Dalle-2 : Hierarchical Text-Conditional Image Generation with CLIP Latents

Jun 2, Jeong Hyun Jae, BME Journal Club

Hyunjae Jeong (tAILab, CCIDS)

Yonsei University, Medical Life Systems Information Center (TAIL Lab)
Severance Hospital, Center for Clinical Imaging Data Science (CCIDS)
Severance Hospital, Radiology

CCIDS 의료영상데이터사이언스센터 Center for Clinical Imaging Data Science

tAILab.

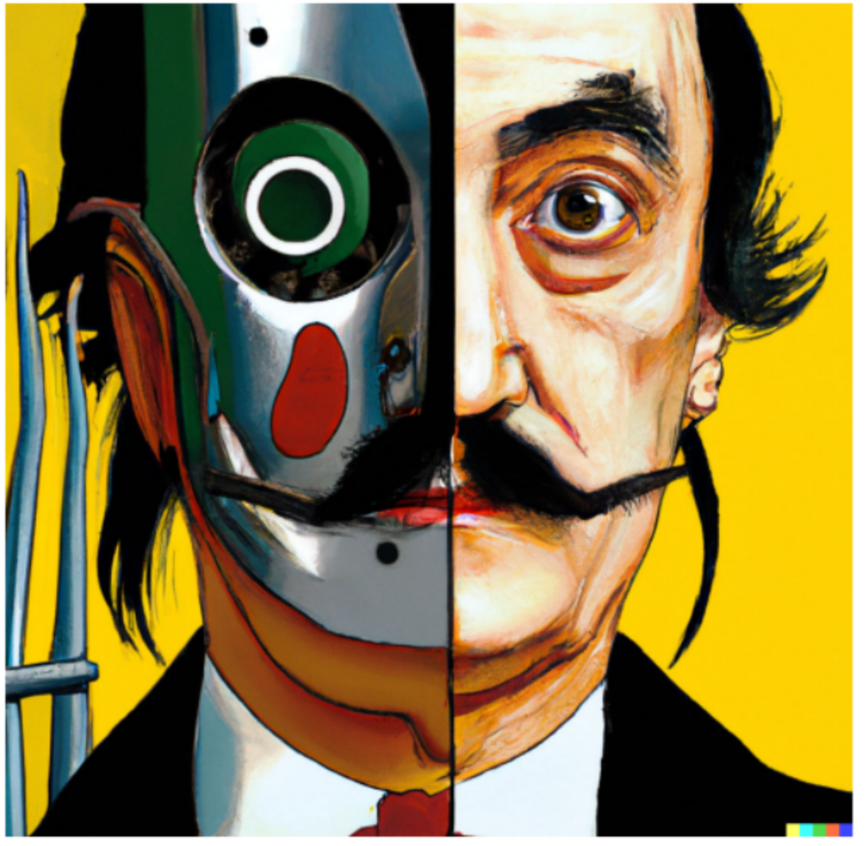
01 Introduction

tAILab.

Text-to-Image Generation

- Automatically describing the content of an image is a fundamental problem in artificial intelligence that connects computer vision and natural language processing.

A teddy bear on a skateboard in times square



Vibrant portrait painting of Salvador Dalí with a robotic half face

01 Introduction

tAILab.

Why DALLE-2?

- DALL·E 2 is a new AI system that can create realistic images and art from a description in natural language.
- DALL·E 2 can create original, realistic images and art from a text description. It can combine concepts, attributes, and styles -> Splendid capability, scales (zero-shot, representation learning, contrastive learning, high resolution) !!!

Original Image

→

DALLE2 Variations

01 Introduction

tAILab.

Zero-shot learning (ZSL), Contrastive learning, Representation learning

- Zero-shot learning (ZSL)

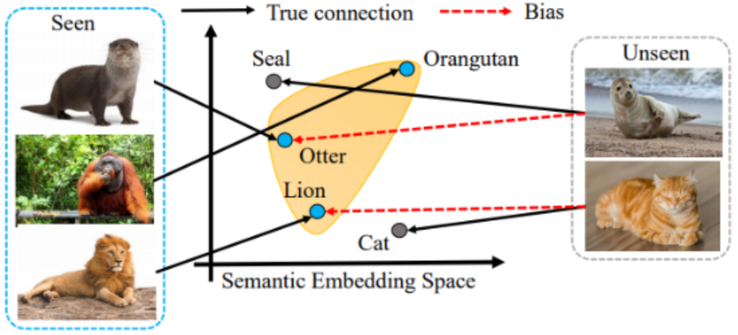
  - Predict unseen data not included in the training set through semantic information (e.g. word imbedding in this journal)

- Contrastive learning

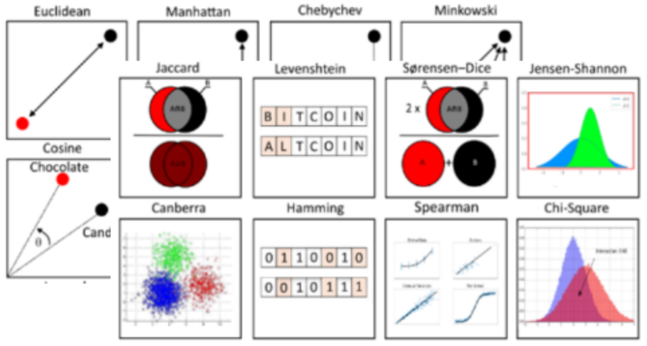
  - Contrast : great difference between two or more things which is clear when you compare them.
  - The goal is to learnin a similarity function that measures how similar or related two objects are.

