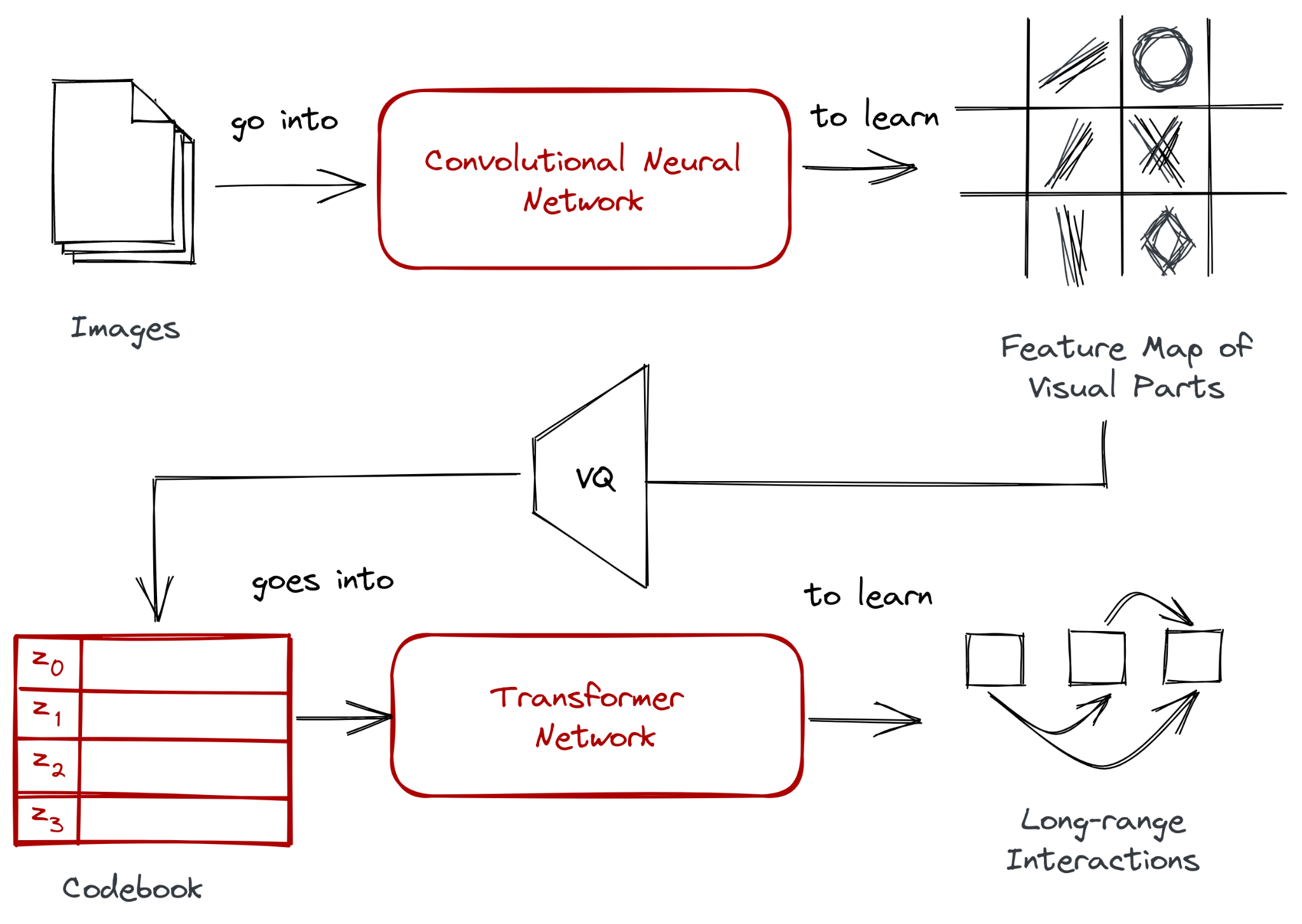


Trong VQGAN, có một cấu trúc hai giai đoạn tương tự nhưng khác biệt so với các phương pháp truyền thống khác. Thay vì giảm kích thước ảnh, VQGAN sử dụng một codebook để đại diện cho các phần của hình ảnh. Thay vì mô hình hóa ảnh từ mức độ pixel trực tiếp, VQGAN sử dụng các từ mã từ codebook đã học để biểu diễn hình ảnh.



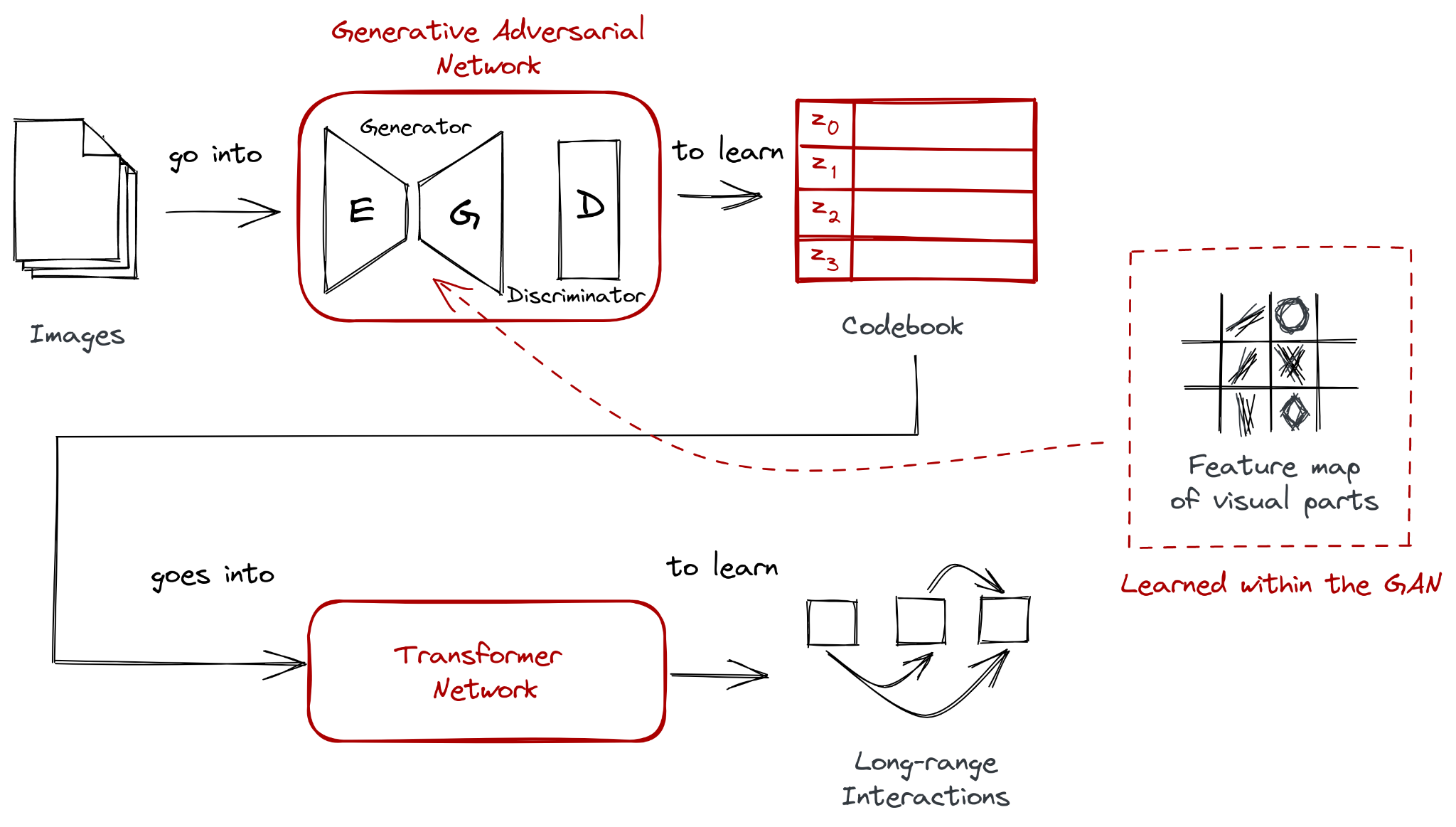
Kiến trúc của VQGAN bao gồm ba thành phần chính:

* Mạng nơ-ron tích chập (Convolutional Neural Network - CNN): Mạng này được sử dụng để xử lý tập hợp các hình ảnh và học các phần tương ứng của chúng. CNN tận dụng các tương tác cục bộ của pixel để học các đặc điểm cấp cao và xây dựng hình ảnh mới. Bằng cách này, thông qua việc chia nhỏ ảnh thành các phần nhỏ hơn và xử lý chúng, CNN có thể học được các đặc trưng phức tạp của hình ảnh.
* Mạng transformer: Sau khi một tập hợp các phần hình ảnh đã được học thông qua CNN, mạng transformer được sử dụng để xử lý chuỗi các biểu diễn của các phần đó. Mạng transformer làm cho việc hiểu các mối quan hệ xa hơn giữa các phần hình ảnh trở nên dễ dàng hơn. Với một biểu diễn rời rạc của các phần hình ảnh, transformer cho phép chúng ta hiểu các mối quan hệ giữa các phần này trên toàn bộ hình ảnh.
* Codebook (bảng mã): Codebook được tạo ra thông qua quá trình vector quantization. Nó bao gồm các từ mã rời rạc được sử dụng để biểu diễn các phần của hình ảnh. Việc sử dụng codebook giúp dễ dàng huấn luyện một mạng transformer trên đó, vì mạng transformer chỉ cần xử lý các biểu diễn rời rạc của các phần hình ảnh từ codebook, thay vì xử lý thông tin pixel trực tiếp. Điều này làm cho việc xây dựng mạng và huấn luyện mạng trở nên hiệu quả hơn.



