CVPR2024 Best Papers

Generative Image Dynamics
Rich Human Feedback for Text-to-Image Generation

Suk-Hwan Lee

Artificial Intelligence

Creating the Future

Dong-A University

Division of Computer Engineering &
Artificial Intelligence

References

Top papers from CVPR 2024 : A Comprehensive Overview

- Medium : Djohra IBERRAKEN
- https://medium.com/@djohraiberra/top-papers-from-cvpr-2024-comprehensive-overview-7cd32398fc41

- More than 11,532 papers were submitted, with only 2,719 papers accepted, making it an overall acceptance rate of **23.6%.**

Table of Contents:

References

1. Best paper award winners

- **Generative Image Dynamics** (Zhengqi Li et al.☞ Google Research) : https://openaccess.thecvf.com/content/CVPR2024/papers/Li_Generative_Image_Dynamics_CVPR_2024_paper.pdf

- **Rich Human Feedback for Text-to-Image Generation** (Y Liang et al. ☞ 1University of California, San Diego, 2Google Research) : https://openaccess.thecvf.com/content/CVPR2024/papers/Liang_Rich_Human_Feedback_for_Text-to-Image_Generation_CVPR_2024_paper.pdf

2. Vision Transformers

- **Learning CNN on ViT: A Hybrid Model to Explicitly Class-specific Boundaries for Domain Adaptation** (Ngo, Do-Tran et al. ☞ Chonnam National University) : https://openaccess.thecvf.com/content/CVPR2024/papers/Ngo_Learning_CNN_on_ViT_A_Hybrid_Model_to_Explicitly_Class-specific_CVPR_2024_paper.pdf

- **You Only Need Less Attention at Each Stage in Vision Transformers** (Zhang, liu et al. ☞ 1Huazhong University of Science and Technology, 2Microsoft Research Asia) : https://openaccess.thecvf.com/content/CVPR2024/papers/Zhang_You_Only_Need_Less_Attention_at_Each_Stage_in_Vision_CVPR_2024_paper.pdf

References

3. Diffusion Models

- **GenTron: Diffusion Transformers for Image and Video Generation** (Chen, Xu et al. ☞ 1The University of Hong Kong 2Meta): https://openaccess.thecvf.com/content/CVPR2024/papers/Chen_GenTron_Diffusion_Transformers_for_Image_and_Video_Generation_CVPR_2024_paper.pdf

4. Large Language Models

- **Large Language Models are Good Prompt Learners for Low-Shot Image Classification** (Zheng et al. ☞ University of Southern California): https://openaccess.thecvf.com/content/CVPR2024/papers/Zheng_Large_Language_Models_are_Good_Prompt_Learners_for_Low-Shot_Image_CVPR_2024_paper.pdf

- **SEED-Bench-2: Benchmarking Multimodal Large Language Models** (Li, Ge et al. ☞ Tencent AI Lab) : https://arxiv.org/pdf/2311.17092

- **Hallucination Augmented Contrastive Learning for Multimodal Large Language Model** (☞ Jiang et al. Peking University, Alibaba Group) : https://openaccess.thecvf.com/content/CVPR2024/papers/Jiang_Hallucination_Augmented_Contrastive_Learning_for_Multimodal_Large_Language_Model_CVPR_2024_paper.pdf

References

1. Best paper award winners

- **Generative Image Dynamics** (Zhengqi Li et al.☞ Google Research) : https://openaccess.thecvf.com/content/CVPR2024/papers/Li_Generative_Image_Dynamics_CVPR_2024_paper.pdf

- **Rich Human Feedback for Text-to-Image Generation** (Y Liang et al. ☞ 1University of California, San Diego, 2Google Research) : https://openaccess.thecvf.com/content/CVPR2024/papers/Liang_Rich_Human_Feedback_for_Text-to-Image_Generation_CVPR_2024_paper.pdf

2. Vision Transformers

- Learning CNN on ViT: A Hybrid Model to Explicitly Class-specific Boundaries for Domain Adaptation (Ngo, Do-Tran et al.)
- You Only Need Less Attention at Each Stage in Vision Transformers (Zhang, liu et al.)

3. Diffusion Models

- GenTron: Diffusion Transformers for Image and Video Generation (Chen, Xu et al.):

4. Large Language Models

- Large Language Models are Good Prompt Learners for Low-Shot Image Classification (Zheng et al.):
- SEED-Bench-2: Benchmarking Multimodal Large Language Models (Li, Ge et al.) :
- Hallucination Augmented Contrastive Learning for Multimodal Large Language Model (Jiang et al.) :

