

# STABLE DIFFUSION CƠ BẢN

Đây là tài liệu nằm trong phần Stable Diffusion cơ bản mà mình chia sẻ, bao gồm các thông tin cũng như thủ thuật giúp bạn nắm được cách dùng Stable Diffusion.

Các tài liệu này được biên soạn dựa trên webui Automatic1111.



## Lời nói đầu

Với mong muốn giúp Stable Diffusion dễ dàng tiếp cận hơn, cũng như góp phần mở rộng cộng đồng Stable Diffusion ở Việt Nam, mình đã và đang chia sẻ các Ebooks và Blogs Tiếng Việt do mình viết và sưu tầm từ nhiều nguồn khác nhau cả trong và ngoài nước. Tất cả đều được chia sẻ miễn phí trên [TRANG CÁ NHÂN](#) của mình.

Toàn bộ Ebooks đều được mình chia sẻ miễn phí. Nếu bạn thích những nội dung mà mình chia sẻ thì có thể donate để ủng hộ mình. Nếu có nhu cầu dùng Ebooks ở một trang web nào khác, xin hãy ghi nguồn và đừng tự chỉnh sửa Ebook này xem như là tôn trọng công sức của những người tạo ra nó.

**XIN HÃY LIÊN HỆ VỚI MÌNH NẾU MUỐN DÙNG CÁC EBOOKS VÀO MỤC ĐÍCH KINH DOANH.**

Chân thành cảm ơn mọi người!

## Nội dung chính

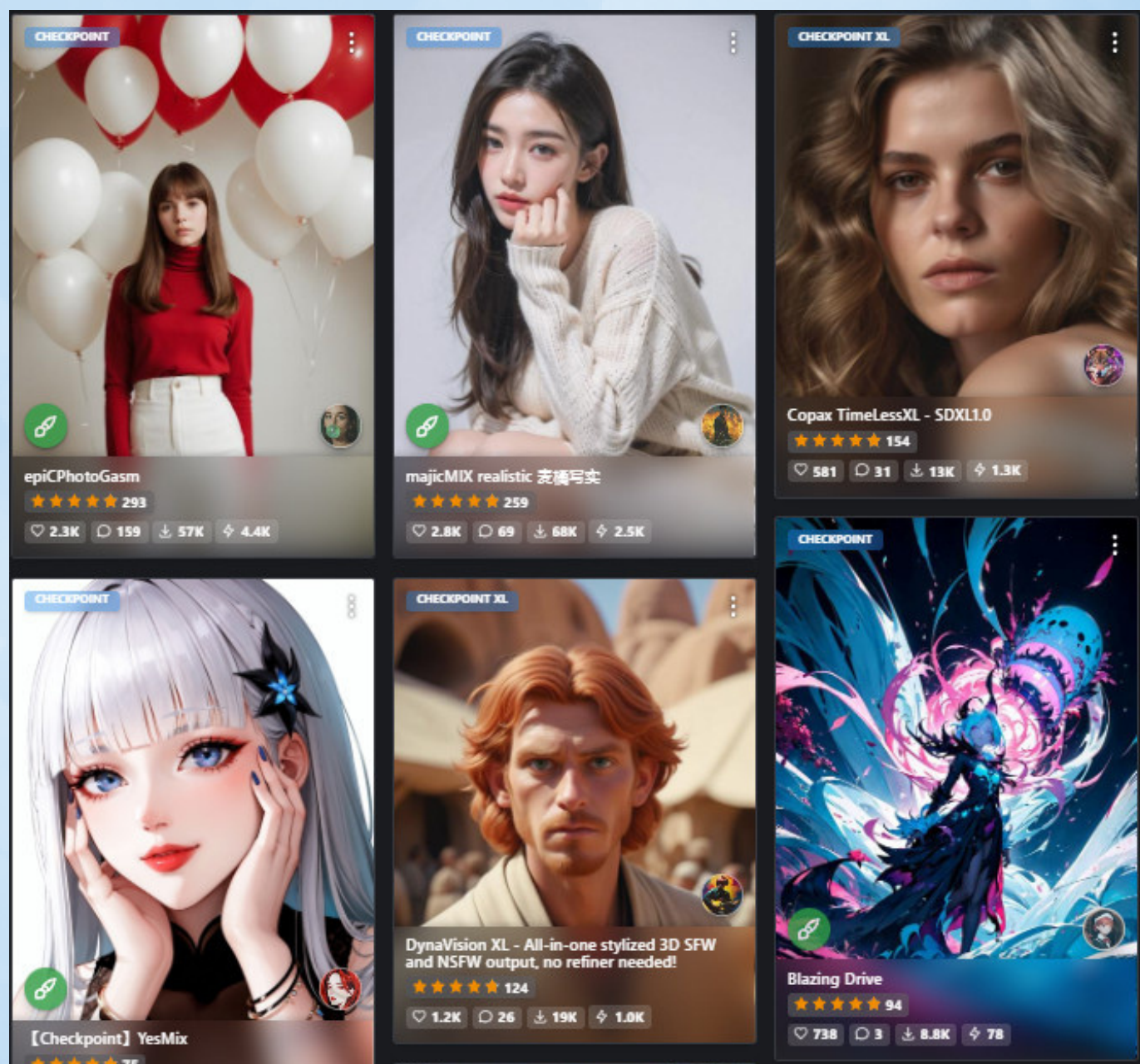
|    |                          |
|----|--------------------------|
| 1. | Checkpoint               |
| 2. | Prompt                   |
| 3. | Sampling method & Step   |
| 4. | Width & Height           |
| 5. | Batch count & Batch size |
| 6. | CFG scale                |
| 7. | Seed                     |
| 8. | Highres fix              |

|     |                   |
|-----|-------------------|
| 9.  | Restore faces     |
| 10. | LoRa              |
| 11. | LyCORIS           |
| 12. | Textual Inversion |

# 1. CHECKPOINT - Lý thuyết

Xem hướng dẫn tải:

**TẠI  
ĐÂY**



Về cơ bản, checkpoint hay model là thứ bạn dùng để tạo nên những bức ảnh.

Mỗi checkpoint như các hộp màu khác nhau, bạn dùng các hộp màu đó để vẽ nên những bức tranh và tùy vào từng model sẽ có các màu sắc, phong cách, thể loại khác nhau.