Có một số điều chỉnh nhỏ sẽ được thực hiện trong kiến trúc của VQGAN:

* Sử dụng mạng sinh đối địch (Generative Adversarial Network - GAN): Thay vì sử dụng mạng nơ-ron tích chập đơn giản, VQGAN sẽ sử dụng một mạng sinh đối địch. Mạng GAN này sẽ thực hiện các phép tích chập tương tự nhưng với một cách tiếp cận GAN, điều này cho phép tạo ra các phần hình ảnh đa dạng hơn và phức tạp hơn. GAN sẽ cố gắng học cách tạo ra hình ảnh mới mà không chỉ đơn thuần là học các đặc điểm cấp cao từ dữ liệu đầu vào. 2
* Học codebook ngay từ đầu: Thay vì có quá trình riêng biệt cho vector quantization, VQGAN sẽ học codebook ngay từ đầu. Việc học bản đồ đặc trưng của các phần hình ảnh sẽ diễn ra bên trong mạng GAN. Với điều này, VQGAN vẫn duy trì cách tiếp cận hai giai đoạn như trước, nhưng thay vì phải chia làm hai giai đoạn riêng biệt, quá trình học codebook và học bản đồ đặc trưng sẽ được thực hiện đồng thời trong một quá trình huấn luyện duy nhất.
* Những điều chỉnh này sẽ giúp cải thiện hiệu suất và tính linh hoạt của kiến trúc VQGAN trong việc tạo ra và mô hình hóa hình ảnh.



**Training VQGAN**

**Training the GAN**

****

Nếu ta xem xét chi tiết cấu trúc của bộ sinh của VQGAN, ta sẽ thấy nó tuân theo một kiến trúc mã hóa-giải mã. Bố cục này gợi nhớ đến autoencoder, trong đó mục tiêu là có một giải mã G có thể tái tạo đúng đắn đầu vào x. Lý thuyết đằng sau là nếu ta có sự tái tạo hoàn hảo, điều đó có nghĩa là bộ mã hóa E đã tìm ra một biểu diễn phù hợp của dữ liệu.

Trong kiến trúc mã hóa-giải mã, bộ mã hóa E được sử dụng để ánh xạ đầu vào x sang một không gian biểu diễn nén. Điều này có thể hiểu là E nén thông tin của x vào một biểu diễn nén chiều thấp. Sau đó, bộ giải mã G nhận biểu diễn nén này và cố gắng tái tạo lại đầu vào x .

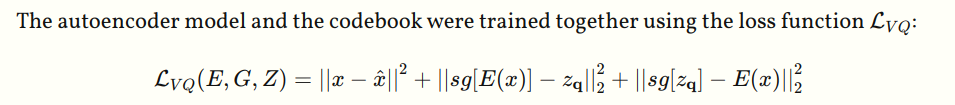
Nếu G có thể tái tạo x một cách chính xác, điều đó chỉ ra rằng E đã học được một biểu diễn phù hợp của x.

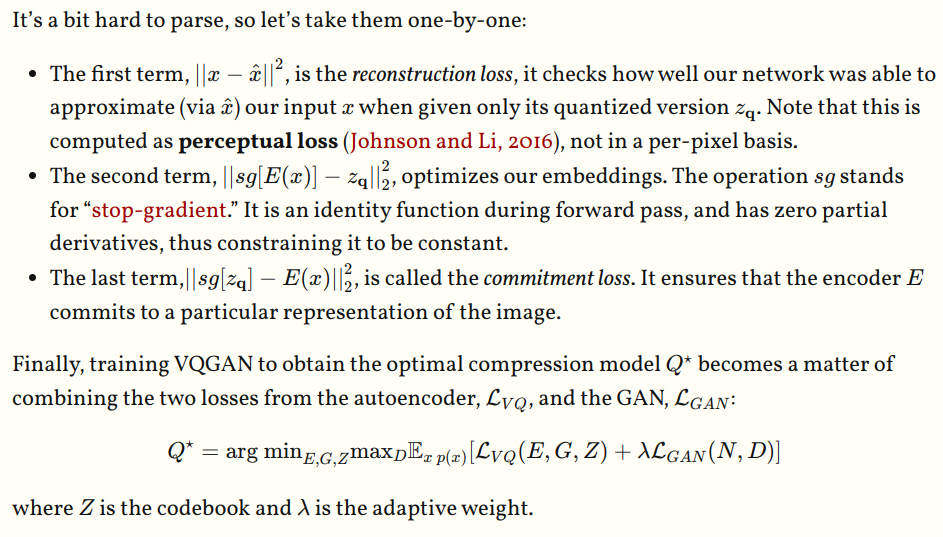
Sau khi x được mã hóa thành z, tức là z ^ =E(x), quá trình vector quantization cũng diễn ra giữa các mạng mã hóa và giải mã. Sau khi có được biểu diễn nén z ^ , ta thực hiện element-wise operation q để thu được phiên bản rời rạc của đầu vào:



Vì vậy, thay vì xây dựng lại từ đầu ra bộ mã hóa z^ , chúng ta thực hiện điều đó từ dạng lượng tử hóa của nó z q. Vì vậy, hình ảnh được tái tạo x^ ≈ x trông như thế này:

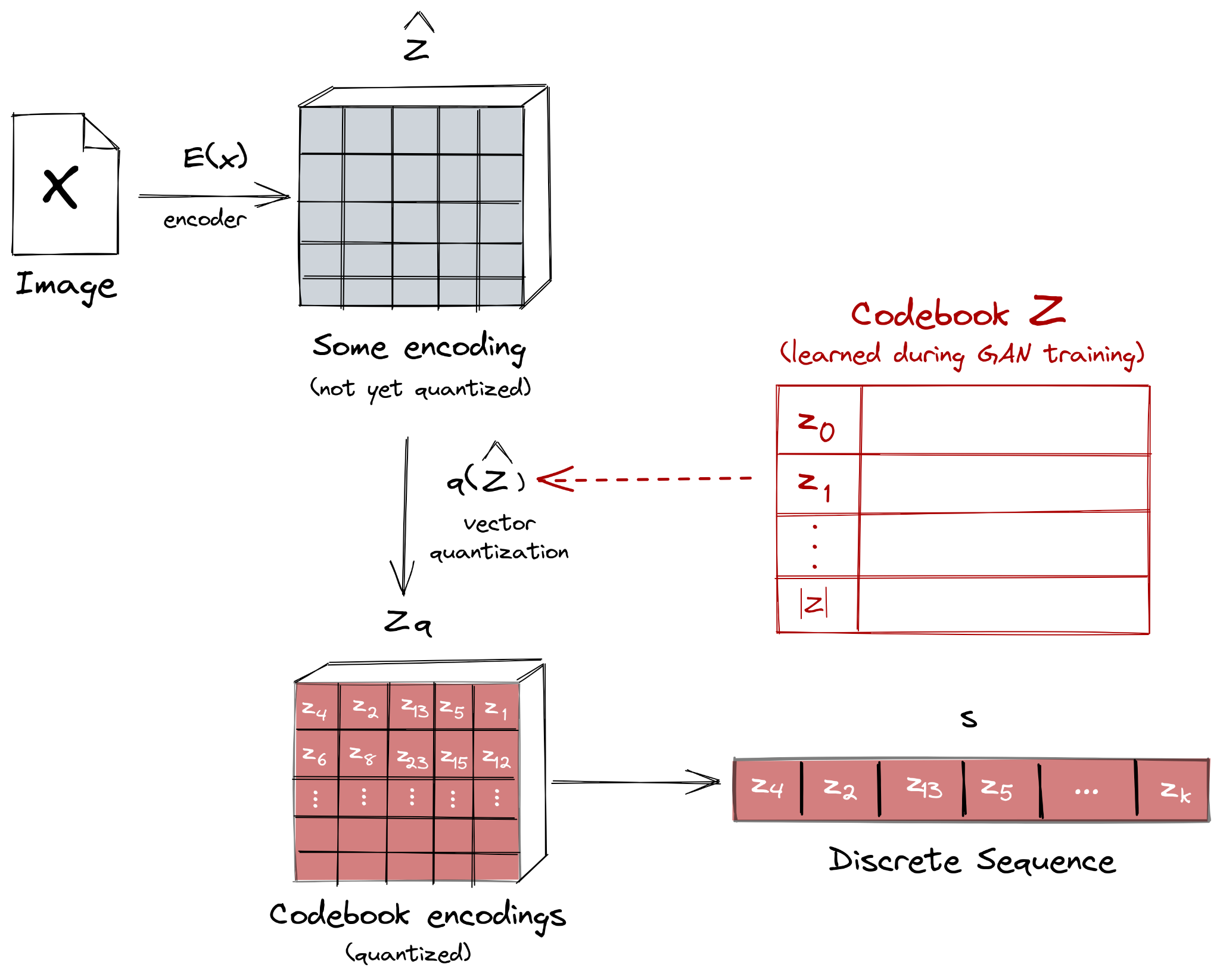






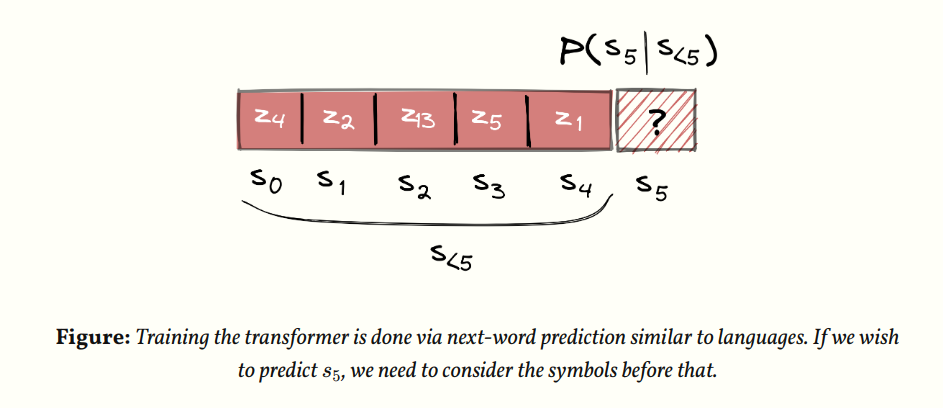
Khi kết hợp với nhau, chúng ta có được khả năng riêng biệt của VQ, khả năng biểu đạt phong phú của GAN và khả năng mã hóa của bộ mã hóa tự động. Điều này cho phép chúng ta thu được các phần hình ảnh phong phú hơn và khác biệt hơn so với mạng nơ-ron tích chập tiêu chuẩn.

**Training the transformer**

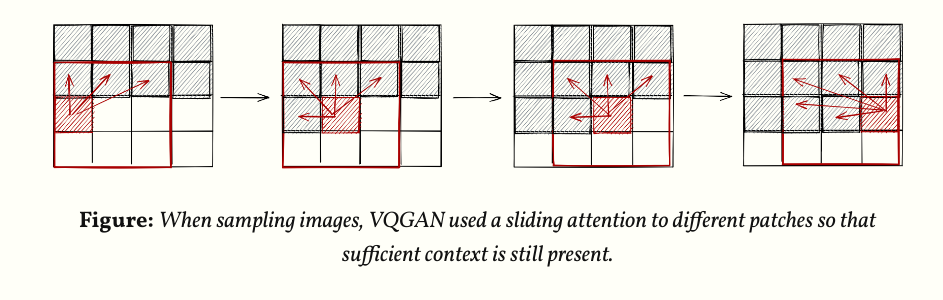


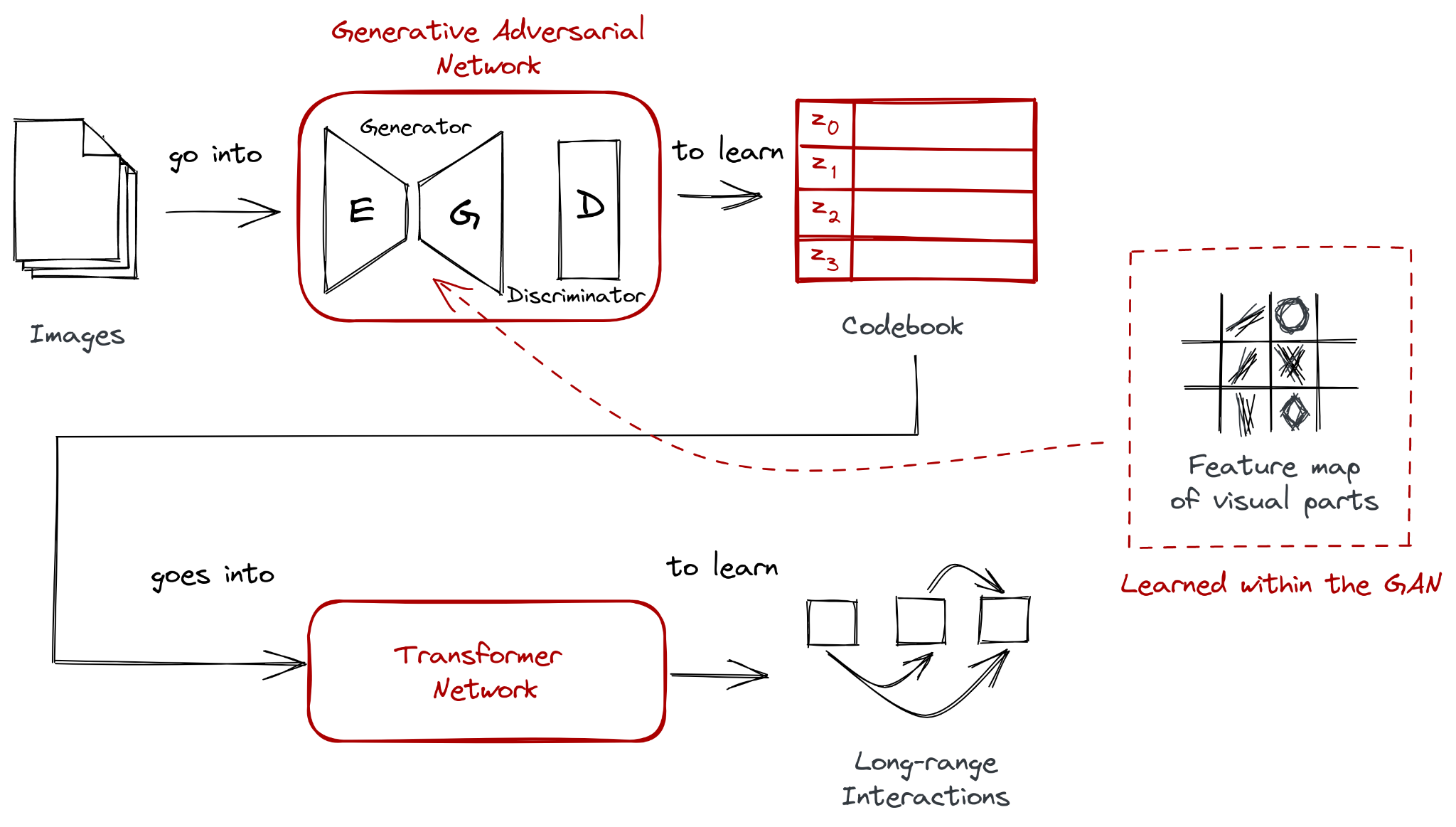
Chúng ta chỉ quan tâm đến codebook Z vì nó cung cấp một biểu diễn rời rạc của các phần hình ảnh của chúng ta có thể được dễ dàng đưa vào một transformer. Để làm điều này, chúng ta biểu diễn các hình ảnh dưới dạng các chỉ số trong codebook của các nhúng của chúng. Điều này có nghĩa là việc mã hóa z\_q của một hình ảnh x có thể được biểu diễn dưới dạng một chuỗi s từ codebook Z , tức là s∈{0,…,∣Z∣−1}.