- Representation learning

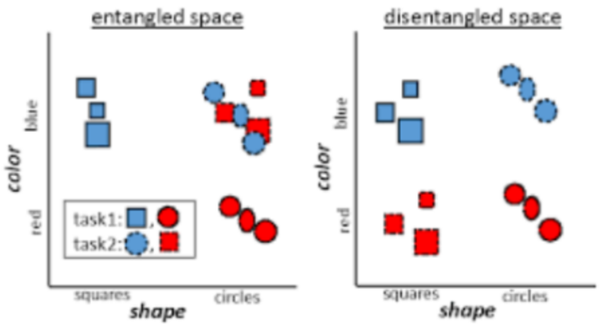
  - Many information processing tasks can be very easy or very difficult depending on how the information i s represented



Zero-shot Learning



Contrastive Learning



Representation Learning

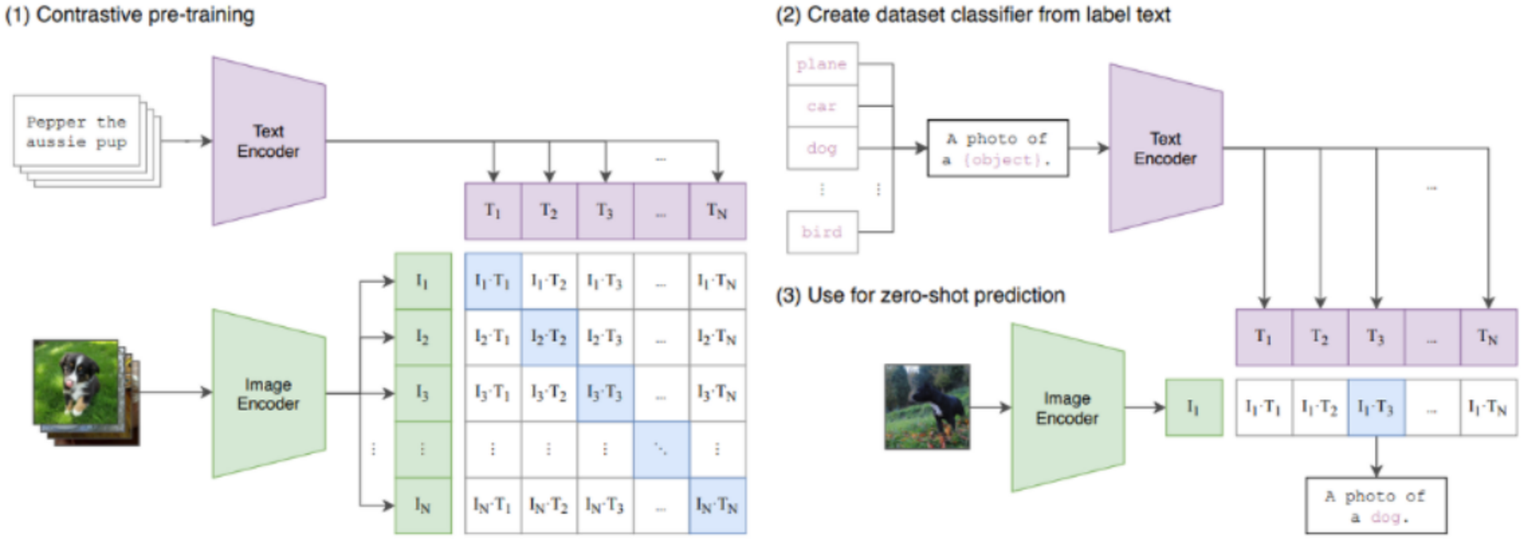
01 Introduction

CLIP, Diffusion Model

- CLIP

  - Typical vision datasets are labor intensive and costly to create while teaching only a narrow set of visual co ncepts; models that perform well on benchmarks have disappointingly poor performance on stress tests

  - CLIP learns the multi-modal embedding space by learning the image encoder and the text encoder togeth er to maximize the cosine similarity of N real pairs of image embeddings and text embeddings and minimize the cosine similarity of false pairs.