Các checkpoint bạn tìm thấy hiện nay hầu hết đều được đào tạo và pha trộn từ các checkpoint gốc là SD1.x và SD2.x do StabilityAI và Runway phát triển.

Có rất nhiều trang web để tải các model về nhưng có 2 trang lớn là [CivitAi](#) và [Huggingface](#). Nếu bạn là một người không chuyên về công nghệ thì CivitAi là nơi tuyệt vời dành cho bạn. Nhưng hãy nhớ rằng đây là trang web có những nội dung 18+ nên hãy tùy chỉnh để bảo vệ đôi mắt của bản thân nhé.

## 2. P R O M P T - Lý thuyết

Prompt hay lời nhắc là cách mà chúng ta dùng ngôn ngữ để điều khiển các model cho ra tác phẩm theo ý muốn của mình.

Trong Stable Diffusion sẽ có 2 kiểu prompt là prompt (lời nhắc) và negative prompt (lời nhắc ngược).

Negative prompt là nơi mà chúng ta cho vào đó những lời nhắc để model cho ra các chi tiết mà mình mong muốn, còn negative prompt là nơi bạn điền những thứ mà bạn KHÔNG MUỐN model tạo ra.

Thường thì một câu prompt chuẩn sách giáo khoa sẽ có công thức như sau:

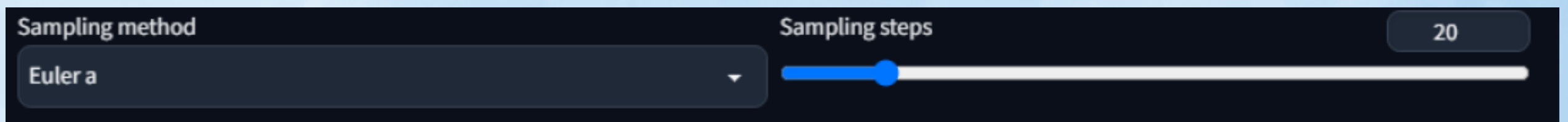
- **Thể loại:** photo, painting,...
- **Vật thể muốn vẽ:** house, man, girl, dog,...
- **Thêm những chi tiết:**
  - **Ánh sáng:** *soft light, ambient light, sun lighting, neon light,...*
  - **Môi trường:** *indoor, outdoor, underwater,...*
  - **Màu chủ đạo:** *vibrant, dark, pastel,...*
  - **Góc nhìn:** *front, overhead, side,...*
  - **Khung cảnh:** *solid color, garden, forest, ocean,...*
- **Style & camera:** 3D render, studio ghibli, movie poster, oil painting, macro, telephoto, fisheye,...

### Ví dụ

**A painting** of **a cute dog** wearing a suit, natural light, in the sky, with bright colors, 3D render

Mình xin chia sẻ một cuốn ebook khá chuẩn chỉnh về prompt từ OpenArt, bạn có thể xem [ở đây](#).

### 3.SAMPLING METHOD & STEP - Lý Thuyết



Về cơ bản thì nó là cách mà máy tính tạo ra ảnh từ model, nếu bạn tưởng tượng model là các hộp màu khác nhau thì sampling method là các cây cọ khác nhau. Mỗi cái sẽ có các đặc điểm riêng tùy theo từng mục đích sử dụng. Về chi tiết thì mình nghĩ viết mọi người cũng lười đọc nên mình để một link rất tần tật về Sampling method [ở đây](#). Step là số vòng lặp mà method đó thực hiện, nó tùy thuộc vào sampling method cũng như model.

Về cơ bản thì các sampling method cho ra kết quả không quá khác biệt như sự khác nhau giữa các model. Không có sampling method nào tốt nhất, tùy vào mục đích sử dụng của bạn.

Hội tụ (convergence) có thể hiểu theo cách là khả năng tạo ra một bức ảnh ổn định ở nhiều lần chạy khác nhau. Điều này rất quan trọng nếu bạn muốn tái tạo lại những bức ảnh dựa trên những thông tin metadata.

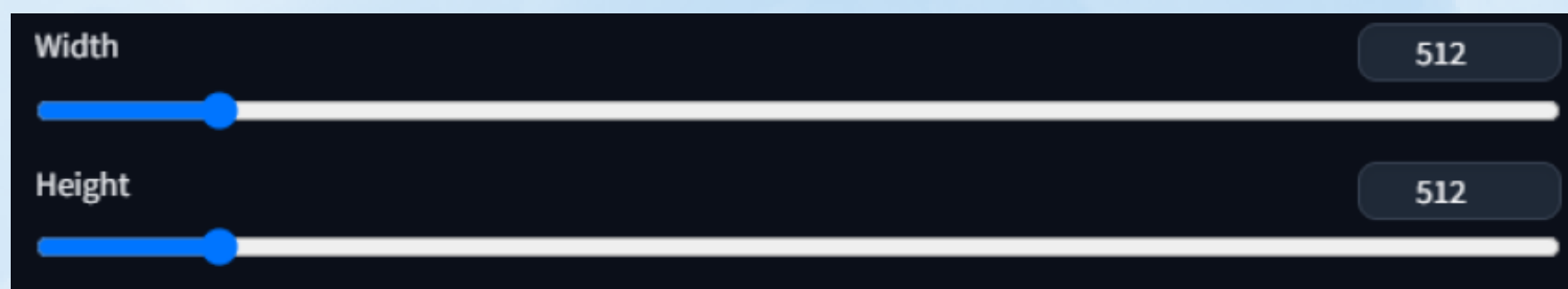
Những sampling method không hội tụ có thể tạo ra những bức ảnh khác so với ban đầu dù sử dụng cùng metadata.

Nếu bạn muốn có **ảnh nhanh, hội tụ và chất lượng ổn** thì có thể dùng **DPM++ 2M Karras hoặc UniPC từ 20-30 step**.

Nếu bạn muốn **ảnh đẹp và không quan tâm về hội tụ** thì có thể dùng **DPM++ SDE Karras (sampling method này khá chậm) từ 8-12 step hay DDIM từ 10-15 step**.

Còn nếu bạn muốn một cái gì đó đơn giản thì hãy dùng **Euler hay Euler a**.

## 4. WIDTH & HEIGHT - Lý thuyết



Nó là kích cỡ của hình ảnh mà bạn sẽ tạo ra  
 Để chọn chiều dài và chiều rộng cho bức ảnh của bạn, có thể kéo thả hoặc nhập trực tiếp vào ô.