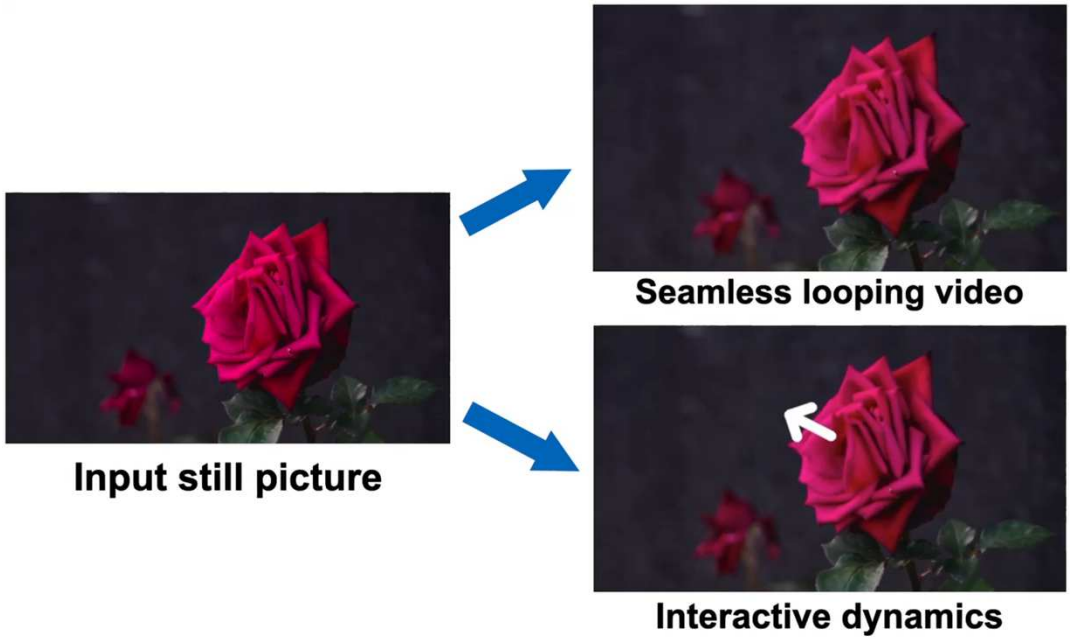
Generative Image Dynamics

Zhengqi Li, Richard Tucker, Noah Snavely, Aleksander Holynski
Google Research
pp. 24142-24153
CVPR2024

https://generative-dynamics.github.io/

- A new approach to **producing photorealistic animations from a single image** - **Modeling an image-space prior on scene motion**.
- Train our prior model from a **collections of motion trajectories extracted from real video sequences**.
- Model this **dense and long-term motion prior** in the **Fourier domain as a spectral volume;** given a single image, our trained model uses a **frequency-coordinated diffusion sampling process to predict a spectral volume**, which can be converted into a motion texture that spans an entire video.

- Along with an image-based rendering module, these trajectories can be used for a number of downstream applications;
- **Image-to-video** : by animating a still image.
- **Seamless looping** : turning still images into seamlessly looping videos
- **Interactive dynamics from a single image** : allowing users to realistically interact with objects in real pictures by interpreting the **spectral volumes** as **image-space modal bases**, which **approximate object dynamics**.



Our approach models an **image-space prior on scene dynamics** that can be used to turn **a single image** into a **seamless looping video** or an **interactive dynamic scene**.

Generative Image Dynamics

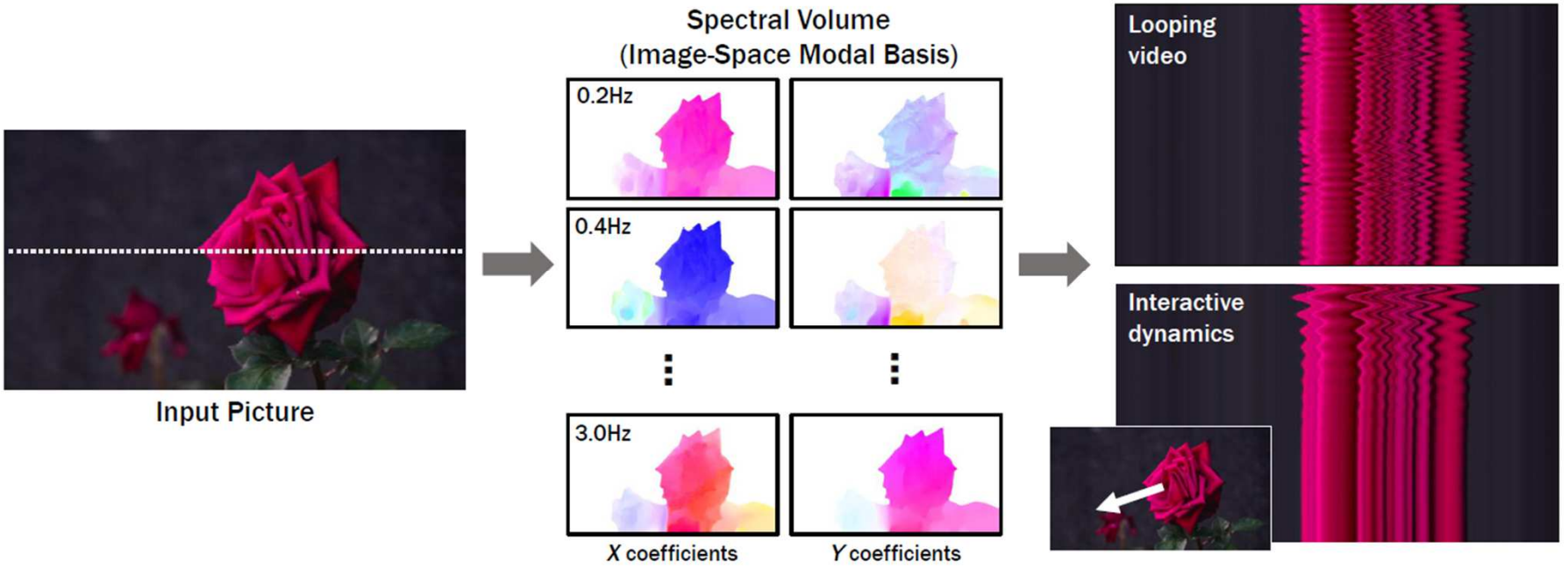


Figure 1. We model a generative image-space prior on scene motion: from a single RGB image, our method generates ***a spectral volume*** [23], a motion representation that models dense, long-term pixel trajectories in the Fourier domain. Our learned motion priors can be used to turn a single picture into a seamlessly looping video, or into an interactive simulation of dynamics that responds to user inputs like dragging and releasing points. On the right, we visualize output videos as space-time X-*t* slices (along the input scanline shown on the left).

Generative Image Dynamics

https://generative-dynamics.github.io/

Our method automatically turns single still images into seamless looping videos.

Seamless looping video






Generative Image Dynamics

https://generative-dynamics.github.io/

Our method automatically turns single still images into seamless looping videos.