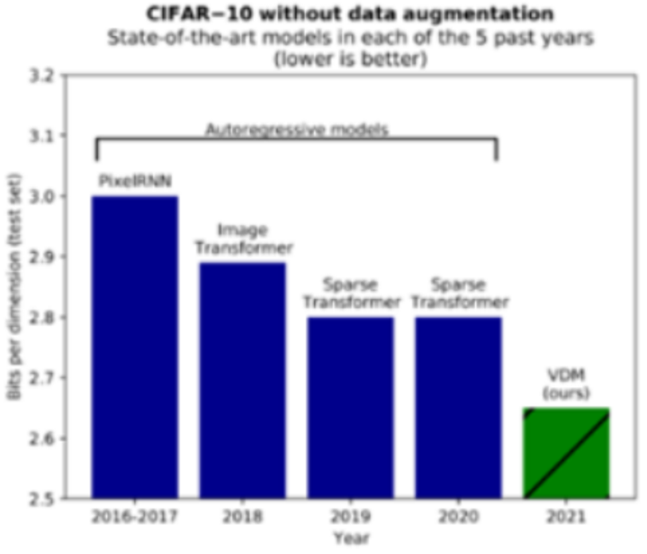
CLIP Approaches

01 Introduction

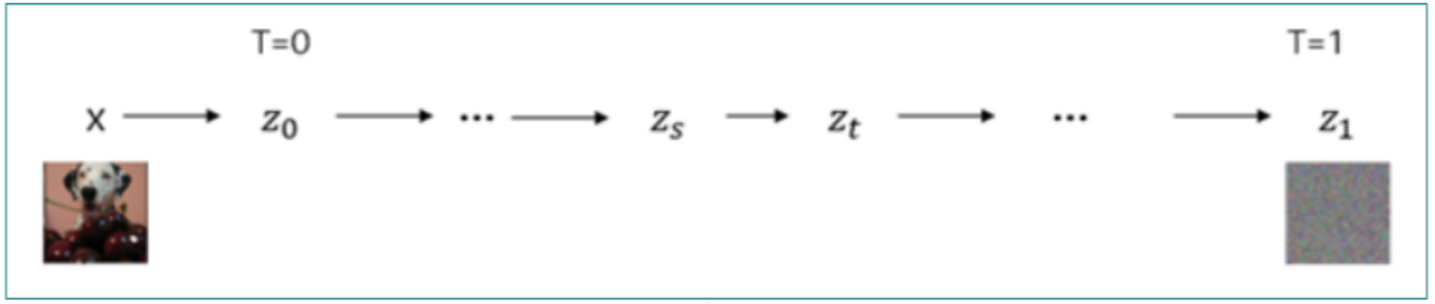
CLIP, Diffusion Model

- Diffusion Model

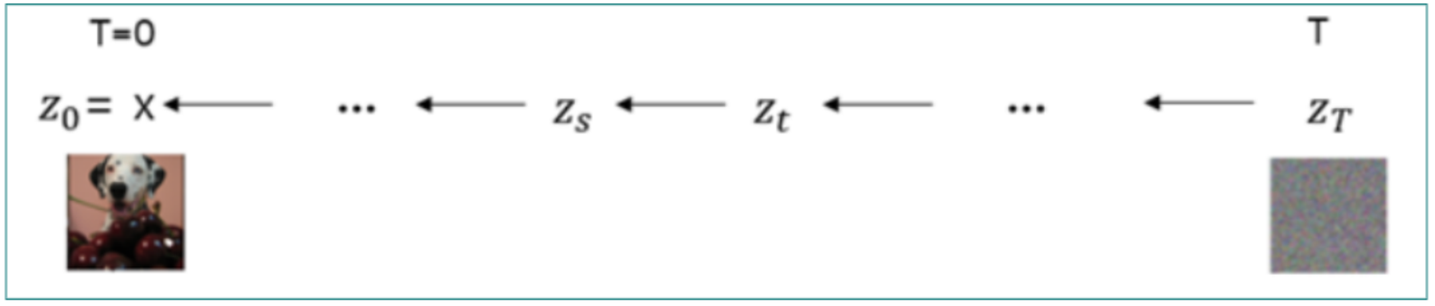
  - Diffusion model is a model inspired by thermodynamics and is largely divided into two stages. First, we gradually ad d noise to the given data x. This process is called the diffusion process. Then, the process that reverses the diffusion process defined earlier is calculated. This process is the process of gradually removing noise from noise data.



(a) CIFAR-10 without data augmentation



Diffusion Process



Reverse Diffusion Process

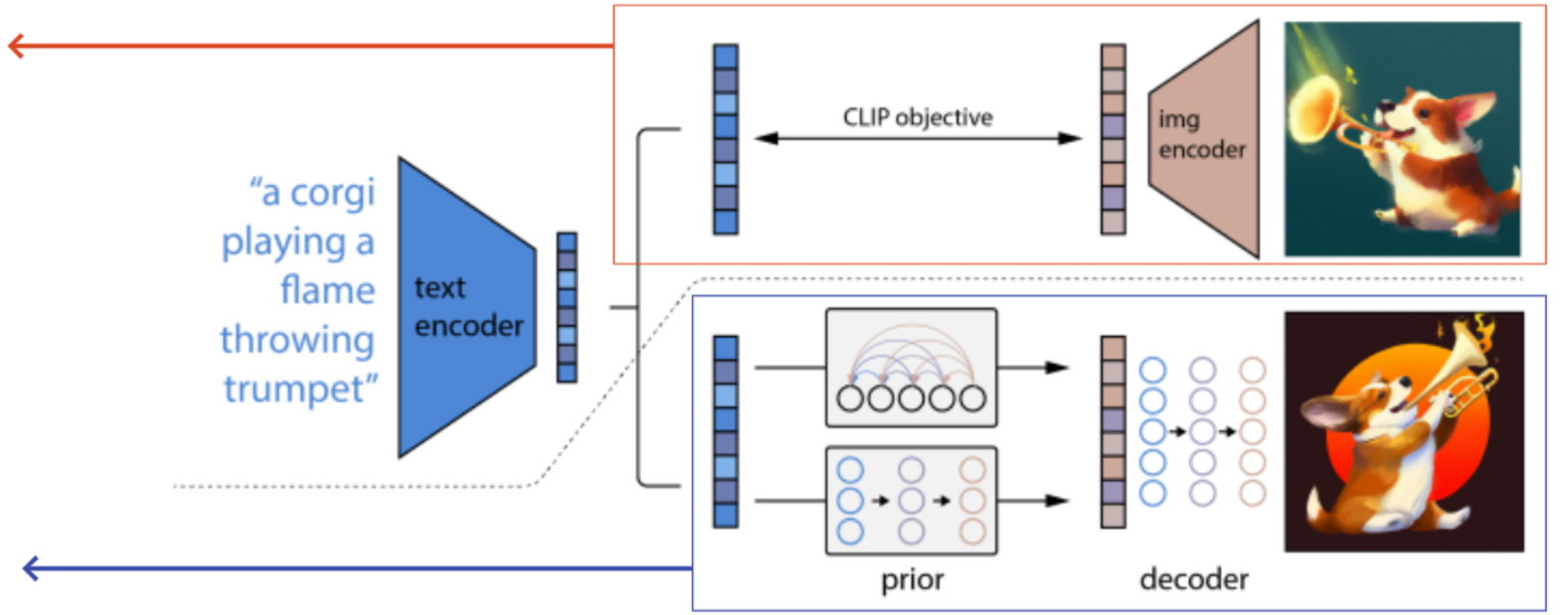
tAILab.

A total assembly of the SOTA : DALLE-2

- unCLIP

  - Above the dotted line, we depict the CLIP training process, through which we learn a joint representation space for text and images. Below the dotted line, we depict our text-to-image generation process:

CLIP : Learn the joint representation space of text images



Text-to-image generation process : The image embedding is generated from the text encoder of CLIP into the prior, the embedding is converted into the final image through the decoder.