Đa số các model đều được đào tạo bởi các ảnh có kích thước 512x512 trở lên, đừng tạo những bức ảnh nhỏ hơn kích thước này vì sẽ không tận dụng được sức mạnh của model.  
 Nếu muốn đổi ngược chiều dài và rộng thì có thể dùng nút [1]



**Bonus cái ảnh mình hay dùng để coi tỉ lệ**

| (w:h) | h    |      |      |      |      |      |     |      |      |      |      |      |      |
|-------|------|------|------|------|------|------|-----|------|------|------|------|------|------|
| w     | 1:1  | 3:4  | 4:3  | 2:3  | 3:2  | 1:4  | 4:1 | 1:2  | 2:1  | 16:9 | 9:16 | 21:9 | 9:21 |
| 0     | 0    | 0    | 0    | 0    | 0    | 0    | 0   | 0    | 0    | 0    | 0    | 0    | 0    |
| 64    | 64   |      |      |      |      | 256  | 128 |      |      |      |      |      |      |
| 128   | 128  |      |      | 192  |      | 512  | 256 | 64   |      |      |      |      |      |
| 192   | 192  | 256  |      |      | 128  | 768  | 384 |      |      |      |      |      | 448  |
| 256   | 256  |      | 192  | 384  |      | 1024 | 64  | 512  | 128  |      |      |      |      |
| 320   | 320  |      |      |      |      | 1280 |     | 640  |      |      |      |      |      |
| 384   | 384  | 512  |      | 576  | 256  | 1536 | 768 | 192  |      |      |      |      | 896  |
| 448   | 448  |      |      |      |      | 1792 | 896 |      |      |      |      | 192  |      |
| 512   | 512  |      | 384  | 768  |      | 2048 | 128 | 1024 | 256  |      |      |      |      |
| 576   | 576  | 768  |      |      | 384  |      |     | 1152 |      |      | 1024 |      | 1344 |
| 640   | 640  |      |      | 960  |      |      |     | 1280 | 320  |      |      |      |      |
| 704   | 704  |      |      |      |      |      |     | 1408 |      |      |      |      |      |
| 768   | 768  | 1024 | 576  | 1152 | 512  |      | 192 | 1536 | 384  |      |      |      | 1792 |
| 832   | 832  |      |      |      |      |      |     | 1664 |      |      |      |      |      |
| 896   | 896  |      |      | 1344 |      |      |     | 1792 | 448  |      |      | 384  |      |
| 960   | 960  | 1280 |      |      | 640  |      |     | 1920 |      |      |      |      |      |
| 1024  | 1024 |      | 768  | 1536 |      |      | 256 | 2048 | 512  | 576  |      |      |      |
| 1088  | 1088 |      |      |      |      |      |     |      |      |      |      |      |      |
| 1152  | 1152 | 1536 |      | 1728 | 768  |      |     |      | 576  |      | 2048 |      |      |
| 1216  | 1216 |      |      |      |      |      |     |      |      |      |      |      |      |
| 1280  | 1280 |      | 960  | 1920 |      |      | 320 |      | 640  |      |      |      |      |
| 1344  | 1344 | 1792 |      |      | 896  |      |     |      |      |      |      |      | 576  |
| 1408  | 1408 |      |      |      |      |      |     |      | 704  |      |      |      |      |
| 1472  | 1472 |      |      |      |      |      |     |      |      |      |      |      |      |
| 1536  | 1536 | 2048 | 1152 |      | 1024 |      | 384 |      | 768  |      |      |      |      |
| 1600  | 1600 |      |      |      |      |      |     |      |      |      |      |      |      |
| 1664  | 1664 |      |      |      |      |      |     |      | 832  |      |      |      |      |
| 1728  | 1728 |      |      |      | 1152 |      |     |      |      |      |      |      |      |
| 1792  | 1792 |      | 1344 |      |      |      | 448 |      | 896  |      |      | 768  |      |
| 1856  | 1856 |      |      |      |      |      |     |      |      |      |      |      |      |
| 1920  | 1920 |      |      |      | 1280 |      |     |      | 960  |      |      |      |      |
| 1984  | 1984 |      |      |      |      |      |     |      |      |      |      |      |      |
| 2048  | 2048 |      | 1536 |      |      |      | 512 |      | 1024 | 1152 |      |      |      |

# 5. BATCHCOUNT & BATCHSIZE

## Lý Thuyết

Batch (lô) là số lượng hình ảnh mà bạn sẽ tạo ra trong một lần chạy thay vì phải nhấp nút Generate liên tục

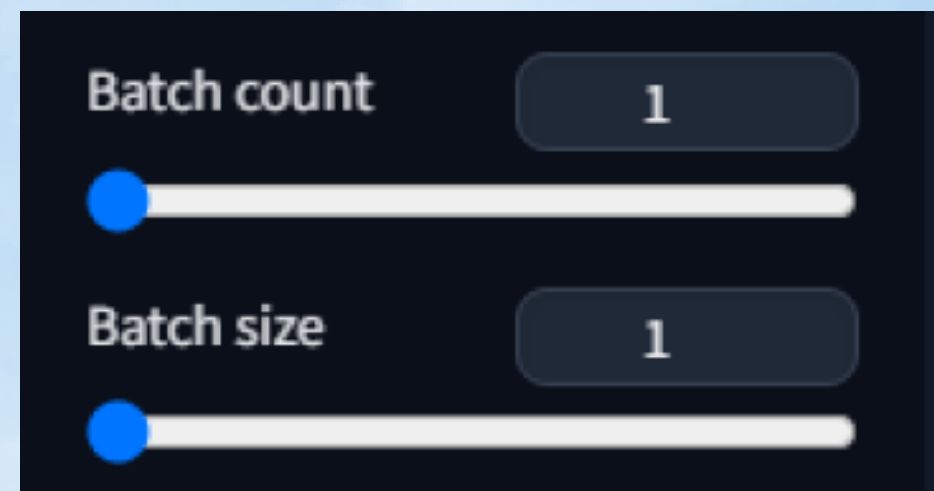
Batch size: số lượng ảnh trong một batch hay nói cách khác, số lượng ảnh mà Stable Diffusion sẽ tạo ra cùng một lúc. GPU càng mạnh càng nhiều VRAM thì càng nhiều ảnh có thể tạo ra cùng một lúc.