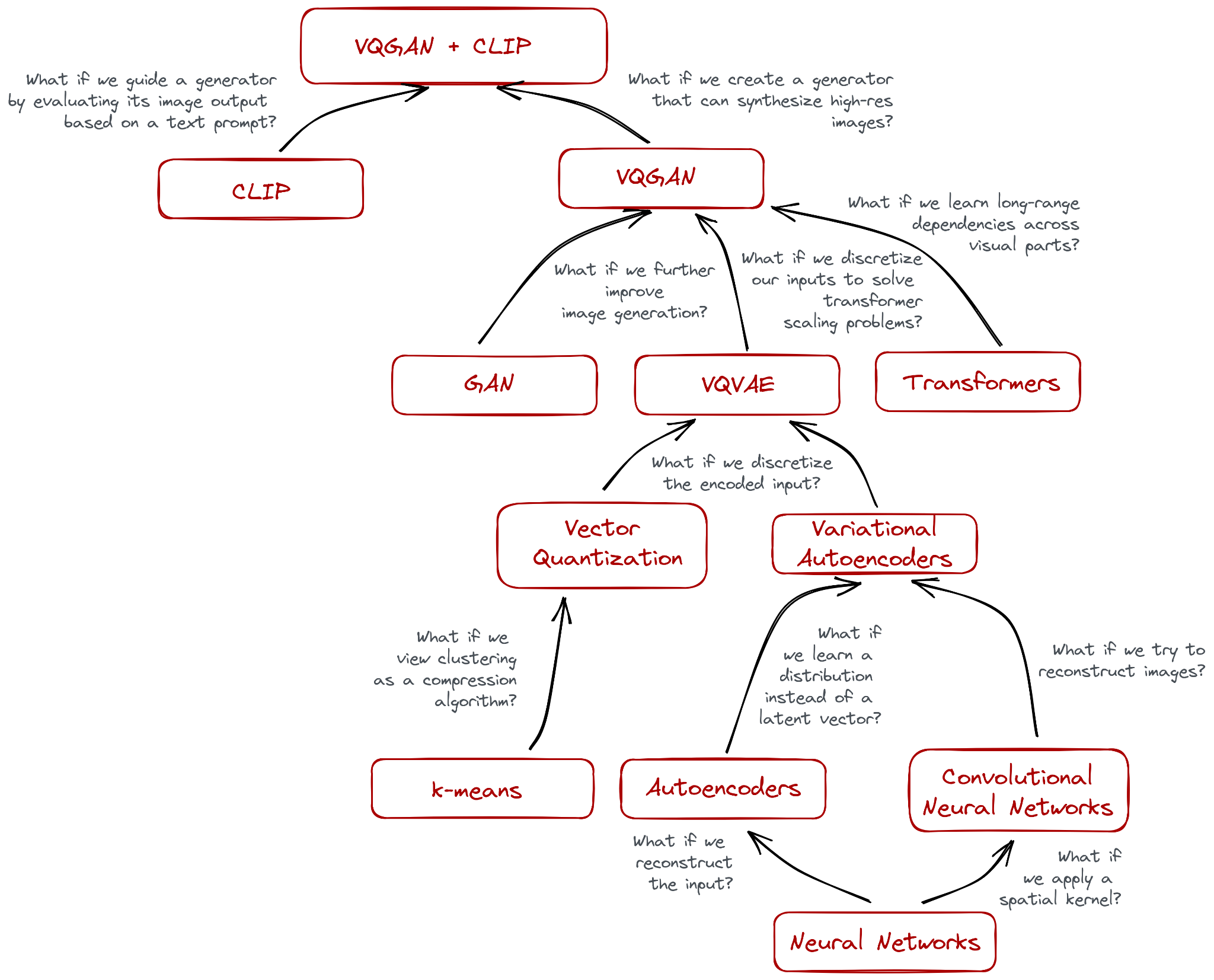
Trong biểu diễn này, mỗi chỉ số trong chuỗi s tương ứng với một từ mã trong codebook Z . Bằng cách này, chúng ta có thể biểu diễn mỗi hình ảnh dưới dạng một chuỗi các chỉ số, mà mỗi chỉ số này chỉ ra từ mã trong codebook mà đại diện cho một phần cụ thể của hình ảnh. Điều này giúp chúng ta dễ dàng chuyển đổi hình ảnh thành dữ liệu đầu vào phù hợp cho một mạng transformer để xử lý tiếp.

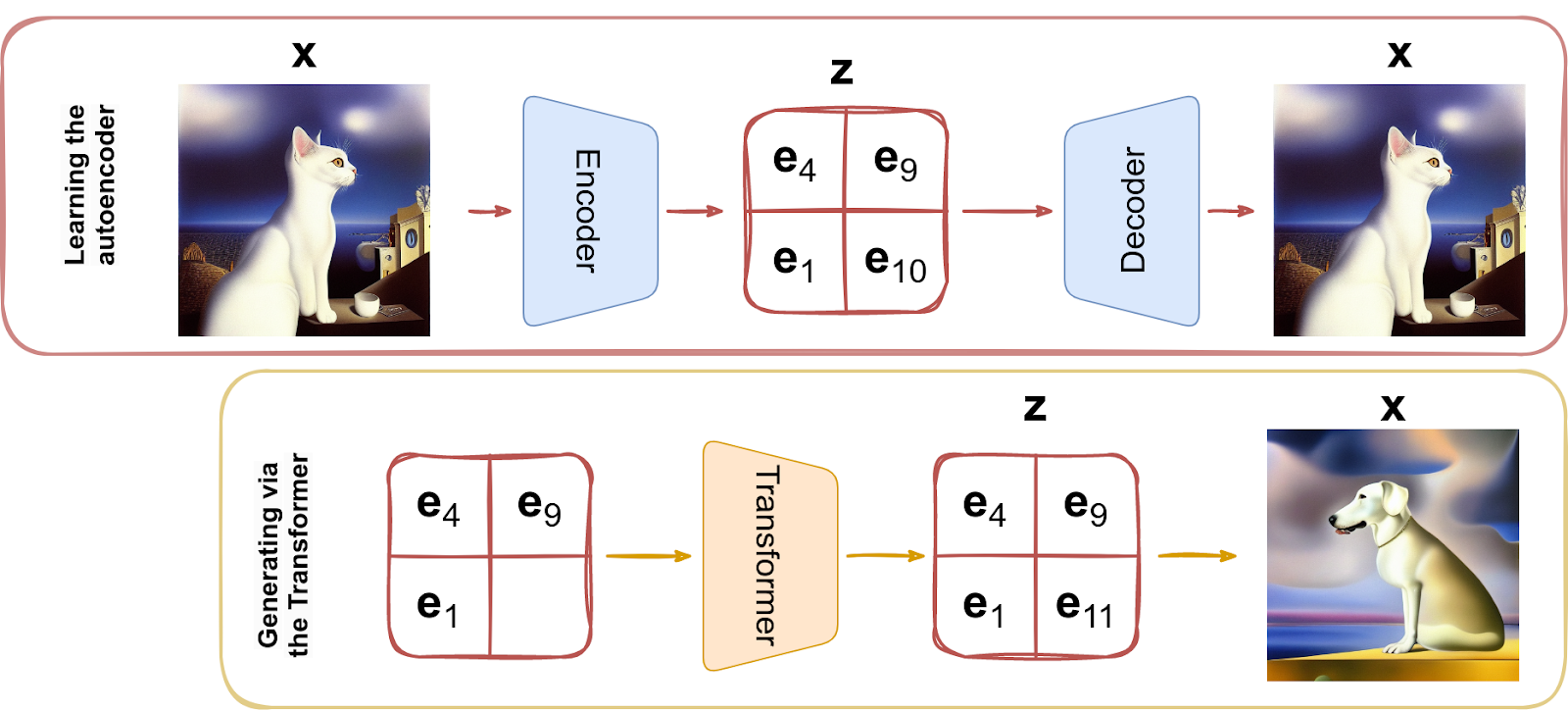


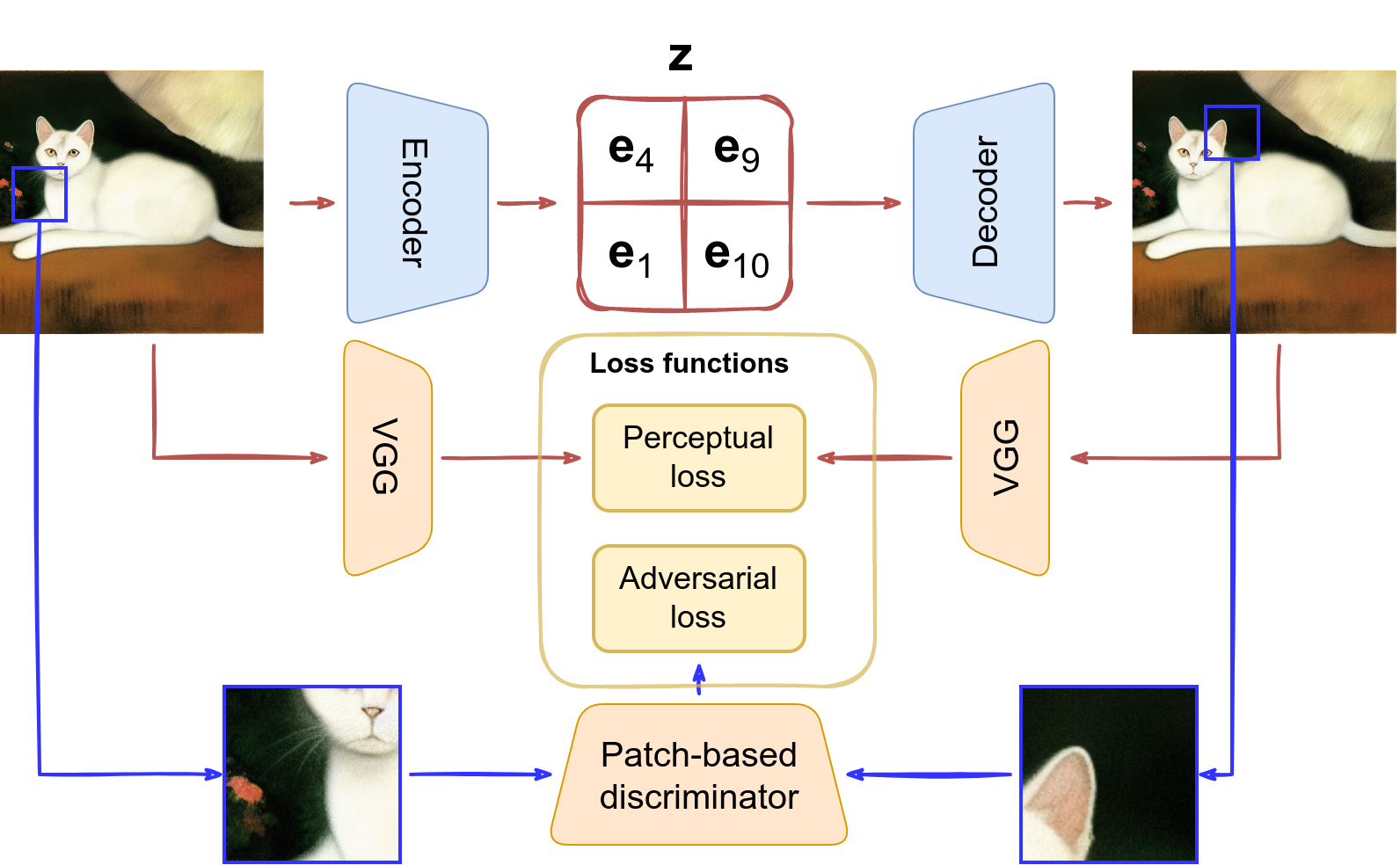
Cuối cùng, khi tạo hình ảnh có độ phân giải cao, tác giả đã giới hạn bối cảnh thông qua cửa sổ trượt. Điều này có nghĩa là khi tạo mỗi bản vá, nó chỉ lấy thông tin từ các bản vá lân cận. Đó là một “thủ thuật” hay để cải thiện hiệu quả sử dụng tài nguyên khi sử dụng transformer.