Proposed Method : unCLIP

A total assembly of the SOTA : DALLE-2

- Notations
  - x : image, y : captions
  - z(i) = CLIP image embedding, z(t) : CLIP text embedding

- A *prior* $P(z_i|y)$ that produces CLIP image embeddings z_i conditioned on captions y.
- A *decoder* $P(x|z_i, y)$ that produces images x conditioned on CLIP image embeddings z_i (and optionally text captions y).

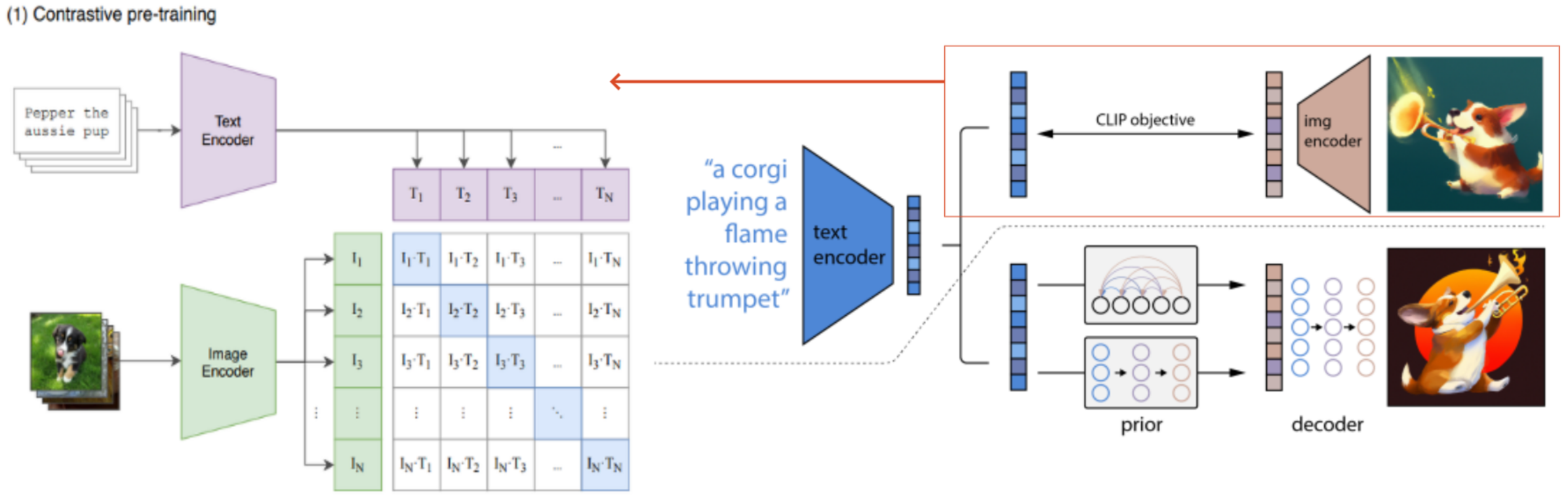
- The decoder allows us to invert images given their CLIP image embeddings, while the prior allows us to learn a generative model of the image embeddings themselves. Stacking these two components yields a generative model P(x|y) of images x given captions y:

$$P(x|y) = P(x, z_i|y) = P(x|z_i, y)P(z_i|y).$$

Model 1. CLIP

- CLIP

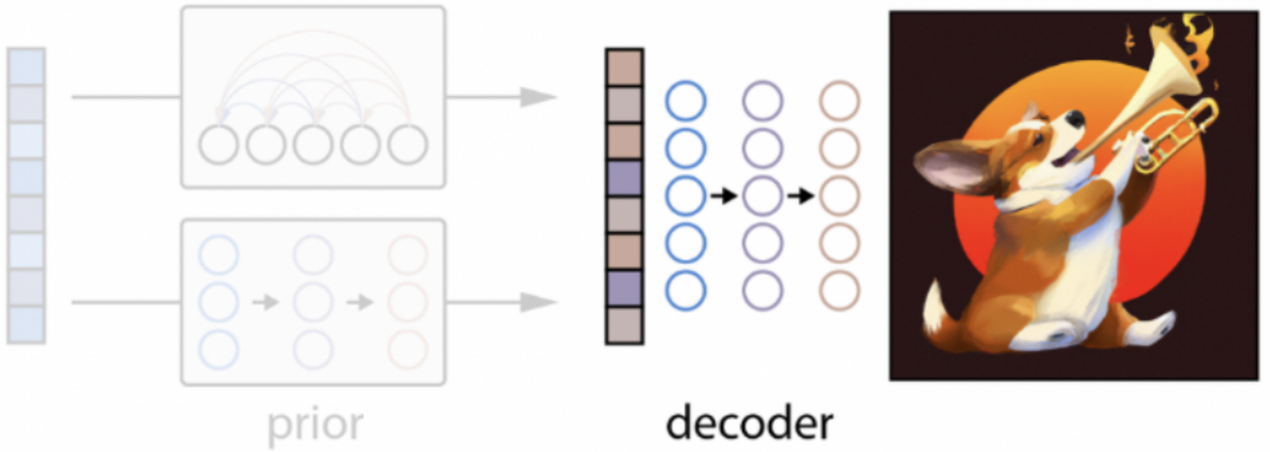
  - Typical vision datasets are labor intensive and costly to create while teaching only a narrow set of visual co ncepts; models that perform well on benchmarks have disappointingly poor performance on stress tests