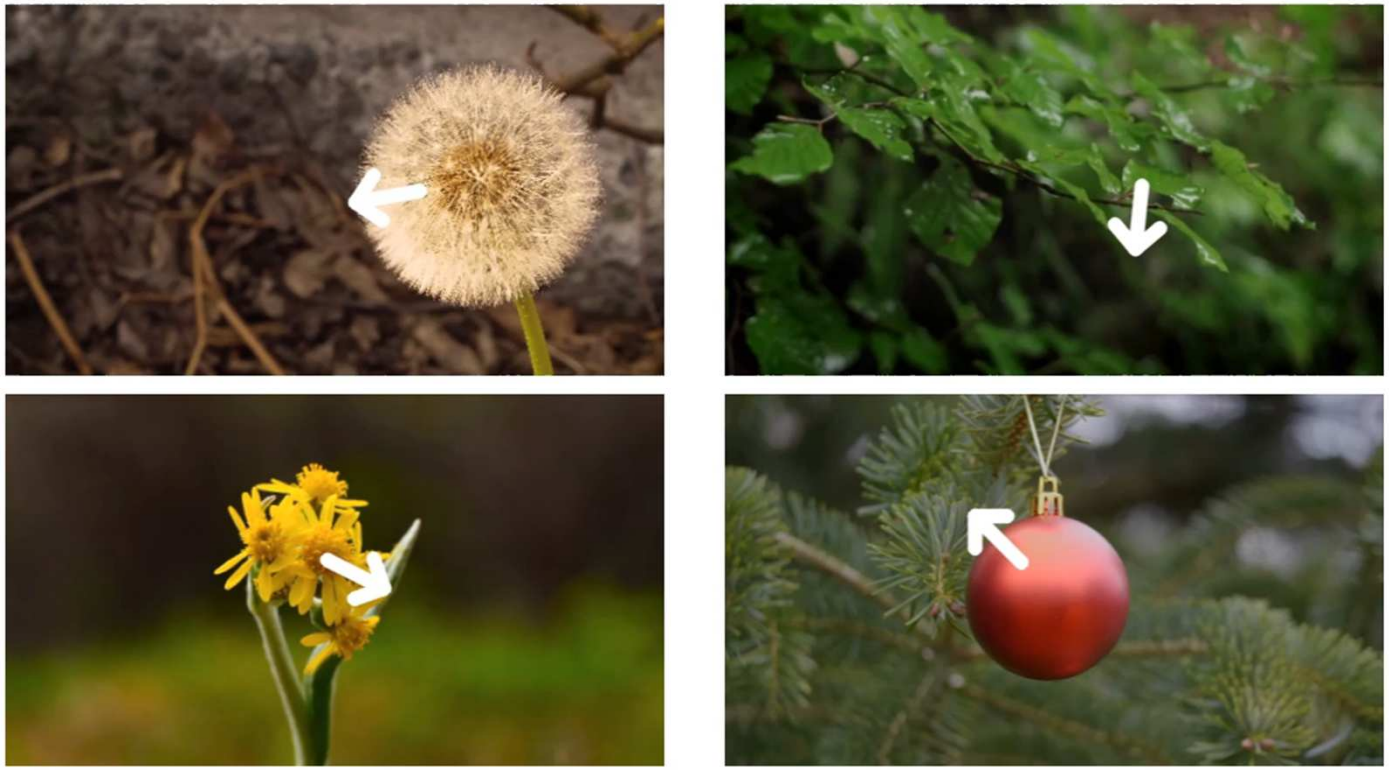
Seamless looping video

Generative Image Dynamics

https://generative-dynamics.github.io/

We can simulate the response of object dynamics to an interactive user excitation using modal analsysis by Davis et al., interpreting generated spectrum volume as image-space modal basis.

Interactive Dynamics



Generative Image Dynamics

➢ Overview

- Given a single picture I_0, our goal is to generate a video $\{\hat{I}_1, \hat{I}_2, \cdots, \hat{I}_T\}$ featuring oscillatory motions such as those of trees, flowers, or candle flames swaying in the breeze.

- Our system consists of two modules: **a motion prediction module** and **an image-based rendering module**.

➢ Motion prediction module

- Begins by using a **latent diffusion model (LDM)** to **predict a spectral volume** $S = (S_{f_0}, S_{f_1}, \cdots, S_{f_{K-1}})$ for the input I_0.
- Then transform the predicted spectral volume to **a motion texture** $\mathrm{F} = (F_1, F_2, \cdots, F_T)$ through an inverse discrete Fourier transform.
- This motion determines the position of each input pixel at every future time step.

➢ Image-based rendering module

- Given a predicted motion texture, we then animate the input RGB image using a **neural image-based rendering technique** (Sec. 5).
- We explore applications of this method, including producing **seamless looping animations** and simulating **interactive dynamics**, in Sec. 6..