Batch count: đơn giản là số lượng batch cần chạy

Tận dụng batch size nếu bạn có một chiếc card đồ họa mạnh. Do chúng là số ảnh tạo ra cùng một lúc nên sẽ nhanh hơn so với dùng batch size.

Ví dụ:

Tạo 4 ảnh với batch size = 4, batch count = 1 thường sẽ nhanh hơn nhiều so với tạo 4 ảnh với batch size = 1, batch count = 4.



# 6. CFG SCALE - Lý thuyết



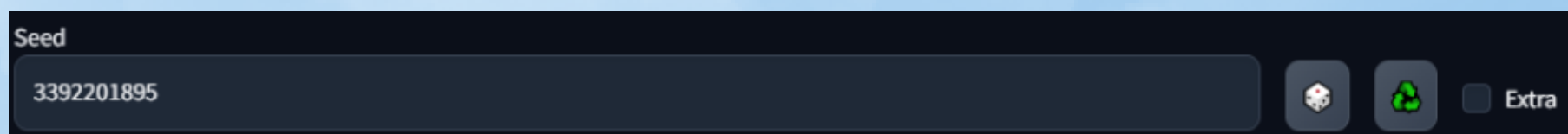
CFG scale là mức độ bạn muốn Stable Diffusion phải tạo ra ảnh giống như prompt bạn viết. Nghĩa là CFG scale càng cao thì nó sẽ tạo ra ảnh càng gần với prompt của bạn. CFG quá thấp sẽ không tạo ra được một bức ảnh ổn định, nhưng quá cao lại làm mất đi sự sáng tạo cũng như có thể gây nhiễu.

Thông thường nên dùng CFG scale trong khoảng 5-15.

Mặc định là 7, nếu bạn muốn SD vừa tuân theo prompt, vừa sáng tạo thì nên để scale trong khoảng 7-9.

Chỉ khi nào bạn tự tin là prompt mình là đầy đủ và đủ tốt để thể hiện tất cả những gì bạn muốn thì mới dùng CFG scale từ 12-15.

# 7. SEED - Lý thuyết



Seed là một số nguyên và sẽ được gán cho từng bức ảnh, can thiệp vào quá trình tạo ảnh. Seed có tác dụng là làm đa dạng các bức ảnh khi các thông số khác không thay đổi. Seed cũng có tác dụng làm metadata nếu bạn muốn tái tạo lại một bức ảnh mà không làm thay đổi quá nhiều so với ảnh gốc.

Để seed = -1 nó sẽ tạo ngẫu nhiên một số nguyên và gán cho các bức ảnh.

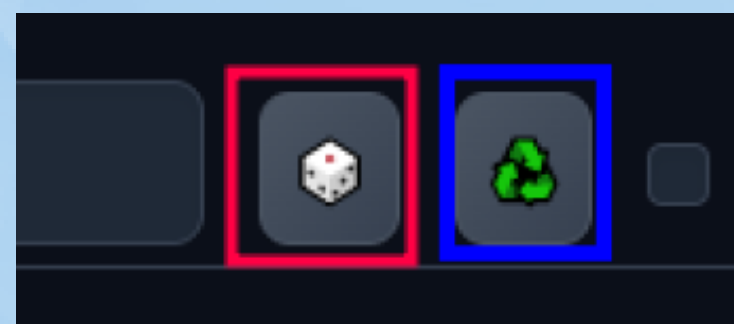
Sự thật là luôn có một số seed tốt hơn số còn lại, do đó mới có nhu cầu tái tạo bức ảnh dựa trên một seed nào đó.

Seed là một số nguyên nhưng có giới hạn, nếu bạn nhập số quá lớn quá dài thì sẽ bị lỗi đấy.

Khi tạo ảnh theo batch thì các seed của các ảnh sẽ nối tiếp nhau chứ không giống nhau.

Nút [đỏ] giúp set nhanh seed = -1

Nút [xanh] giúp set nhanh seed của ảnh gần nhất vừa được tạo



## 8. HIGHRES FIX - Lý thuyết

Trước khi nói về highres fix và cách dùng thì phải hỏi trước một câu là: Tại sao lại cần dùng nó nhỉ?

Do các model của Stable Diffusion được đào tạo trên những ảnh 1:1 với kích thước 512x512 hoặc 768x768 nên khi muốn tạo một ảnh có ratio không phải hình vuông như 3:2, 16:9 hay có kích thước lớn thì Stable Diffusion sẽ chập nhiều ảnh với nhau và cố gắng khóa lấp những mảng còn thiếu. Từ đó gây ra hiện tượng "ba đầu sáu tay" mà mọi người vẫn hay gặp.

Highres fix - như tên gọi của nó, sinh ra để giải quyết các vấn đề khi tạo ảnh có kích thước lớn.

### Upscaler

Upscaler bạn có thể hiểu đơn giản là một model AI hay thuật toán giúp tăng độ sắc nét cho ảnh khi phóng to ảnh, hạn chế tình trạng bị nhiễu hoặc mờ.

Lựa chọn upscaler cũng rất quan trọng, mỗi upscaler nó sẽ có khả năng và chất lượng khác nhau. Ví dụ có model upscaler này thích hợp dùng cho ảnh Anime style, có model thích hợp cho ảnh Realistic, có model thích hợp cho khuôn mặt,...

Để sử dụng một model upscaler mới tải về, bạn cần phải xem nó thuộc nhóm upscaler nào (ESRGAN, GFPGAN, SwinIR,...) rồi bỏ nó vào thư mục tương ứng trong thư mục model của Auto1111.

Mình xin giới thiệu một trang web siêu tổng hợp về các model upscaler ở [đây](#). Bạn có thể đọc tham khảo và chọn một vài model phù hợp nhất với nhu cầu của mình.

### Highres step

Highres step là số step mà bạn muốn highres fix chạy trên ảnh. Thường thì bạn nên để số step > 15 để model highres hoạt động tốt.

Nếu để quá thấp thì nó sẽ gây nhiễu cả bức hình.