Model 2. Decoder

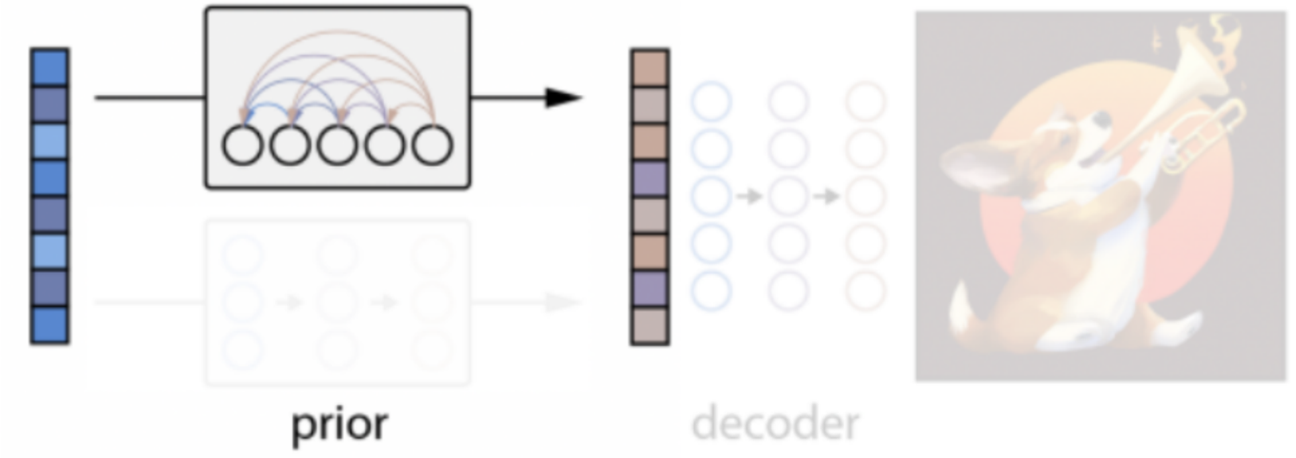
- Create an image by using the diffusion model and receiving CLIP image embedding made as a condition
- The architecture of GLIDE has been modified. Two diffusion upsampler models (ADMNet) were trained to create high-resolution images.

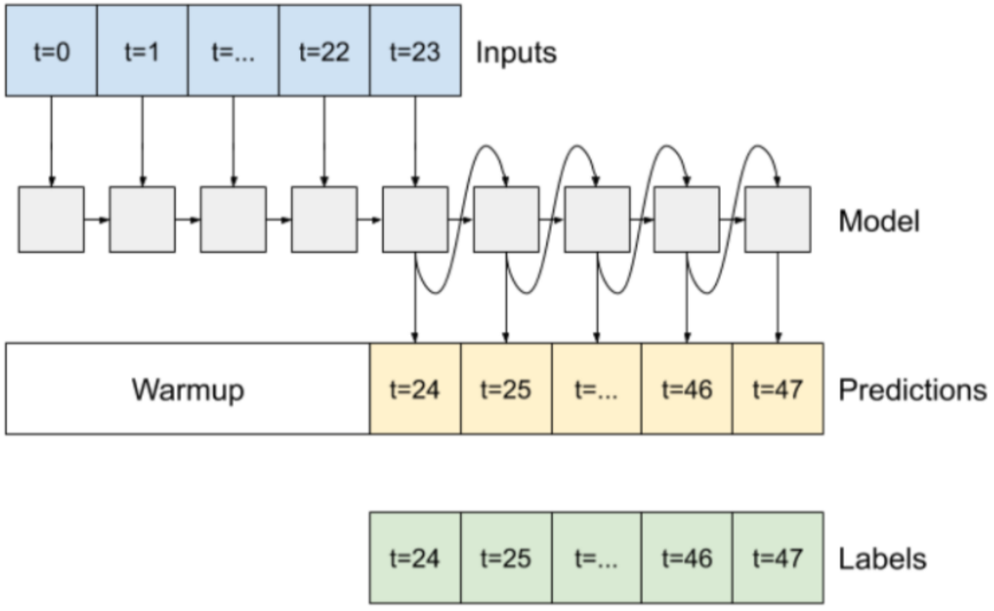
- A *prior* $P(z_i|y)$ that produces CLIP image embeddings z_i conditioned on captions y. ⟶ Prior
- A *decoder* $P(x|z_i, y)$ that produces images x conditioned on CLIP image embeddings z_i (and optionally text captions y). ⟶ Decoder



Model 3-1. Upper Prior : Autoregressive (AR) prior

- The text y is transformed into a sequence of discrete codes to form CLIP image embedding z(i). It is predicted to be autoregressively.
- The dimensionality of z(i) is reduced with PCA to efficiently learn and extract AR priors.
- The rank of the representation space was reduced by learning with the SAM optimizer, and most of the informa tion was preserved even though 319 principal components out of 1024 were maintained.