➢ Prediction Motion

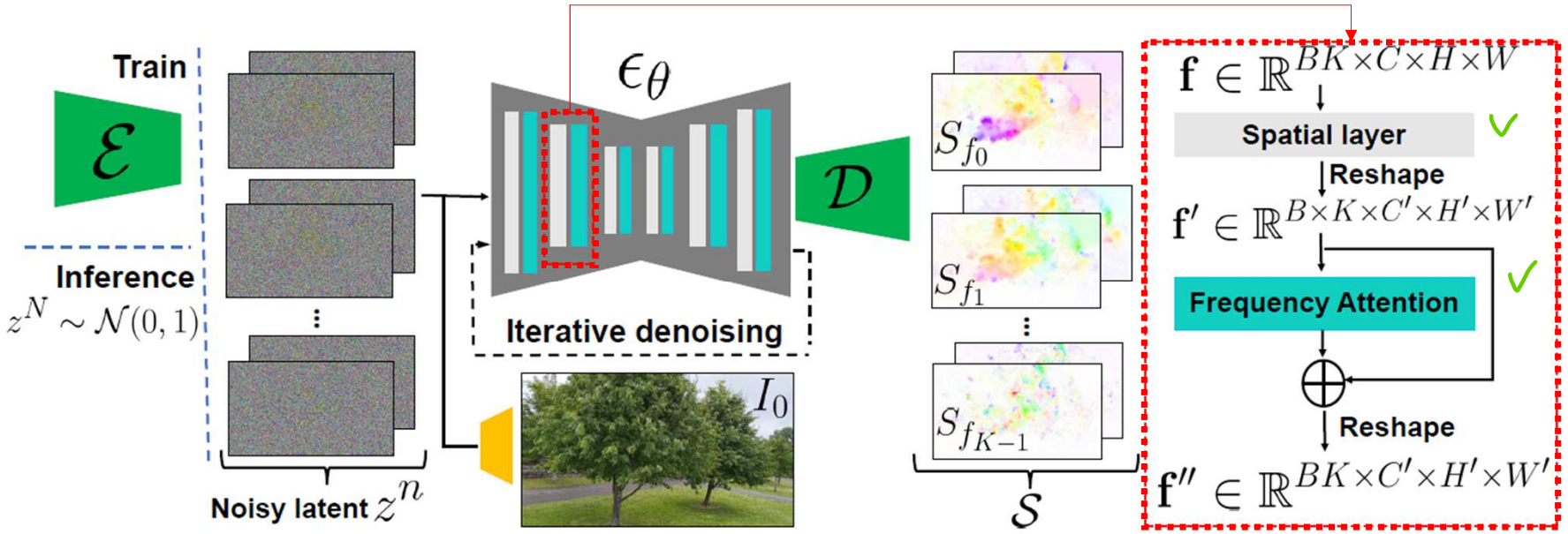


Figure 3. **Motion prediction module**. We predict a spectral volume S through a frequency-coordinated denoising model. Each block of the diffusion network ϵ_θ interleaves 2D spatial layers with attention layers (**red box, right**), and iteratively denoises latent features z^n. The denoised features are fed to a decoder D to produce S. During training, we concatenate the downsampled input I_0 with noisy latent features encoded from a real motion texture via an encoder E, and replace the noisy features with Gaussian noise z^N during inference (**left**).

Generative Image Dynamics

1) Motion representation

- A **motion texture** is **a sequence of time-varying 2D displacement maps**

$$\mathrm{F} = \{F_t | t = 1, \dots, T\}$$

- where the 2D displacement vector $F_t(\mathbf{p})$ at each pixel coordinate $\mathbf{p}$ from input image I_0 defines the position of that pixel at a future time t [20].
- To generate a future frame at time t, one can splat pixels from I_0 using the corresponding displacement map D_t, resulting in a forward-warped image I'_t;

$$I'_t(\mathbf{p} + F_t(\mathbf{p})) = I_0(\mathbf{p}). \quad (1)$$

❖ The size of the motion texture would **need to scale with the length of the video**: Generating T output frames implies predicting T displacement fields.

- We adopt from Davis et al. [23] an efficient frequency space representation of motion in a video called a *spectral volume*, visualized in Fig. 3.
- A **spectral volume** is the **temporal Fourier transform of per-pixel trajectories** extracted from a video.

Generative Image Dynamics

1) Motion representation

- Formulate the **motion prediction problem** as a **multi-modal image-to-image translation task**: from an **input image** to an **output motion spectral volume**.

- Adopt **latent diffusion models (LDMs)** to generate **spectral volumes** comprised of a **4K-channel 2D motion spectrum map**, where K << T is the number of frequencies modeled, and where at each frequency we need **four scalars** to represent the complex Fourier coefficients for the *x*- and *y*-dimensions.

- The motion trajectory of a pixel at future time steps.

$$\mathrm{F}(\mathbf{p}) = \{F_t(\mathbf{p}) | t = 1, \dots, T\}$$

- its representation as a spectral volume

$$\mathrm{S}(\mathbf{p}) = \{S_{F_k}(\mathbf{p}) | k = 0,1, \dots, \frac{T}{2} - 1\}$$

- They are related by the Fast Fourier transform (FFT)

$$\mathcal{S}(\mathbf{p}) = \mathrm{FFT}(\mathcal{F}(\mathbf{p})). \quad (2)$$

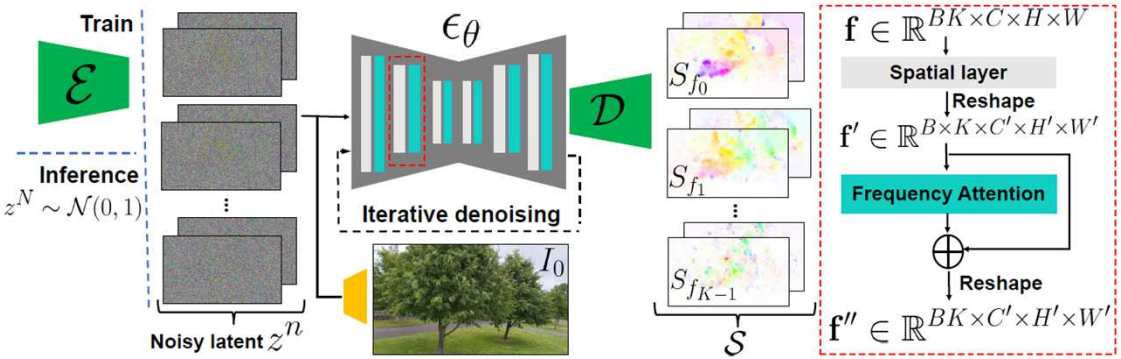
- The first K = 16 Fourier coefficients are sufficient to realistically reproduce the original natural motion in a range of real videos and scenes

Generative Image Dynamics

2) Predicting motion with a diffusion model

❖ A standard LDM consists of two main modules

- (1) a variational autoencoder (VAE) compresses the input image to a latent space through an encoder z = E(I), then reconstructs the input from the latent features via a decoder I = D(z)
- (2) a U-Net based diffusion model that learns to iteratively denoise features starting from Gaussian noise.