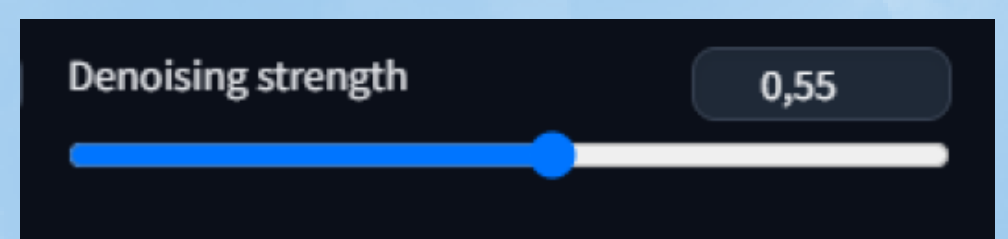
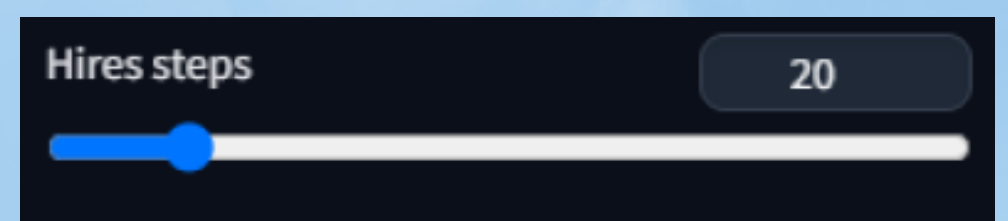
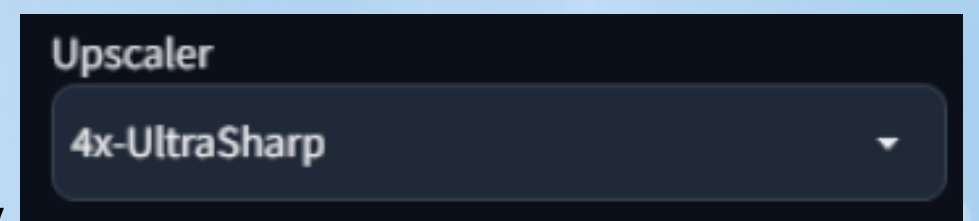
Nếu để highres step = 0, nó sẽ **mặc định lấy số highres step bằng với số sampling step.**

Ví dụ bạn tạo ảnh với 30 step và chạy highres step = 20 thì tổng số step bạn chạy là  $30 + 20 = 50$  step.

### Denoising strength

Denoising strength là cách model denoise (giảm nhiễu) cho ảnh của bạn.

Bạn cứ hiểu đơn giản là denoising strength càng lớn thì ảnh của bạn càng được thêm thắt nhiều chi tiết. Denoising strength quá nhỏ thì nó thiếu độ sắc nét, còn quá lớn thì nó biến thành ảnh khác luôn =>) Khi dùng highres fix, mình đề xuất mọi người chỉ nên dùng denoising strength trong khoảng 0.5-0.65.



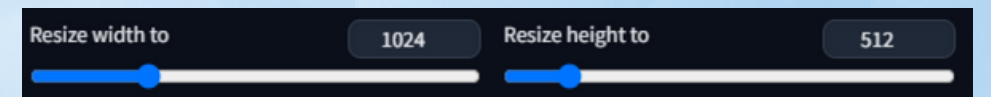
## Size

Nếu bạn còn nhớ cách hoạt động của highres mình ghi trong slide trước thì sau giai đoạn upscale ảnh lên, nó sẽ crop ảnh theo cái size bạn đã resize lại.

Ví dụ bạn chọn size ảnh đầu vào là 512x512, và chọn resize lại thành 1024x512. Nó sẽ upscale tấm gốc 512x512 thành 1024x1024, sau khi xử lý, nó sẽ crop lại thành size 1024x512. Do đó nếu là ảnh của một nhân vật thì rất có thể phần đầu và chân của nhân vật sẽ bị crop đi.



kích thước ban đầu: 512x512



kích thước resize: 1024x512

=> Do đó, bạn nên để kích thước resize có cùng ratio như kích thước đầu vào, điều đó sẽ giúp bảo toàn nguyên vẹn bức tranh. Hoặc nếu bạn không muốn tính toán, cứ để resize width và height về 0 rồi dùng thanh upscale by để chỉnh

## Cách dùng chuẩn

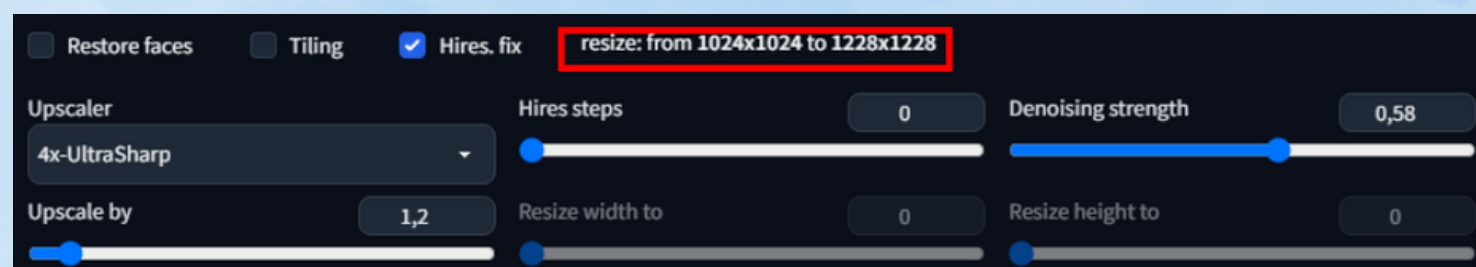
Tất nhiên để tìm được thông số chuẩn thì phải thực nghiệm thôi, nhưng nếu bạn muốn một cái gì đó nhanh chóng và có khả năng áp dụng được cho đa số trường hợp thì đây:

Upscaler: Dùng 4x-UltraSharp chuẩn với hầu hết trường hợp.

Highres step: Cứ để 0 hoặc chỉnh từ 15 trở lên

Denoising strength: 0.58, vừa giữ nguyên vẹn tấm hình, vừa có thể thêm thắt các chi tiết cho tranh.

Size: Chỉnh resize về 0 rồi dùng thanh Upscale by, bạn có thể thấy được kích thước sau khi resize ở phía trên thanh highres step.