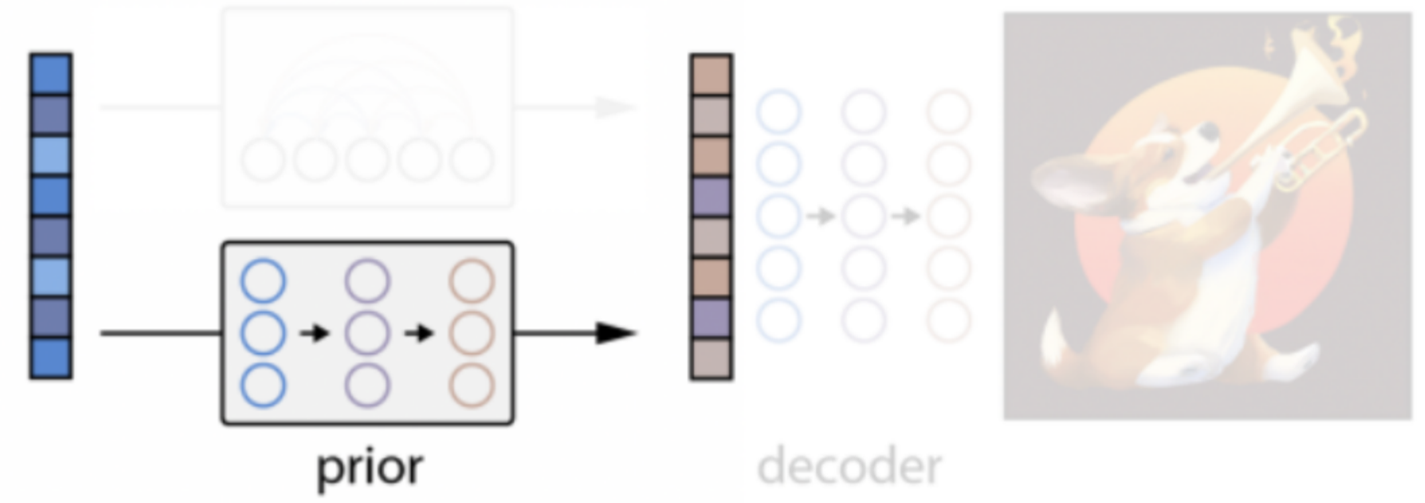


Autoregressive model example : LSTM

Model 3-2. Lower Prior : Diffusion prior

- A continuous vector z(i) is constructed through a Gaussian diffusion model in which the caption y is given as a condition. (decoder-only Transformer)
- In order : the encoded text, the CLIP text embedding, an embedding for the diffusion timestep, the noised CLIP image embedding, and a final embedding whose output from the Transformer is used to predict the unnoised CLIP image embedding.

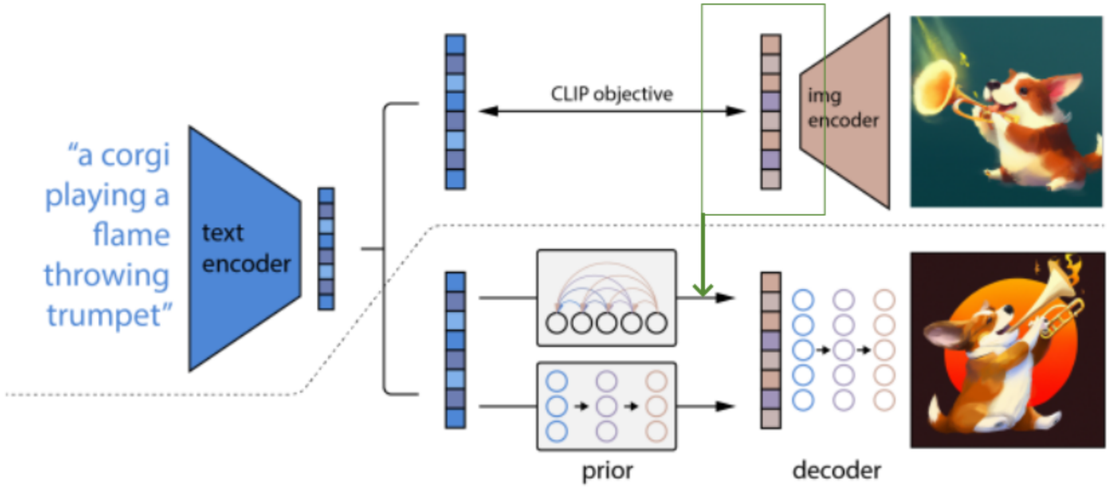
$$L_{\text{prior}} = \mathbb{E}_{t\sim[1,T],z_i^{(t)}\sim q_t}\left[\|f_\theta(z_i^{(t)},t,y) - z_i\|^2\right]$$



03 Experiments

Image Manipulations

- Encode the image x generated by upCLIP into a bipartite latent representation {z(i),x(T)} and manipulate this l atent space to create a new image



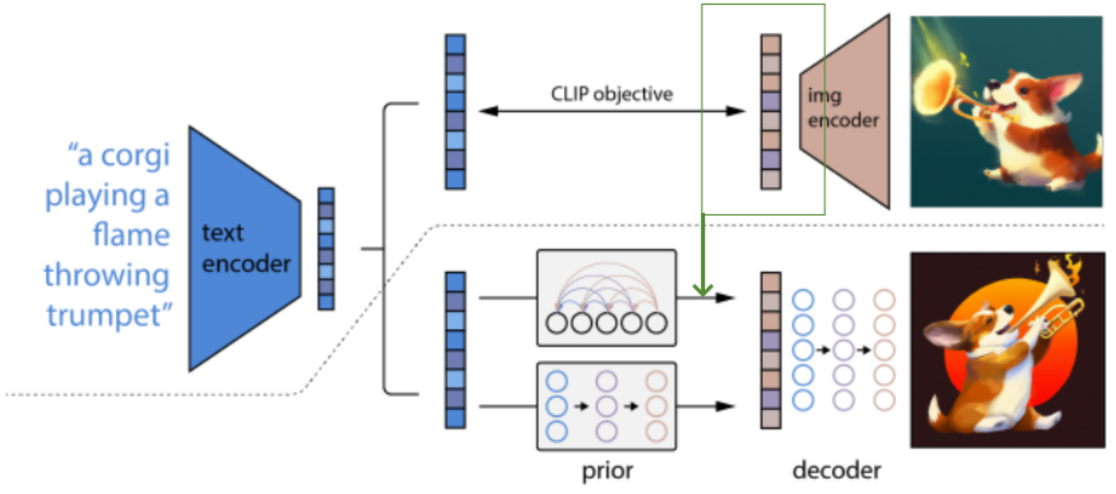
3.1 Variations

Figure 3: Variations of an input image by encoding with CLIP and then decoding with a diffusion model. The variations preserve both semantic information like presence of a clock in the painting and the overlapping strokes in the logo, as well as stylistic elements like the surrealism in the painting and the color gradients in the logo, while varying the non-essential details.

03 Experiments

Image Manipulations

- Encode the image x generated by upCLIP into a bipartite latent representation {z(i),x(T)} and manipulate this l atent space to create a new image



3.1 Variations

Figure 3: Variations of an input image by encoding with CLIP and then decoding with a diffusion model. The variations preserve both semantic information like presence of a clock in the painting and the overlapping strokes in the logo, as well as stylistic elements like the surrealism in the painting and the color gradients in the logo, while varying the non-essential details.