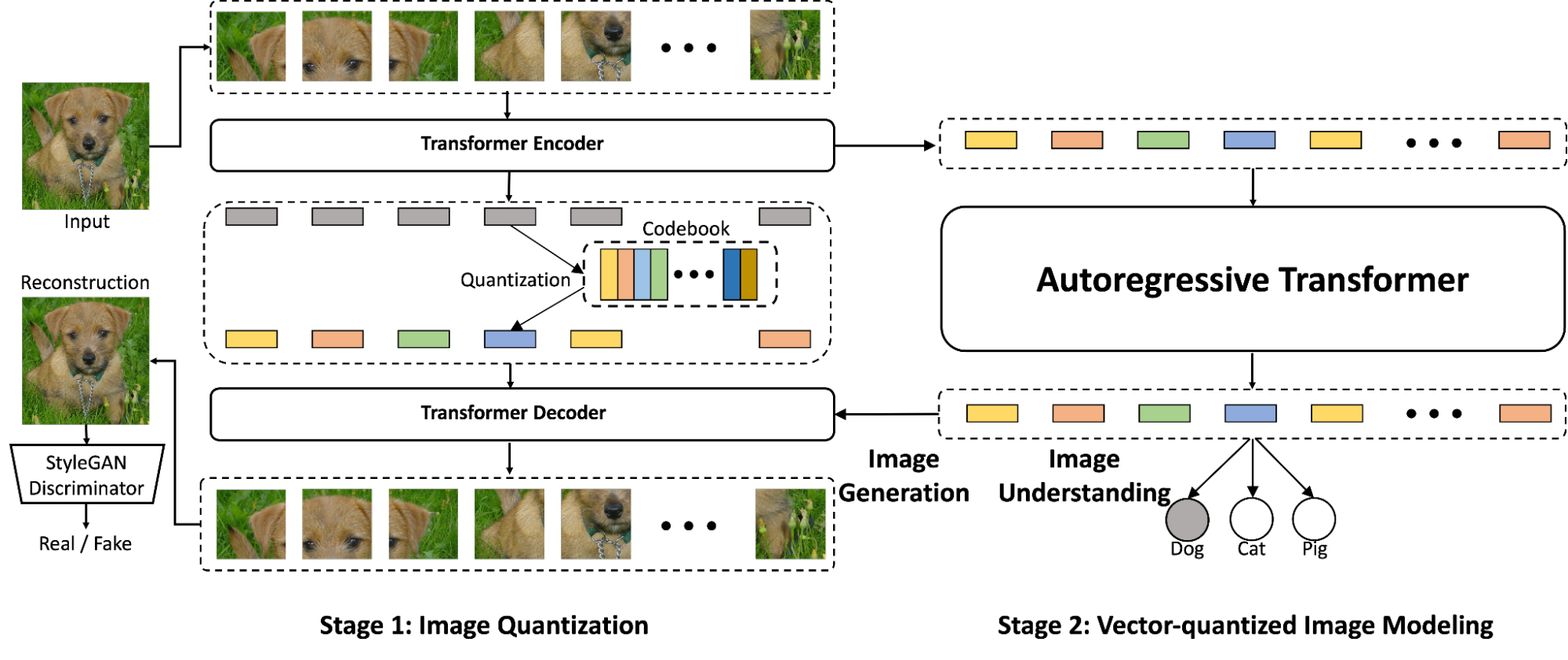








1. **VECTOR-QUANTIZED IMAGES WITH VIT-VQGAN**

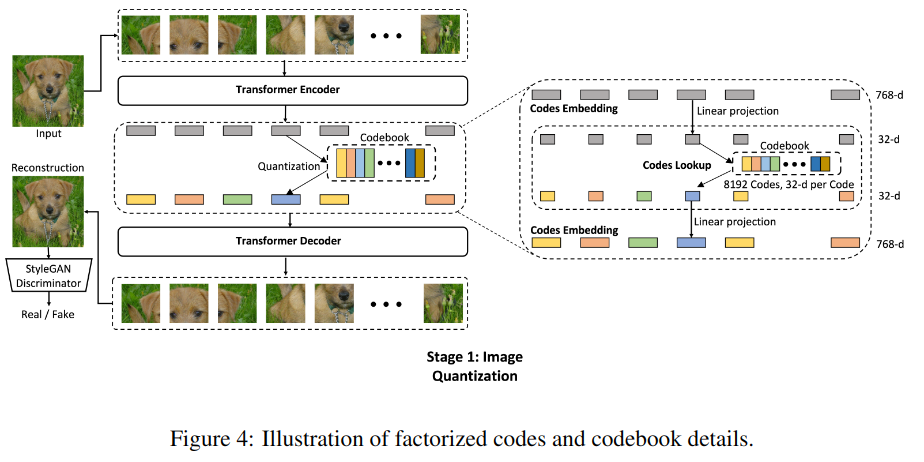
****

Tác giả khám phá một phương pháp gọi là Vector-quantized Image Modeling (VIM) và áp dụng nó vào cả các nhiệm vụ sinh hình ảnh và hiểu hình ảnh. VIM tuân theo một phương pháp hai giai đoạn:

* Giai đoạn 1: Lượng tử hóa Ảnh: Cho một hình ảnh có độ phân giải là 256×256, một mô hình VQGAN dựa trên Vision Transformer mã hóa nó thành các mã latente rời rạc kích thước 32×32, với kích thước của bảng mã là 8192. Tác giả đề xuất nhiều cải tiến từ kiến trúc đến việc học bảng mã cho VQGAN. Kết quả thu được là ViT-VQGAN hiệu quả hơn và cải thiện độ chính xác của việc tái tạo theo từng pixel, Inception Score (IS) và Fréchet Inception Distance (FID). ViT-VQGAN được huấn luyện end-to-end trên dữ liệu chỉ có hình ảnh với các hàm mục tiêu kết hợp bao gồm mất mát logit-laplace, mất mát L2, adversarial loss và perceptual loss.
* Giai đoạn 2: Vector-quantized Image Modeling: Huấn luyện một mô hình Transformer để dự đoán các token hình ảnh được raster hóa 32×32 = 1024 theo cách tự sinh một cách độc lập, trong đó các token hình ảnh được mã hóa bởi một mô hình ViT-VQGAN đã học ở Giai đoạn 1. Đối với việc tổng hợp hình ảnh không điều kiện hoặc học không giám sát, tiền huấn luyện một mô hình Transformer chỉ có decoder để dự đoán token tiếp theo. Đối với việc tổng hợp hình ảnh có điều kiện lớp, một token ID class được thêm vào trước các token hình ảnh. Để đánh giá chất lượng của việc học không giám sát, tác giả lấy trung bình các đặc trưng trung gian của Transformer và học một phần đầu tuyến tính để dự đoán logit của các lớp (còn gọi là linear-probe).
  1. **VQGAN WITH VISION TRANSFORMERS**
* Kiến trúc mạng lõi được cả VQVAE và VQGAN sử dụng để mã hóa và tái tạo hình ảnh là CNN. VQGAN giới thiệu các phần tử giống như transformer ở dạng non-local attention block, cho phép nó nắm bắt các tương tác ở xa với ít lớp hơn. Tác giả đề xuất thực hiện phương pháp này thêm một bước nữa bằng cách thay thế bộ mã hóa và giải mã CNN bằng Vision Transformer (ViT). Khi có đủ dữ liệu (có rất nhiều dữ liệu hình ảnh không được gắn nhãn) chúng tôi thấy rằng ViT-VQGAN ít bị hạn chế hơn bởi các nguyên tắc quy nạp do tích chập áp đặt. Hơn nữa, ViT-VQGAN mang lại hiệu quả tính toán tốt hơn trên máy gia tốc và tạo ra các bản tái tạo chất lượng cao hơn, như trong Bảng.
* Bộ mã hóa của ViT-VQGAN trước tiên ánh xạ các 8x8 non-overlapping image patches thành image tokens, tiếp theo là các khối Transformer, mã hóa hình ảnh có độ phân giải 256×256 thành chuỗi token 32×32=1024. Bộ giải mã thực hiện thao tác ngược lại, ánh xạ từng token hình ảnh từ các biến tiềm ẩn trở lại các patch hình ảnh 8 × 8 và nhóm lại chúng thành hình ảnh 256 × 256. Ở đầu ra của các khối transformer, chúng ta áp dụng two-layer feed-forward network với lớp kích hoạt tanh ở giữa. Không có activation nào được áp dụng ở đầu ra của bộ mã hóa hoặc bộ giải mã ViT-VQGAN (ngoại trừ dự đoán trung bình về tổn thất logit-laplace). Sigmoid được áp dụng cho giá trị trung bình dự đoán của bộ giải mã do loss logit-laplace. Tác giả thấy rằng phương pháp đơn giản này mang lại kết quả cao tái tạo chất lượng mà không có bất kỳ grid artifacts đáng chú ý nào.
  1. **CODEBOOK LEARNING**
* VQVAEs (Vector Quantized Variational Autoencoders) thường gặp vấn đề về việc sử dụng mã codebook thấp, chủ yếu do sự khởi tạo không đủ của codebook. Điều này có nghĩa là trong quá trình huấn luyện, một phần đáng kể các mã trong codebook vẫn không được sử dụng hoặc "chết". Sự giảm kích thước codebook hiệu quả dẫn đến việc tái tạo kém chất lượng trong giai đoạn lượng tử hóa ban đầu và góp phần vào việc thiếu đa dạng trong các giai đoạn sau của tổng hợp hình ảnh.
* Để giải quyết những vấn đề này, VQGAN sử dụng các heuristics lấy mẫu top-k và top-p (hạt nhân) với kích thước codebook mặc định là 1024 để đạt được kết quả tốt hơn trong tổng hợp hình ảnh. Những heuristics này giúp ưu tiên các mã quan trọng nhất từ codebook trong quá trình lấy mẫu.
* Các cải tiến được đề xuất nhằm mục đích tăng cường việc sử dụng codebook ngay cả với kích thước codebook lớn hơn là 8192. Cải tiến đầu tiên liên quan đến việc đơn giản hóa quá trình lấy mẫu trong quá trình tổng hợp hình ảnh bằng cách sử dụng phương pháp lấy mẫu trực tiếp với nhiệt độ là 1.0, mà không phụ thuộc vào các heuristics top-k và top-p. Điều này gợi ý rằng mỗi mã trong codebook có cơ hội bằng nhau được chọn trong quá trình lấy mẫu, từ đó khuyến khích việc sử dụng hiệu quả hơn của toàn bộ codebook.
* Mục tiêu huấn luyện của lượng tử hóa vector được xác định như sau:



* Ở đây, sg(x) ≡ x, sg(x) ≡ 0 là toán tử dừng gradient, β là siêu tham số commitment loss được đặt thành 0,25 trong tất cả các thử nghiệm và e là vectơ sách mã. Chỉ số sổ mã lượng tử hóa được xác định bằng cách tra cứu vectơ sổ mã gần nhất với các đặc điểm đầu vào ze(x) theo khoảng cách Euclide, thu được 
* **Factorized codes**: Chúng ta thực hiện một phép chiếu tuyến tính từ đầu ra của bộ mã hóa tới một không gian biến ẩn có số chiều thấp cho việc tìm kiếm chỉ mục mã (ví dụ, giảm từ một vector 768 chiều xuống còn một vector 32 hoặc 8 chiều cho mỗi mã), và thấy rằng điều này mang lại một sự tăng cường ngay lập tức về việc sử dụng codebook. Phép phân rã này có thể được xem xét như việc tách biệt việc tìm kiếm mã và việc nhúng mã: chúng ta tìm kiếm biến số gần nhất được mã hóa từ đầu vào trên một không gian tìm kiếm chiều thấp, sau đó chiếu mã biến số tìm được vào không gian nhúng chiều cao. Các thí nghiệm của chúng tôi cho thấy việc giảm chiều của không gian tìm kiếm từ 256-d xuống còn 32-d luôn cải thiện chất lượng tái tạo.



* **l2-normalized codes:** Chúng ta cũng áp dụng chuẩn hóa l2 trên các biến tiềm ẩn được mã hóa ze(x) và biến tiềm ẩn trong bảng mã e. Các biến sổ mã được khởi tạo từ phân bố chuẩn. Bằng cách ánh xạ tất cả các biến tiềm ẩn trên một hình cầu, khoảng cách Euclide của l2 biến tiềm ẩn được chuẩn hóa  tiến triển thành độ tương tự cosine của hai vectơ giữa ze(x) và e, cải thiện hơn nữa tính ổn định của quá trình đào tạo và chất lượng tái thiết được thể hiện trong các thử nghiệm của chúng ta.
  1. **VIT-VQGAN TRAINING LOSSES**
* Chúng ta sử dụng một kết hợp của các hàm mất mát logit-laplace, mất mát l2, perceptual loss (dựa trên mạng VGG) và loss GAN với kiến trúc của bộ phân biệt StyleGAN. Trọng số cân bằng tổn thất được cấu hình bằng cách quét siêu tham số để tối ưu hóa chất lượng tái tạo hình ảnh, mức sử dụng sổ mã, FID và IS. Sau đó, chúng ta áp dụng cùng một bộ siêu tham số của các hàm mất mát huấn luyện cho tất cả các tập dữ liệu bao gồm CelebA-HQ, FFHQ và ImageNet. Mất mát logit-laplace có thể được xem xét như mất mát l1 được chuẩn hóa, giả định rằng nhiễu ở cấp độ điểm ảnh được phân phối theo phân phối laplace trong khi mất mát l2 giả định rằng nhiễu được phân phối theo phân phối Gaussian. Tác giả thấy rằng loss logit-laplace đóng góp vào việc sử dụng codebook trong khi mất mát `l2 và perceptual loss đóng góp đáng kể vào FID. Kết hợp loss cuối cùng mà chúng ta sử dụng mặc định là:

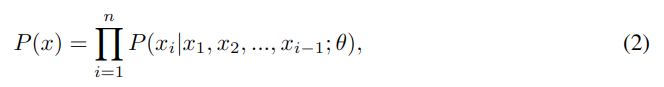


* Một điểm cần lưu ý về perceptual loss dựa trên VGG là mạng VGG được tiền huấn luyện với mất mát phân loại giám sát, vì vậy sự giám sát có thể rò rỉ vào Giai đoạn 2 để đo lường độ chính xác của linear-probe. Do đó, đối với tất cả các kết quả học không giám sát mà tác giả báo cáo, tác giả loại bỏ perceptual loss trong quá trình huấn luyện của ViT-VQGAN. Đối với tất cả các tổng hợp hình ảnh không điều kiện và có điều kiện lớp, tác giả sử dụng bộ mã hóa ViT-VQGAN được huấn luyện với perceptual loss, vì điều này dẫn đến việc tái tạo với độ chính xác cao hơn.

1. **VECTOR-QUANTIZED IMAGE MODELING**

Trong một mô hình ViT-VQGAN đã học, các hình ảnh được mã hóa thành các mã latents rời rạc được làm phẳng theo thứ tự raster, tương tự như Image GPT. Một mô hình Transformer chỉ có bộ giải mã được sử dụng để mô hình hóa mật độ dữ liệu hình ảnh

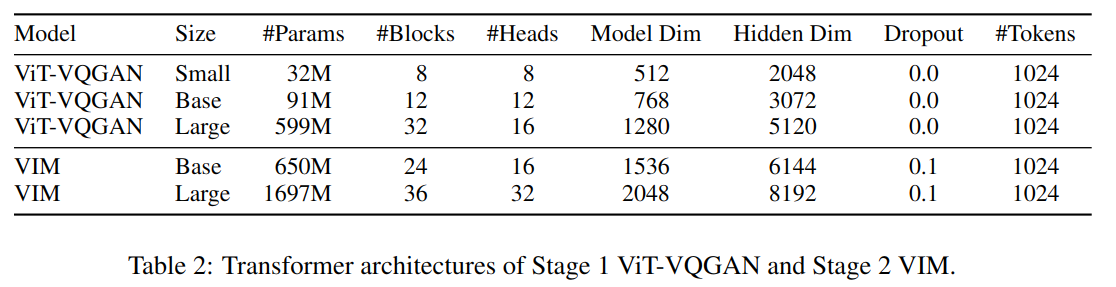
P (x) theo cách tự hồi quy như sau:



Trong đó θ là các trọng số có thể học được. Mục tiêu huấn luyện là tối thiểu hóa negative log-likelihood của dữ liệu:



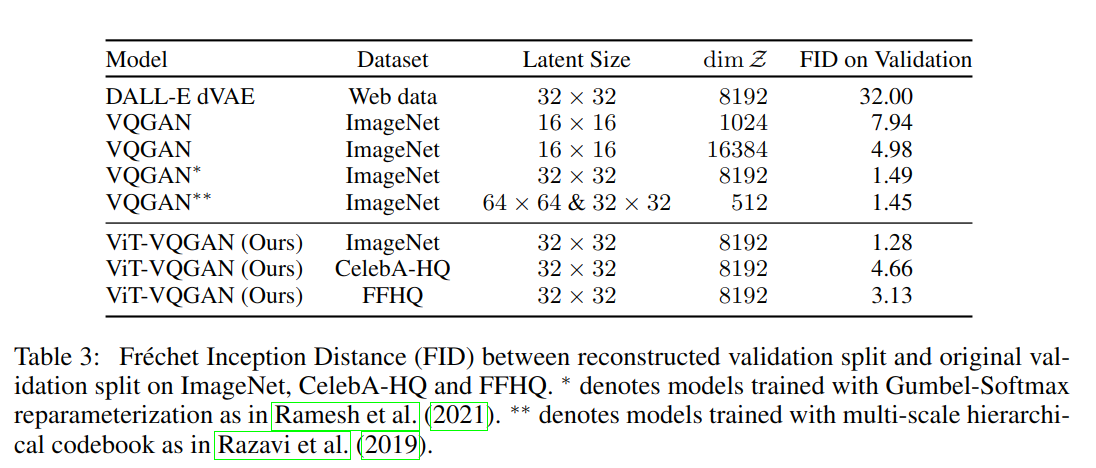
Bảng 2 tóm tắt cấu hình kiến trúc cho các Transformer. Đầu tiên, chúng ta nhúng các id token hình ảnh rời rạc vào không gian nhúng có thể học được tại mỗi vị trí, với một embedding vị trí 2D có thể học được được thêm vào. Cả hai chiều nhúng đều giống như kích thước của mô hình. Chúng ta áp dụng mộtstacl các khối Transformer cho các đầu vào với sự chú ý tương tác ở toàn bộ chuỗi. Tỉ lệ dropout là 0.1 cho tất cả các đầu ra của các khối residual, activation và attention. Ở lớp cuối cùng của tất cả các khối Transformer, chúng ta áp dụng một lớp normalization bổ sung.



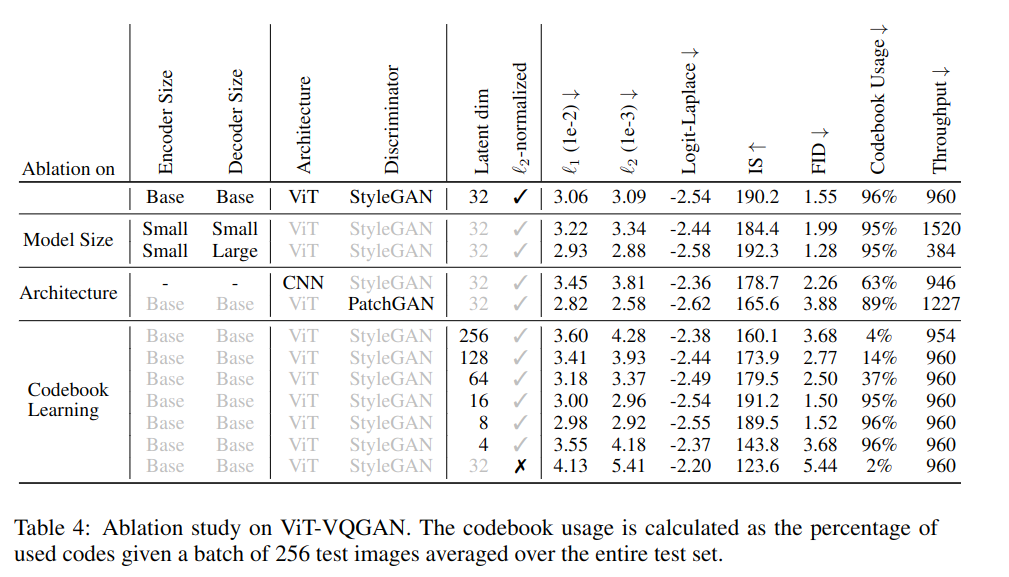
* 1. **IMAGE SYNTHESIS**
* Với một mô hình Transformer sinh được tiền huấn luyện, việc tạo hình ảnh không điều kiện được thực hiện bằng cách đơn giản lấy mẫu token từ phân phối softmax đầu ra. Tất cả các mẫu được sử dụng cho cả kết quả chất lượng và định lượng đều được thu được mà không giảm temperature. Các token được lấy mẫu sau đó được đưa vào bộ giải mã của ViT-VQGAN để giải mã hình ảnh đầu ra. Mặc định, Giai đoạn 1 của ViT-VQGAN mã hóa các hình ảnh đầu vào có độ phân giải 256 × 256 thành các mã latents 32 × 32 với kích thước codebook là 8192, trong khi Giai đoạn 2 của Transformer lấy các token hình ảnh được phẳng với tổng độ dài là 1024.
* Class-conditioned ImageNet generation cũng là một thước đo phổ biến được sử dụng để đo khả năng của các mô hình trong tổng hợp hình ảnh. Tác giả mở rộng việc tạo hình ảnh không điều kiện thành việc tạo hình ảnh có điều kiện lớp bằng cách thêm một token class-id trước các token hình ảnh. Các lớp embedding riêng biệt được học từ đầu cho token class-id và các token hình ảnh, với kích thước nhúng giống như kích thước của mô hình Transformer. Trong quá trình lấy mẫu, một token class-id được cung cấp tại vị trí đầu tiên để giải mã các token hình ảnh còn lại theo cách tự hồi quy.
  1. **UNSUPERVISED LEARNING**
* Đối với nhiệm vụ hiểu hình ảnh, chúng ta đưa tất cả các token hình ảnh của đầu vào vào một Transformer được tiền huấn luyện, và nhận được một chuỗi các đặc trưng token có tổng cộng là 1024. Tương tự như trong Image GPT, chúng ta chọn một đầu ra của lớp tại một khối cụ thể l trong tổng số khối L, tính trung bình trên chuỗi các đặc trưng token (được đóng băng) và chèn một lớp softmax (có thể học được) để chiếu các đặc trưng trung bình thành các class logit. Chúng ta chỉ lấy một đầu ra cụ thể của khối Transformer thay vì nối các đầu ra khác nhau của các khối như trong iGPT. Tác giả thấy rằng most discriminating feature cho việc linear-probe thường nằm gần giữa tất cả các khối Transformer.

1. **Experimental & Results**
   1. **IMAGE QUANTIZATION**

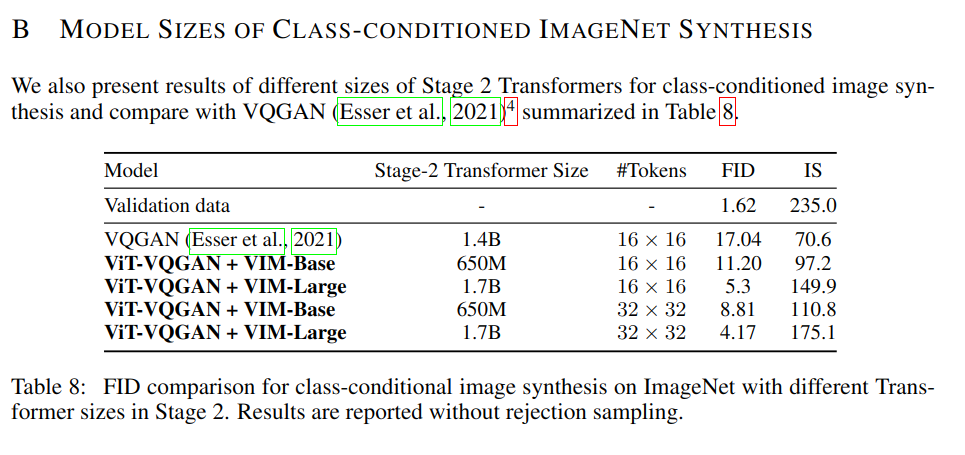
* Tác giả huấn luyện mô hình đề xuất ViT-VQGAN trên ba bộ dữ liệu riêng biệt, gồm CelebA-HQ, FFHQ và ImageNet. Đối với CelebA-HQ và FFHQ, phân chia mặc định giữa tập huấn luyện và tập xác thực như trong VQGAN. Đối với giai đoạn 1 của việc lượng tử hóa hình ảnh, thử nghiệm ba kích thước kiến trúc khác nhau, gồm Small, Base và Large cho cả bộ mã hóa và giải mã, như được định nghĩa trong Bảng 2. Mô hình nhỏ nhất ViT-VQGAN-SS có bộ mã hóa kích thước nhỏ và giải mã kích thước nhỏ, trong khi ViT-VQGAN-BB có bộ mã hóa kích thước trung bình và giải mã kích thước trung bình. Mô hình lớn nhất ViT-VQGAN-SL có một bộ mã hóa kích thước nhỏ không đối xứng và giải mã kích thước lớn, với động cơ là việc huấn luyện giai đoạn 2 chỉ yêu cầu truyền tiến của bộ mã hóa của ViT-VQGAN (trong quá trình suy luận/giải mã cho tổng hợp hình ảnh, vẫn cần giải mã của ViT-VQGAN để giải mã hình ảnh từ các mã dự đoán trong giai đoạn 2).