## Một số Upscaler thú vị

- 4x-UltraSharp: Cái này thì mình đã giới thiệu ở slide trước, nó khá tiện dụng và hiệu quả cho hầu hết các nhu cầu từ ảnh chân dung đến phong cảnh, họa tiết,...
- 4x-AnimeSharp: Cũng là UltraSharp nhưng phiên bản dành cho Anime, cá nhân mình đánh giá nó cao hơn R-ESRGAN 4x+ Anime6B.
- 4x\_fatal\_Anime\_500000\_G: Chủ yếu là dùng cho ảnh Anime, có khả năng fix những chi tiết nhỏ, đồng thời một phần nào đó có xác suất giúp cải thiện khả năng vẽ tay cho nhân vật.
- KDM003\_scans\_1x: giảm thiểu noise và chỉnh màu, chuyên dùng cho các bác hay in ấn bằng màu CMYK, một phần nào đó giảm thiểu sự đổi màu so với hình gốc khi in.

Và tất nhiên ngoài những upscaler trên còn vô số upscaler thú vị mà mình không biết. Tất cả những upscaler này đều có trên trang web mà mình share ở [đây](#).

Mình sẽ để phần so sánh một vài upscaler trong phần "Cách dùng Extra để upscale ảnh" vì chỉ dùng highres fix không thể hiện được hết sức mạnh của các upscaler này. Nhưng nó hoàn toàn đáng giá để bạn thử đấy.



# Highres fix và Upscale

Sự khác biệt khi dùng highres fix và Upscale là highres fix dùng để sửa ảnh khi nó có những lỗi như "ba đầu sáu tay", cơ thể không cân đối, chi tiết quá nhòe,... khi bạn tạo ra các ảnh có độ phân giải lớn và ratio khác 1:1.

Upscale dùng khi bạn đã hoàn toàn ưng ý với ảnh của mình và muốn biến nó thành một ảnh có độ phân giải cao Full HD, 2K, 8K,...

Lời khuyên của mình là nên chỉnh sửa thật kĩ các chi tiết chưa ưng ý bằng các thủ thuật khác nhau (và highres fix chỉ là một trong số đó) trước khi Upscale vì những chi tiết lỗi đó cũng sẽ to lên khi sử dụng Upscale, và tất nhiên, việc chọn lựa một Upscaler phù hợp cũng là một việc quan trọng.

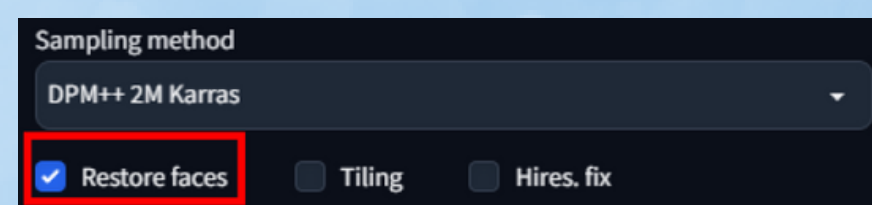


## 9.RESTORE FACES

### *Lý thuyết*

Thực ra phần Restore faces này chỉ là bonus để giải đáp cái nút này làm gì thôi chứ bình thường mình không dùng nó, vì nó không đa dụng và thực sự không hữu ích đối với cá nhân mình.

Được rồi, chức năng của nó chính là chỉnh sửa lại khuôn mặt cho đẹp hơn khi tạo ảnh toàn thân (thường bị biến dạng khuôn mặt như hình dưới).



**Trước**



**Sau**

# 10.LORA

## Lý thuyết

LoRA (LowRank-Adaptation of LLM), có thể hiểu đơn giản như một mini model

được train trên một tập dữ liệu nhất định. Khi hoạt động, nó sẽ trở thành các layer được "chèn" vào các layer của model gốc, khiến SD tạo ra nhân vật hay phong cách mà LoRA đó được train.

Nhưng dùng LoRA thì có lợi ích gì? Hãy giả sử bạn muốn tạo hình nhân vật Doraemon, nhưng nếu model đó chưa được train trên tập dữ liệu Doraemon, nó sẽ không biết đó là ai, vẻ bề ngoài như thế nào.

Dù bạn có miêu tả kĩ lưỡng nhân vật bằng prompt đến mức nào thì model vẫn sẽ không tạo được trọn vẹn một nhân vật mà nó còn chưa "nhìn thấy" bao giờ.

Khi đó, bạn chỉ cần dùng LoRA Doraemon, model sẽ sinh ra ngay cho bạn nhân vật yêu thích mà không cần phải miêu tả gì nhiều.

Có thể nói, LoRA đem đến cho bạn

sức mạnh của một model với chi phí lưu trữ ít hơn so với một model chính hiệu.

## Cách dùng LoRA

Để dùng được LoRA, hãy đảm bảo phiên bản của bạn không quá cũ và có phần Extra network

(góc phải bên dưới nút Generate)

Tải LoRA về máy và lưu vào thư mục: <đường dẫn>\stable-diffusion-webui\models\Lora

Khi gõ prompt, bạn hãy thêm LoRA và weight của nó vào dưới dạng <lora:tên lora:weight>.

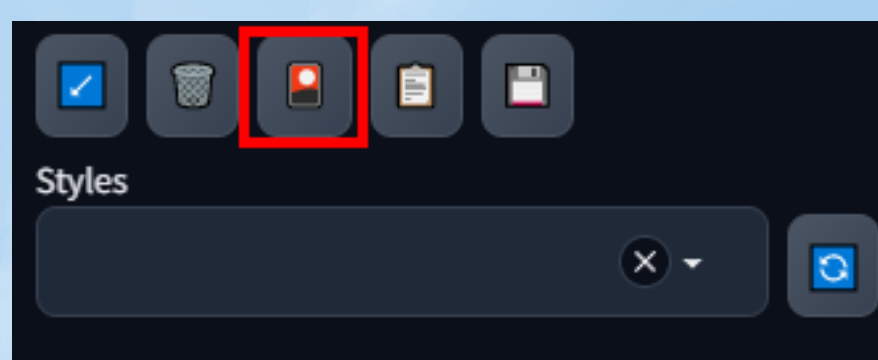
Ví dụ:

<lora:doraemon\_lora:0.6>

Nếu bạn không nhớ tên LoRA cũng không sao, hãy nhấp vào LoRA bạn muốn dùng trong Extra network

,LoRA sẽ được tự động thêm vào prompt.