03 Experiments

3.2 Interpolations

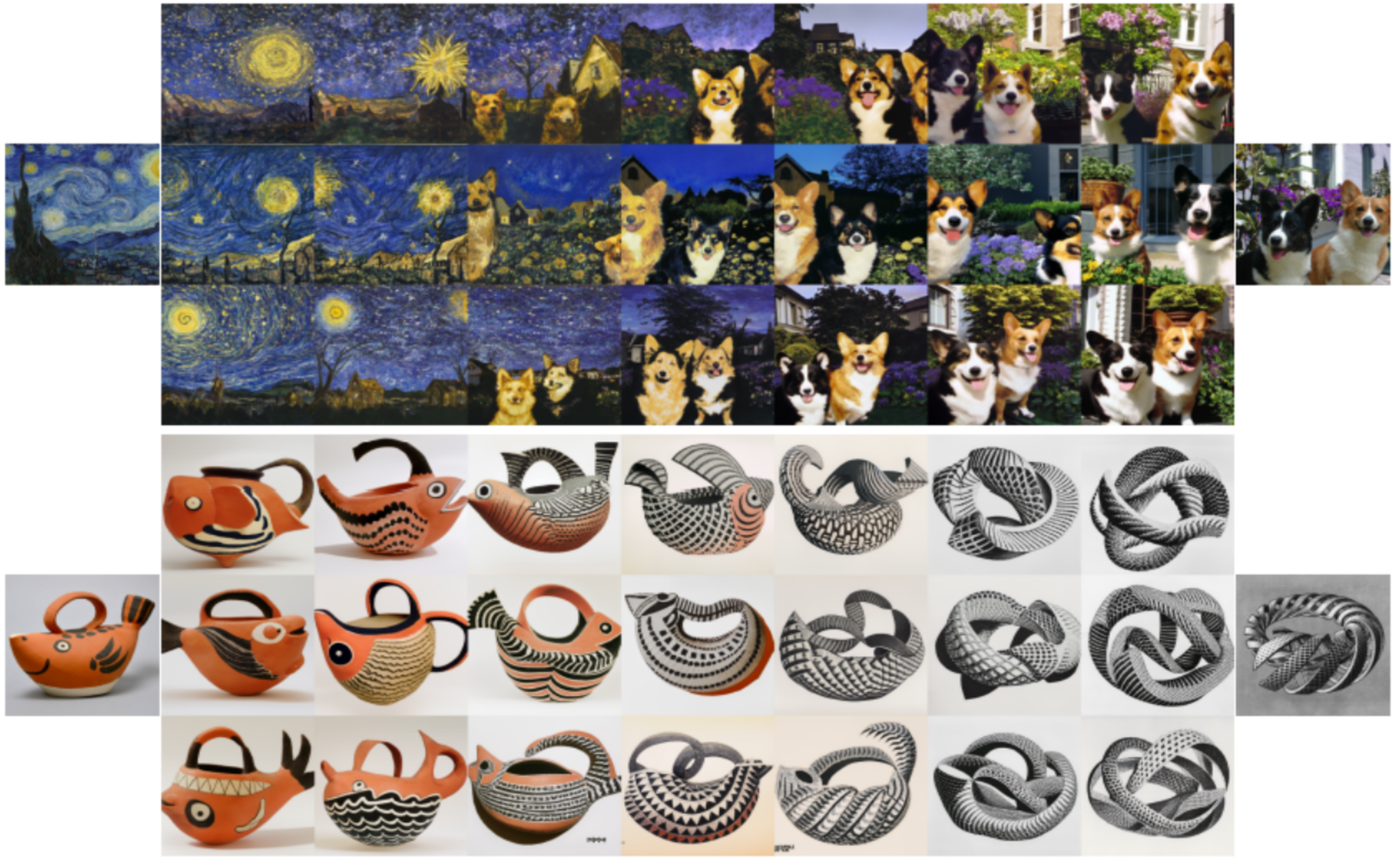
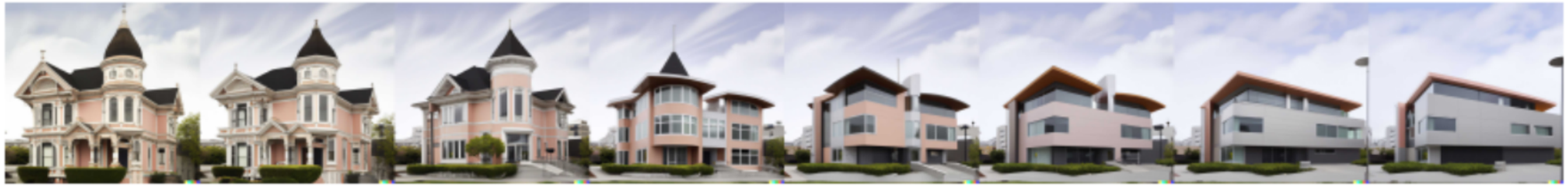


Figure 4: Variations between two images by interpolating their CLIP image embedding and then decoding with a diffusion model. We fix the decoder seed across each row. The intermediate variations naturally blend the content and style from both input images.