- Our training process applies to **spectral volumes** from real video sequences, which are encoded and then diffused for *n* steps with a **pre-defined variance schedule** to produce noisy latents z^n.
- 2D U-Nets are trained to denoise the noisy latents by iteratively estimating the noise $\epsilon_\theta(z^n; n; c)$ used to update the latent feature at each step $n \in (1,2,\dots,N)$.

- The training loss for the LDM

$$\mathcal{L}_{\text{LDM}} = \mathbb{E}_{n\in\mathcal{U}[1,N],\epsilon^n\in\mathcal{N}(0,1)}\left[||\epsilon^n - \epsilon_\theta(z^n; n, c)||^2\right] \quad (3)$$

- c is the embedding of any conditional signal, such as text, or, in our case, the first frame of the training video sequence, I_0.
- The clean latent features z_0 are then passed through the decoder to recover the spectral volume.

Generative Image Dynamics

Frequency adaptive normalization

- As **diffusion models** require that the absolute values of the output are between **-1 and 1 for stable training and denoising** [44], we must normalize the coefficients of S extracted from real videos before using them for training

➢ Effective frequency adaptive normalization method

① We independently normalize Fourier coefficients at each frequency based on statistics computed from the training set. Namely, for each individual frequency f_j, we **compute the 95th percentile of Fourier coefficient magnitudes over all input samples** and use that value as **a per-frequency scaling factor** s_{f_j} .

② We then apply **a power transformation to each scaled Fourier coefficient** to pull it away from extreme values.

③ The final coefficient values of spectral volume $S(\mathbf{p})$ at frequency f_j (used for training our LDM) are computed as

$$S'_{f_j}(\mathbf{p}) = \text{sign}(S_{f_j}) \sqrt{\left| \frac{S_{f_j}(\mathbf{p})}{s_{f_j}} \right|}. \quad (4)$$

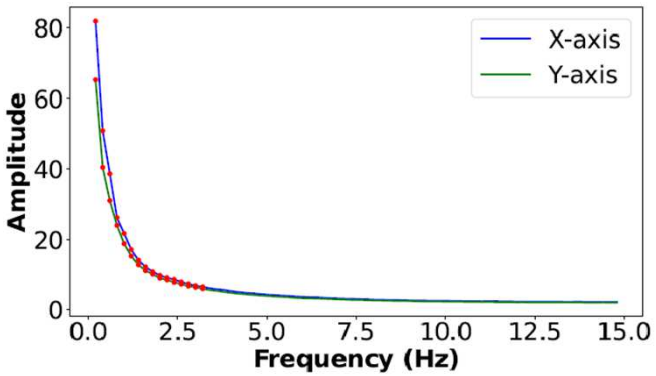


Figure 2. **Left**: We visualize the average power spectrum for the X and Y motion components extracted from real videos, shown as the blue and green curves. Natural oscillation motions are composed primarily of low-frequency components, and so we use the first K = 16 terms, marked with red dots.

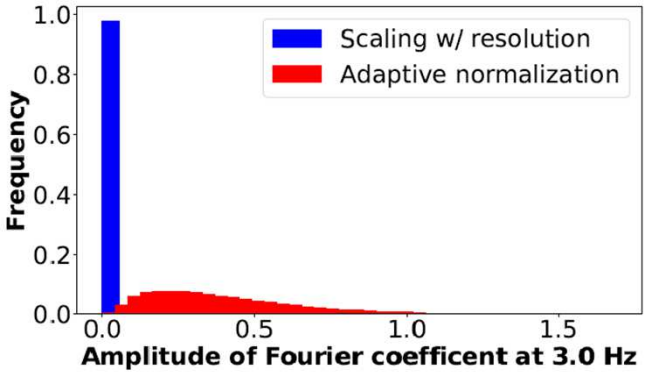


Figure 2. **Right**: we show a histogram of the amplitude of Fourier terms at 3:0 Hz after (1) scaling amplitude by image width and height (blue), or (2) frequency adaptive normalization (red). **Our adaptive normalization prevents the coefficients from concentrating at extreme values.**

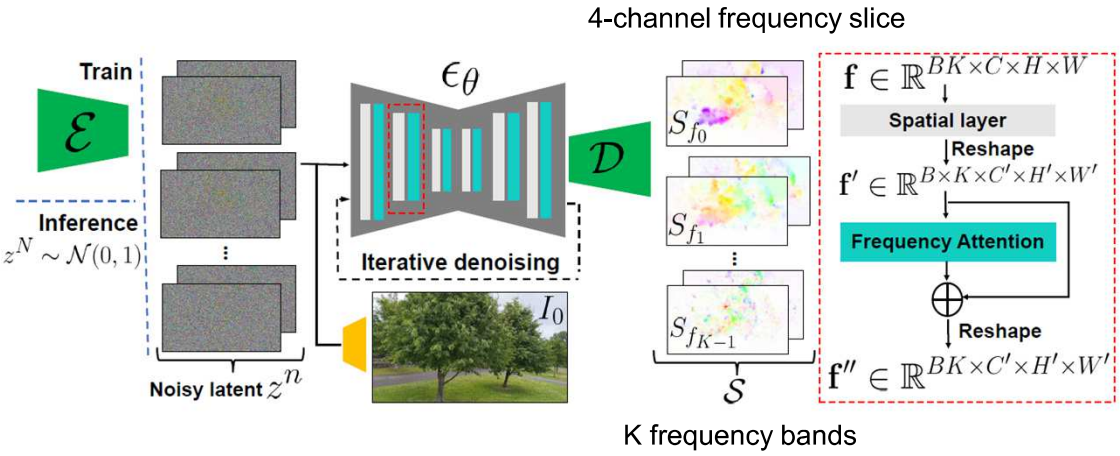
Generative Image Dynamics

Frequency-coordinated denoising

- The straightforward way to **predict a spectral volume S with K frequency bands** is to **output a tensor of 4K channels from a single diffusion U-Net**.