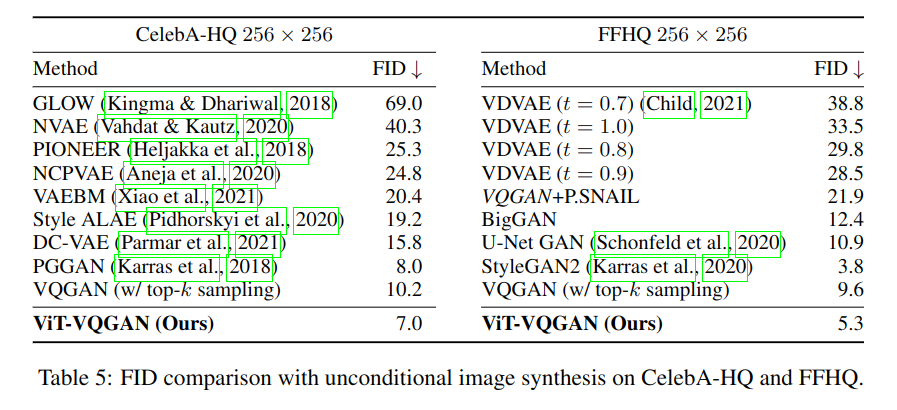
* Bảng 4 cung cấp các phân tích mô phỏng chi tiết về các sự điều chỉnh được đề xuất, với kết quả thực nghiệm về khoảng cách trung bình l1, khoảng cách l2, khoảng cách logit-laplace, Inception Score và FID trên ImageNet. Trong số các kích thước mô hình khác nhau, ViT-VQGAN-SS (bộ mã hóa nhỏ, bộ giải mã nhỏ) thực hiện kém hơn so với ViT-VQGAN-BB (bộ mã hóa cơ bản, bộ giải mã cơ bản) và ViT-VQGAN-SL (bộ mã hóa nhỏ, bộ giải mã lớn), nhưng đạt được hiệu suất qua lại nhanh hơn nhiều. Kiến trúc VQGAN dựa trên CNN tồi tệ cả về chất lượng và hiệu suất so với VQGAN dựa trên ViT. Bộ phân biệt dựa trên StyleGAN (Karras et al., 2019) ổn định hơn và mang lại chất lượng tái tạo tốt hơn so với PatchGAN (Isola et al., 2017) (được sử dụng cho VQGAN). Đối với việc học codebook, các mã được phân rã với biến ẩn tiềm ẩn có số chiều thấp thường đạt được chất lượng tái tạo tốt hơn khi kích thước tiềm ẩn được giảm từ 256 xuống còn 16 hoặc 8. Hơn nữa, việc loại bỏ l2-normalization dẫn đến kết quả tồi tệ hơn nhiều.

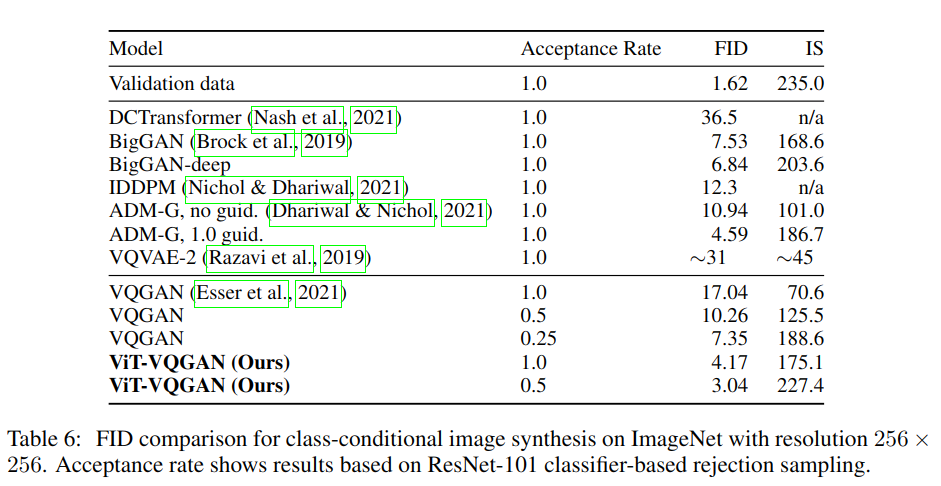


* 1. **IMAGE SYNTHESIS**

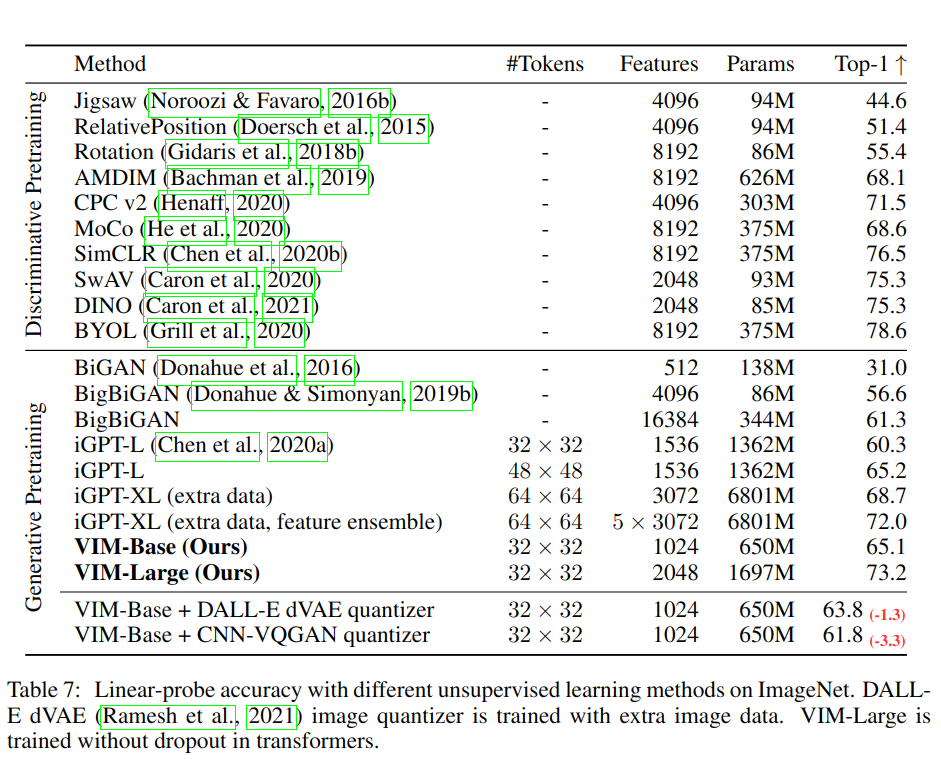
****

* Trên cơ sở của bộ mã hóa hình ảnh ViT-VQGAN đã được học trước, tác giả huấn luyện các mô hình transformer giai đoạn 2 cho tổng hợp hình ảnh không điều kiện và có điều kiện lớp và so sánh với công trình trước đó. Sử dụng kích thước mô hình mặc định là ViT-VQGAN-SS (bộ mã hóa nhỏ, bộ giải mã nhỏ) cho giai đoạn 1 và VIM-Large cho giai đoạn 2 (các kiến trúc mô hình được liệt kê trong Bảng 2), vì tác giả nhận thấy rằng việc đưa thêm tính toán vào giai đoạn 2 sẽ mang lại lợi ích trong khi giữ cho các transformer giai đoạn 1 nhẹ nhàng.





* 1. **UNSUPERVISED LEARNING**
* Sau quá trình tiền huấn luyện tổng quát để mô hình hóa tự động hồi quy mật độ của mã thông báo hình ảnh được lượng tử hóa ViT-VQGAN, chúng ta đánh giá biểu diễn đã học theo giao thức tuyến tính chung về phân loại ImageNet. Chúng tôi tuân theo các siêu tham số đào tạo tương tự như các mô hình tổng hợp hình ảnh vô điều kiện trên ImageNet và sử dụng bộ lượng tử hóa hình ảnh ViT-VQGAN-SS để tăng hiệu suất đào tạo.



1. **References**

<https://ljvmiranda921.github.io/notebook/2021/08/08/clip-vqgan/>

<https://arxiv.org/abs/2110.04627>

<https://arxiv.org/abs/2012.09841>

<https://blog.research.google/2022/05/vector-quantized-image-modeling-with.html>

<https://synthesis.ai/2023/03/21/generative-ai-ii-discrete-latent-spaces/>

<https://mlberkeley.substack.com/p/vq-vae>

-------------------Ending Mệt vãi ---------------------