03 Experiments

3.3 Text Differences

a photo of a cat → an anime drawing of a super saiyan cat, artstation



a photo of a victorian house → a photo of a modern house

a photo of an adult lion → a photo of lion cub

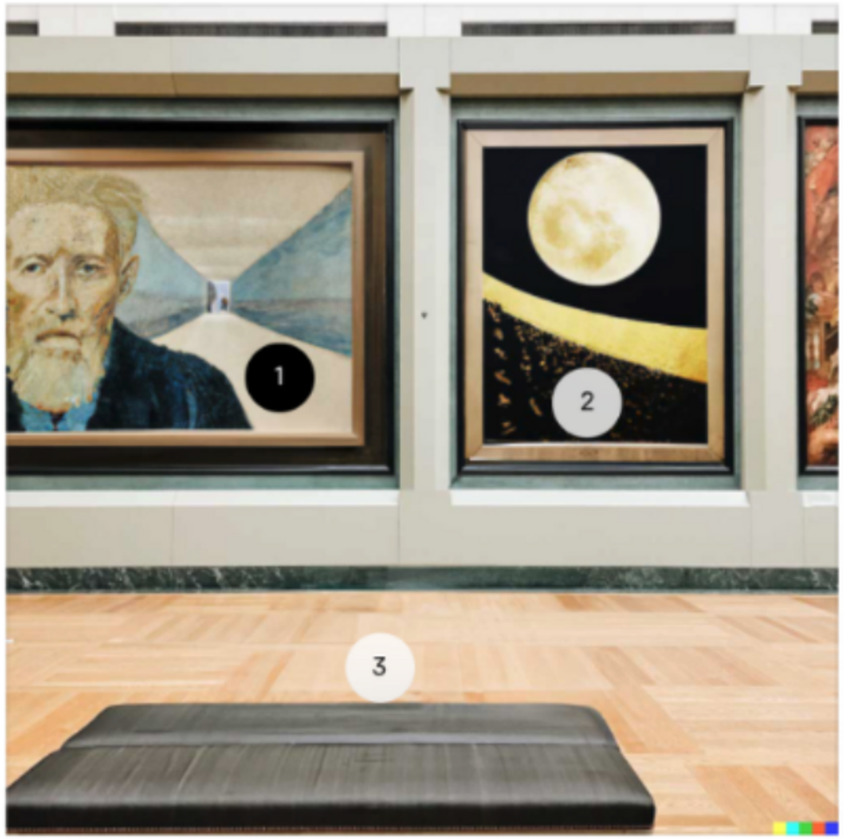
a photo of a landscape in winter → a photo of a landscape in fall

Figure 5: Text diffs applied to images by interpolating between their CLIP image embeddings and a normalised difference of the CLIP text embeddings produced from the two descriptions. We also perform DDIM inversion to perfectly reconstruct the input image in the first column, and fix the decoder DDIM noise across each row.

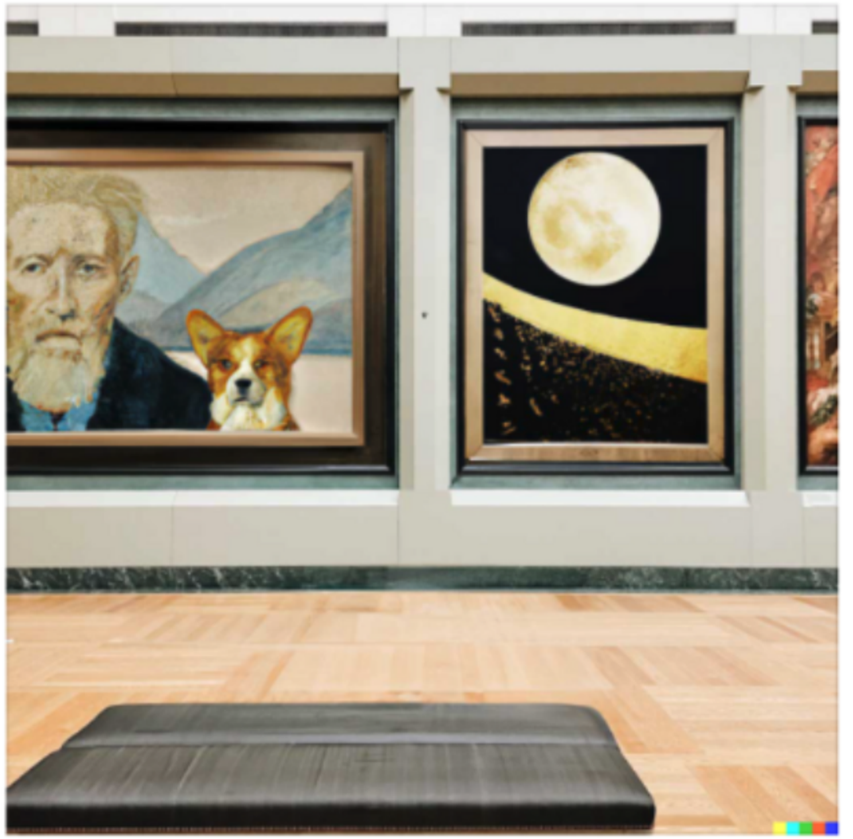
03 Experiments

3.4 Text conditional image Impainting

ORIGINAL IMAGE



DALL·E 2 EDITS



03 Experiments

3.4 Text conditional image Impainting

"a girl hugging a corgi on a pedestal"

"a corgi wearing a bow tie and a birthday hat"

"a cozy living room"

"a painting of a corgi on the wall above a couch"

"a round coffee table in front of a couch"

"a vase of flowers on a coffee table"

"a couch in the corner of a room"

Figure 3. Iteratively creating a complex scene using GLIDE. First, we generate an image for the prompt "a cozy living room", then use the shown inpainting masks and follow-up text prompts to add a painting to the wall, a coffee table, and a vase of flowers on the coffee table, and finally to move the wall up to the couch.

03 Experiments

4. DALLE-2 (unCLIP) results

5. More results

https://openai.com/dall-e-2/

vibrant portrait painting of Salvador Dalí with a robotic half face

a shiba inu wearing a beret and black turtleneck

a close up of a handpalm with leaves growing from it

an espresso machine that makes coffee from human souls, artstation

panda mad scientist mixing sparkling chemicals, artstation

a corgi's head depicted as an explosion of a nebula

a dolphin in an astronaut suit on saturn, artstation

a propaganda poster depicting a cat dressed as french emperor napoleon holding a piece of cheese

a teddy bear on a skateboard in times square