✓ Training a model to produce a large number of channels → Yield **Over-smoothed**, **Inaccurate outputs**

✓ Predict independently each individual frequency slice → **Uncorrelated predictions** in the frequency domain, leading to **unrealistic motion**



➢ A frequency-coordinated denoising strategy

- Given an input image I_0, we first **train an LDM ϵ_θ to predict a single 4-channel frequency slice of spectral volume S_{f_j}**, where we inject an **extra frequency embedding** along with **the time-step embedding** into the LDM.
- We then **freeze the parameters** of this LDM ϵ_θ, introduce **attention layers** interleaved with the 2D spatial layers of ϵ_θ across the K **frequency bands**, and **fine-tune.**
- Specifically, for a batch size B, the 2D spatial layers of ϵ_θ treat the corresponding $B \cdot K$ **noisy latent features of channel size** C as independent samples with shape $R^{(B\cdot K)\times C\times H\times W}$.
- The attention layer then interprets these as consecutive features spanning the frequency axis, and we reshape the latent features from previous 2D spatial layers to $R^{B\times K\times C\times H\times W}$ before feeding them to the attention layers.
- In other words, the frequency attention layers are fine-tuned to coordinate all frequency slices so as to produce coherent spectral volumes

Generative Image Dynamics

➢ Image Based Rendering

- Describe how we take a spectral volume S predicted for a given input image I_0 and render a future frame $\hat{I}_t$ at time t.

- We first derive a motion texture in the time domain using the inverse temporal FFT applied at each pixel

$$\mathcal{F}(\mathbf{p}) = \mathrm{FFT}^{-1}(\mathcal{S}(\mathbf{p}))$$

- To produce **a future frame $\hat{I}_t$**, we adopt **a deep image-based rendering technique** and perform **splatting with the predicted motion field F_t** to **forward-warp the encoded I_0**,

√ Forward-warping – lead to hole

√ Multiple source pixels - map to the same output 2D location

→ Adopt the **feature pyramid softmax splatting strategy** proposed in prior work on frame interpolation [67]

- With the **motion field F_t** and **weights W**, we apply **softmax splatting** to **warp the feature map at each scale** to produce a warped feature. The warped features are then injected into the corresponding blocks of an image synthesis decoder to produce a final rendered image $\hat{I}_t$.

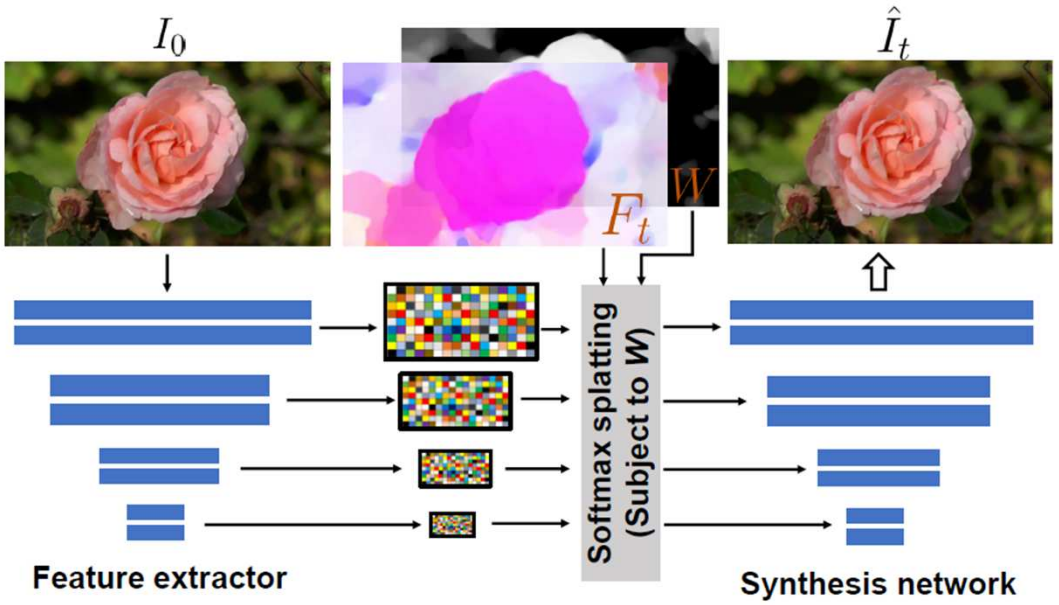


Figure 4. **Rendering module**. We fill in missing content and refine the warped input image using a deep image-based rendering module, where multi-scale features are extracted from the input image I_0. Softmax splatting is then applied over the features with a motion field F_t from time 0 to t (subject to the weights W). The warped features are fed to an image synthesis network to produce the rendered image $\hat{I}_t$.

Generative Image Dynamics

➢ Image Based Rendering

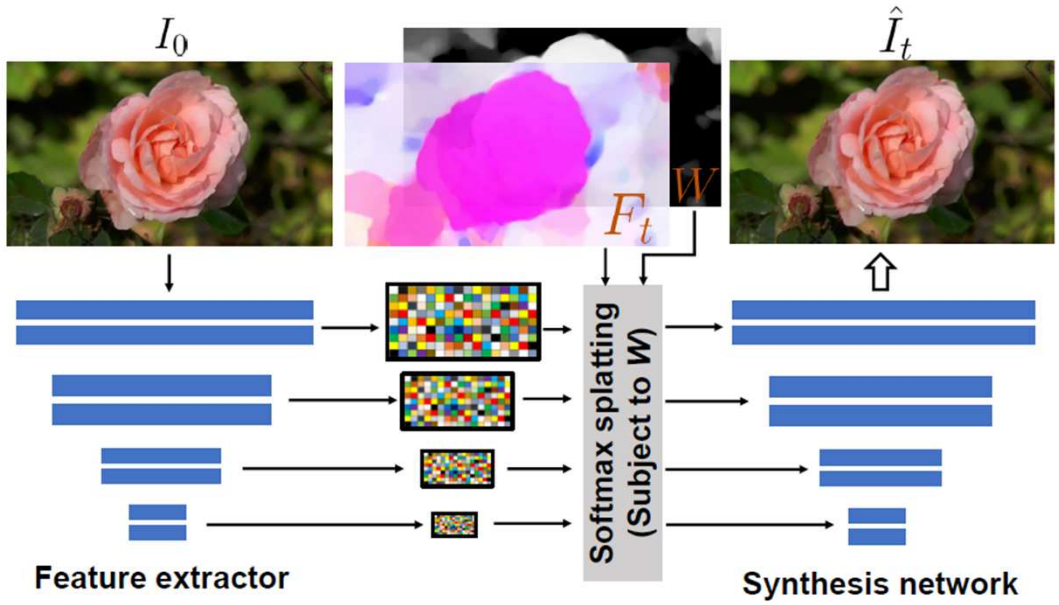
❖ Feature pyramid softmax splatting strategy