Đừng quên mất thêm trigger words cần thiết (nếu có) vào prompt để LoRA phát huy sức mạnh nhé.



| Details       |                   |
|---------------|-------------------|
| Type          | LORA              |
| Downloads     | 5.758             |
| Uploaded      | Jun 11, 2023      |
| Base Model    | SD 1.5            |
| Trigger Words | BANDEAU           |
| Hash          | AUTOV2 FAAF9FFC03 |

## Vị trí của LoRA trong prompt

Khi dùng LoRA thì cũng như dùng model vậy, nếu sử dụng cùng thông số thì ảnh tạo ra bao nhiêu lần sẽ y hệt nhau?

**KHÔNG HỀ BẠN ƠI!**

Mặc dù chỉ mang tính chất Enable LoRA, thế nhưng LoRA ở các vị trí khác nhau trong prompt dù dùng chung thông số khác, cũng sẽ cho ra các kết quả khác nhau đấy! Tất nhiên là có phụ thuộc vào Sampler nữa, nhưng sự khác nhau là rõ rệt. Do đó nếu bạn muốn (và có nhiều thời gian) có thể thử xem vị trí nào của LoRA trong prompt sẽ mang lại kết quả bạn ưng ý nhất.



LoRA đặt ở đầu prompt.  
Sampler: DPM++ 2M Karras



LoRA đặt ở cuối prompt.  
Sampler: DPM++ 2M Karras

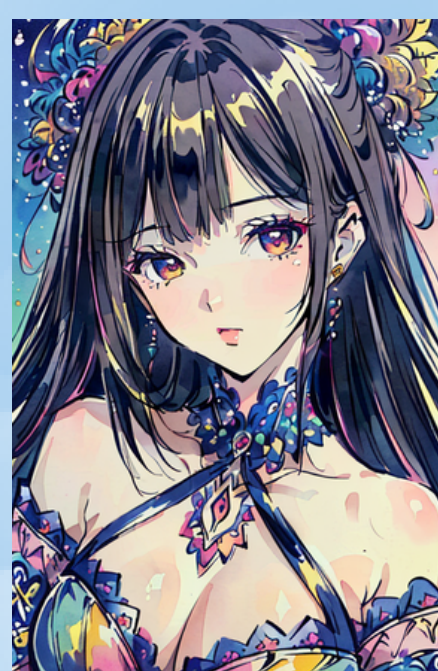
## Vị trí của LoRA trong prompt



LoRA ở đầu prompt.  
Sampler: UniPC



LoRA ở cuối prompt.  
Sampler: UniPC



LoRA ở đầu prompt.  
Sampler: DPM++ SDE



LoRA ở cuối prompt.  
Sampler: DPM++ SDE

# 11.LYCORIS

## LyCORIS

(Lora be Yond Conventional methods, Other Rank adaptation Implementations for Stable diffusion)

nổi lên như một thế lực mới, hứa hẹn đem đến những kết quả còn tuyệt vời hơn so với LoRA, vậy thực ra LyCORIS là cái gì?

LyCORIS, đùa thôi chứ thật ra nó cũng không hẳn là cái gì mới cả. Đây là một project cố gắng sử dụng nhiều thuật toán khác nhau để áp dụng LowRankAdaptation (LRA) vào SD model một cách hiệu quả hơn, bao gồm cả LoRA.

LoRA là thứ đầu tiên áp dụng LRA.

LoCon là LoRA nhưng sử dụng Conventional.

LoHA là LoRA nhưng sử dụng Hadamard Product.

Và tất nhiên trong tương lai sẽ còn xuất hiện thêm nhiều biến thể hơn nữa.

Lycoris thường đem lại cho ta hiệu quả tốt hơn LoRA bằng cách sử dụng các thuật toán.

Một ví dụ là nhiều

LoRA chỉ train trên khuôn mặt nhân vật thường mang lại hiệu quả không tốt trong việc phối hợp gương mặt và trang phục.

Hiện tại, LoHA giải quyết vấn đề này tốt nhất trong nhà LyCORIS

, mặc dù vẫn không hoàn hảo. Trong tương lai, các thuật toán càng xịn xò hơn sẽ được áp dụng và chúng ta có thể mong chờ những phương pháp đem lại kết quả tuyệt vời hơn nữa.

Ngoài weight như LoRA thông thường, bạn còn có thể chỉnh thêm các trọng số khác. Tùy chỉnh và kết hợp chúng lại với nhau có thể sẽ giúp bạn tạo ra được sản phẩm tốt hơn.

Cú pháp là:

<lyco:Model:te=1:unet=0.5:dyn=13>

Trong đó:

**te**: weight của text encoder

**unet**: weight của UNet

**dyn**: số row sử dụng thuật toán của DyLoRA

Nếu bạn muốn nghiên cứu sâu hơn, có thể đọc qua Github của Extension và các tài liệu liên quan để tìm hiểu thêm.

Hoặc bạn cứ sử dụng nó như LoRA thông thường vẫn đem lại kết quả rất tốt.

# 12. Textual Inversion

## Textual Inversion (Embedding)

Trước khi LoRA được áp dụng, để không phải ngồi train lại model, mọi người thường dùng kĩ thuật Textual Inversion để giúp SD tạo raphong cách hay nhân vật mình cần.

Khác với Lora, Textual Inversion không phải là một mini model mà nó tương tự như là một cụm các từ nhắc (prompt) được nén dưới dạng một từ, có tác dụng hướng dẫn SD tạo ra thứ bạn cần chính xác hơn dựa trên bộ dữ liệu mà model đã được train, và hoàn toàn không thêm data hình ảnh vào phần UNet.

Ví dụ thế này, bạn muốn diễn tả style hoàng gia cho bức ảnh của mình, nhưng nếu bạn chỉ dùng một từ prompt "Royal style" thì chưa chắc SD đã hiểu đúng ý bạn. Thay vào đó, một Textual Inversion của "Royal style" sẽ được train để giúp model hình dung ra style hoàng gia có những đặc điểm gì (ví dụ như "sang trọng", "giàu sang", "uyngghi", ...). Có thể hiểu đơn giản là tất cả những từ đó được gói gọn trong chiếc textual inversion mà bạn cho vào prompt, từ đó SD sẽ mô tả tốt hơn thứ bạn mong muốn.