- We encode I_0 through a feature extractor network to produce a multi-scale feature map.
- For each individual feature map at scale j, we resize and scale the predicted 2D motion field F_t according to the resolution.

- As in Davis et al. [22], we use **predicted flow magnitude** as a **proxy for depth** to determine **the contributing weight of each source pixel** mapped to its destination location.
- We compute a per-pixel weight $W(\mathbf{p})$ as the average magnitude of the predicted motion texture.

$$W(\mathbf{p}) = \frac{1}{T}\sum_t ||F_t(\mathbf{p})||_2$$

- Assume **large motions** correspond to **moving foreground objects**, and small or zero motions correspond to background.
- We use **motion-derived weights** instead of learnable ones as [46] because learnable weights are not effective for addressing disocclusion ambiguities.



- We jointly train the feature extractor and synthesis networks with start and target frames (I_0, I_t) randomly sampled from real videos, using the estimated flow field from I_0 to I_t to warp encoded features from I_0, and supervising predictions $\hat{I}_t$ against I_t with a VGG perceptual loss [49].

Generative Image Dynamics

➢ Applications

❖ Image-to-Video

- Enables the animation of a single still picture
- By first predicting a motion spectral volume from the input image and generating an animation by applying our image-based rendering module to the motion texture transformed from the spectral volume.

❖ Seamless looping ; Motion Self-guidance

- No discontinuity between the start and end of the video
- Our method is a ***motion self-guidance*** technique that **guides the motion denoising sampling processing** using **explicit looping constraints**.
- At each iterative denoising step during inference, we incorporate an **additional motion guidance signal alongside standard classifier-free guidance** [45], where we enforce each pixel's position and velocity at the start and end frames to be as similar as possible:

$$\hat{\epsilon}^n = (1+w)\epsilon_\theta(z^n; n, c) - w\epsilon_\theta(z^n; n, \emptyset) + u\sigma^n \nabla_{z^n}\mathcal{L}_g^n$$
$$\mathcal{L}_g^n = ||F_T^n - F_1^n||_1 + ||\nabla F_T^n - \nabla F_1^n||_1 \quad (5)$$

- F_t^n is the predicted 2D displacement field at time t and denoising step n. w is the classifier-free guidance weight, and u is the motion self-guidance weight

➢ Applications

❖ Interactive dynamics from a single image

- Davis et al. [22] show that the **spectral volume** can approximate **an image-space modal basis** that is **a projection of the vibration modes of the underlying scene** (or, more generally, captures spatial and temporal correlations in oscillatory dynamics), and can be used to **simulate the object's response to a user-defined force**.

- By this modal analysis method [22, 69],
- the **image-space 2D motion displacement field** for the object's physical response as a weighted sum of motion spectrum coefficients S_{f_j} modulated by the state of complex modal coordinates $\mathbf{q}_{f_j}(t)$ at each simulated time step t:

$$F_t(\mathbf{p}) = \sum_{f_j} S_{f_j}(\mathbf{p})\mathbf{q}_{f_j}(t) \quad (6)$$

- We simulate the state of the modal coordinates $\mathbf{q}_{f_j}(t)$ via an explicit Euler method applied to the equations of motion for a decoupled mass-spring-damper system represented in modal space [22, 23, 69].

❖ Note that our method produces an interactive scene from a single picture, whereas these prior methods required a video as input.

➢ Experimental Results

Table 1. **Quantitative comparisons on the test set**. We report both image synthesis and video synthesis quality. Here, KID is scaled by 100. Lower is better for all error. See Sec. 7.1 for descriptions of baselines and error metrics.

Method	Image Synthesis FID	KID	Video Synthesis FVD	FVD_{32}	DTFVD	$DTFVD_{32}$
TATS [35]	65.8	1.67	265.6	419.6	22.6	40.7
Stochastic I2V [27]	68.3	3.12	253.5	320.9	16.7	41.7
MCVD [92]	63.4	2.97	208.6	270.4	19.5	53.9
LFDM [66]	47.6	1.70	187.5	254.3	13.0	45.6
DMVFN [48]	37.9	1.09	206.5	316.3	11.2	54.5
Endo *et al.* [29]	10.4	0.19	166.0	231.6	5.35	65.1
Holynski *et al.* [46]	11.2	0.20	179.0	253.7	7.23	46.8
Ours	**4.03**	**0.08**	**47.1**	**62.9**	**2.53**	**6.75**

Table 2. Ablation study. Sec. 7.3 describes each configuration.