Thế nhưng, việc bản thân không mang theo data train cũng có điểm yếu. Ví dụ bạn train một Textual Inversion Doraemon trên một model không có data về Doraemon thì sẽ không đem lại kết quả tốt.

LoRA, do nó có mang data hình ảnh nên có thể dùng ở bất kì model nào. Ở các bản SD2.x, do được train bằng bộ dataset to và tốt hơn nên Textual Inversion được tạo ra bởi các model SD2.x base sẽ hoạt động tốt hơn so với SD1.x base.

Textual Inversion được train ở model base nào chỉ có thể dùng trên các model được chỉnh sửa từ model base đó.

Nghĩa là Textual Inversion được train bởi các model SD2.x sẽ không hoạt động trên model SD1.x

Để dùng Textual Inversion, bạn chỉ cần tải về và bỏ vào thư mục <đường dẫn>\stable-diffusion-webui\embeddings

Sau đó ở Extra Networks, Refresh lại và dùng như một từ prompt thông thường.

Nếu bạn để ý thì cách dùng Textual Inversion không có cú pháp gì như LoRA, hết như một prompt bình thường. Nó được load lên ngay từ đầu bởi webui. Khi bạn dùng, nó sẽ được tự động được load lên và áp dụng vào. Tuy nhiên do không có cú pháp, nếu bạn vô tình gõ prompt có từ khóa của Textual Inversion, nó sẽ ngay lập tức được áp dụng, không có cú pháp nào để ngăn việc tự động áp dụng cả, cách bạn có thể làm chỉ là đổi tên Textual Inversion hoặc remove nó ra khỏi thư mục embeddings hoặc không dùng prompt đó nữa =))

# Negative Embedding

Ngày nay, LoRA đã quá đỗi mạnh nên Textual Inversion không còn phổ biến lắm. Thế nhưng nhiều người dùng lại thích Textual Inversion hơn ở một số điểm:

- Nhẹ, rất nhẹ, chỉ tầm vài chục Kb.
- Nó khai thác hết tinh hoa của một model. Có những thứ không thật sự cần dùng LoRA. Ví dụ như một số kiểu tóc, rất nhiều model đều hiểu được nó là gì, không cần phải lạm dụng LoRA.
- So với dung lượng từ 1 - 200 Mb của LoRA, một textual inversion có thể làm điều tương tự với vài chục Kb. Tuy nhiên, do nhu cầu ngày càng ít do đó không mấy người tiếp tục train Textual Inversion nữa mặc dù mình nghĩ nó là một thứ rất tuyệt nếu nhận được đồng đảo sự yêu quý từ mọi người.

*Tuy nói thế, nhưng ngày nay vẫn còn một dạng Textual Inversion vẫn còn trụ vững và được đồng đảo người sử dụng, chính là NegativeEmbedding.*

## Negative Embedding

Như bên trên đã nói về Textual Inversion, Negative Embedding cũng hoạt động y như thế nhưng

thay vì train trên thứ bạn mong muốn, nó lại được train dựa trên negative prompt.

Tác dụng của nó là thay vì bạn gõ một đoạn negative prompt dài dòng, bạn có thể thay thế nó bằng một Negative Embedding rồi bỏ vào negative prompt mà dùng, rất gọn và hữu ích!

Do cách thức hoạt động khác nhau nên ở thời điểm hiện tại không xuất hiện Negative LoRA, do đó Negative Embedding vẫn được ưa chuộng.

Tuy nhiên, thực tế bạn vẫn có thể tạo ra một kiểu NegativeLoRA nhưng cách dùng hết như LoRA thông thường. Tuy không thể dùng ở Negative Prompt như Negative Embedding, LoRA dạng này vẫn có thể được sử dụng với chức năng tương tự.

## Checkpoint vs LoRA vs LyCORIS vs Textual Inversion

- **Checkpoint** (model): Trái tim của Stable Diffusion, được train trên các tập data lớn và đa dạng, đem lại cho bạn khả năng sáng tạo bất kì thứ gì bạn tưởng tượng ra.
- **LoRA**: Một kĩ thuật tạo ra các mini model, được train trên một tập data nhỏ, thông thường là về một nhân vật hoặc một style cụ thể. Khi hoạt động, nó sẽ được chèn vào model gốc giúp tạo ra thứ mà nó được train một cách dễ dàng.
- **LyCORIS**: Là LoRA nhưng được gắn thêm các thuật toán giúp chúng hoạt động tốt hơn.
- **Textual Inversion** (Embedding): Tuy có cách thức hoạt động khác, chức năng của nó tương tự như LoRA nhưng yếu hơn, do bản thân không mang theo data nên kết quả đem lại có thể không mạnh như LoRA. Tuy nhiên nó siêu nhẹ và hoạt động rất tốt trên các model được train nhiều dữ liệu.

# PHỤ LỤC

Vài đường link cho những ai muốn tìm hiểu sâu hơn

Một video giải thích rất dễ hiểu về cách hoạt động của Model, LoRA,...ở [đây](#).

Tóm tắt thuật toán của một số LyCORIS ở [đây](#).

Paper của LoRA ở [đây](#), còn paper của LoHA ở [đây](#).

Tìm hiểu về cách train Textual Inversion của một đại thần chuyên Textual Inversion ở [đây](#).

Bạn có thể thử train một "Negative LoRA" dựa trên kinh nghiệm của một đại thần ở [đây](#).

Github của Extension LyCORIS, bạn có thể tìm thêm cách chỉnh sửa các weight của LyCORIS ở [đây](#).

Nếu bạn thấy Ebook này là hữu ích, có thể ủng hộ mình bằng cách follow [Twitter](#)

hoặc Gen hình ảnh qua Model trên [Tensor](#) của mình để chia sẻ cũng như học hỏi nhé.

Hoặc nếu có điều kiện, bạn cũng có thể ủng hộ mình qua các kênh sau. Tất cả sự ủng hộ dù nhỏ nhất của mọi người luôn là động lực để mình tìm tòi và tạo ra nhiều bài chia sẻ hơn nữa!

Support me a coffee here: 

 [PayPal jayhuangvn@gmail.com](mailto:jayhuangvn@gmail.com)



Creator profile JAY CREATOR

Realistic, Character, Asian Style, Fashion Model,

 tensor.art