Method	Image Synthesis FID	KID	Video Synthesis FVD	FVD_{32}	DTFVD	$DTFVD_{32}$
Repeat I_0	-	-	237.5	316.7	5.30	45.6
$K = 4$	3.92	0.07	60.3	78.4	3.12	8.59
$K = 8$	3.95	0.07	52.1	68.7	2.71	7.37
$K = 24$	4.09	0.08	48.2	65.1	2.50	6.94
w/o adaptive norm.	4.53	0.09	62.7	80.1	3.16	8.19
Independent pred.	4.00	0.08	52.5	71.3	2.70	7.40
Volume pred.	4.74	0.09	53.7	71.1	2.83	7.79
Baseline splat [46]	4.25	0.09	49.5	66.8	2.83	7.27
Full ($K = 16$)	4.03	0.08	47.1	62.9	2.53	6.75

Rich Human Feedback for Text-to-Image Generation

Youwei Liang*†1 et al.
1University of California, San Diego, 2Google Research, 3University of Southern California, 4University of Cambridge, 5Brandeis University
pp. 19401-19411
CVPR2024

https://research.google/blog/rich-human-feedback-for-text-to-image-generation/

- **Text-to-Image (T2I)** generation models (Stable Diffusion, Imagen etc)
  - Generating high-resolution images based on text descriptions.
  - Suffer from issues such as **artifacts/implausibility**, **misalignment with text descriptions**, and **low aesthetic quality**.

- **Reinforcement Learning with Human Feedback (RLHF)** for large language models
  - Collect human-provided scores as feedback on generated images
  - Train a reward model to improve the T2I generation.

- This paper
  - We enrich the **feedback signal** by
    1) Marking image regions that are implausible or misaligned with the text
    2) Annotating which words in the text prompt are misrepresented or missing on the image.
  - We collect such rich human feedback on 18K generated images (**RichHF-18K**) and **train a multimodal transformer to predict the rich feedback automatically.**

- The predicted rich human feedback
  - Improve image generation, for example, by selecting high-quality training data to finetune and improve the generative models, or by creating masks with predicted heatmaps to inpaint the problematic regions.
  - Used to generate the images on which human feedback data were collected (Stable Diffusion variants).
  - RichHF-18K data set : https://github.com/google-research-datasets/richhf-18k

Rich Human Feedback for Text-to-Image Generation

https://research.google/blog/rich-human-feedback-for-text-to-image-generation/

➢ Main Contributions

1) **Rich Human Feedback dataset (RichHF-18K)** on generated images on 18K Pick-a-Pic images.

   RichHF-18K contains

   (i) point **annotations** on the image that **highlight regions** of **implausibility/artifacts**, and **text-image misalignment**

   (ii) **labeled words** on the prompts specifying **the missing or misrepresented concepts** in the generated image;

   (iii) **four types of fine-grained scores** for **image plausibility**, **text-image alignment**, **aesthetics**, and **overall rating.**

   2 heatmaps (artifact/implausibility and misalignment), 4 fine-grained scores (plausibility, alignment, aesthetics, and overall score), 1 text sequence (misaligned keywords)

2) A **multimodal Transformer model (RAHF)**

   - Learn to predict these rich human annotations on generated images and their associated text prompt.
   - RAHF can predict implausibility and misalignment regions, misaligned keywords, as well as finegrained scores.

3) Demonstrate the **usefulness** of the predicted rich human feedback by RAHF to **improve image generation**

   (i) by **using the predicted heatmaps as masks** to **inpaint problematic image regions**

   (ii) by **using the predicted scores to help finetune** image generation models (like Muse [6]), e.g., via selecting/filtering finetuning data, or as reward guidance.

4) The **improvement on the Muse model**, which differs from the models that generated the images in our training set, shows the **good generalization capacity** of our RAHF model.

Rich Human Feedback for Text-to-Image Generation

➢ Collecting rich human feedback

1) Data collection process

- ❖ RichHF-18K dataset : **2 heatmaps** (artifact/implausibility and misalignment), **4 fine-grained scores** (plausibility, alignment, aesthetics, and overall score), **1 text sequence** (misaligned keywords)
- Annotators label the misaligned keywords and the four types of scores for plausibility, image-text alignment, aesthetic, and overall quality, respectively, on a 5-point Likert scale.
- ❖ Detailed definitions of image implausibility/artifact and misalignment can be found in the supplementary materials

Figure 1. **An illustration of our annotation UI**. Annotators mark points on the image to indicate artifact/implausibility regions (red points) or misaligned regions (blue points) w.r.t the text prompt. Then, they click on the words to mark the misaligned keywords (underlined and shaded) and choose the scores for plausibility, text-image alignment, aesthetics, and overall quality (underlined).

2) Human feedback consolidation (3 annotators)

- For the **point annotations** : Convert them to heatmaps for each annotation, and then compute the average heatmap across annotators.
- For the **scores** : Average the scores from the multiple annotators
- For the **misaligned keyword annotations** : Majority voting to get the final sequence of indicators of aligned/misaligned

Rich Human Feedback for Text-to-Image Generation

3) RichHF-18K: a dataset of rich human feedback

- We select a subset of image-text pairs from the **Pick-a-Pic dataset** for data annotation.
- To have **balanced categories** across the images
  - ✓ We utilized the **PaLI visual question answering (VQA)** model [7] to extract some basic features from the Pick-a-Pic data samples.
  - ✓ We asked the following questions for each image-text pair in Pick-a-Pic.

    1) Is the image photorealistic?

    2) Which category best describes the image? Choose one in 'human', 'animal', 'object', 'indoor scene', 'outdoor scene'.

- We used **PaLI's answ**ers to sample a diverse subset from Pick-a-Pic, resulting in **17K image-text pairs**.
  - ✓ Randomly split the 17K samples into two subsets, **a training set with 16K samples** and **a validation set with 1K samples**.
- We collect **rich human feedback on the unique prompts** and **their corresponding images from the Pick-a-Pic test set** as our **test set**.
- In total, we collected rich human feedback on the **18K image-text pairs** from Pick-a-Pic. Our RichHF-18K dataset consists of **16K training, 1K validation, and 1K test samples**.

4) Data statistics of RichHF-18K

- We standardize the scores s to lie in the range [0, 1];

$$\frac{s - s_{min}}{s_{max} - s_{min}} \; (s_{max} = 5, s_{min} = 1)$$

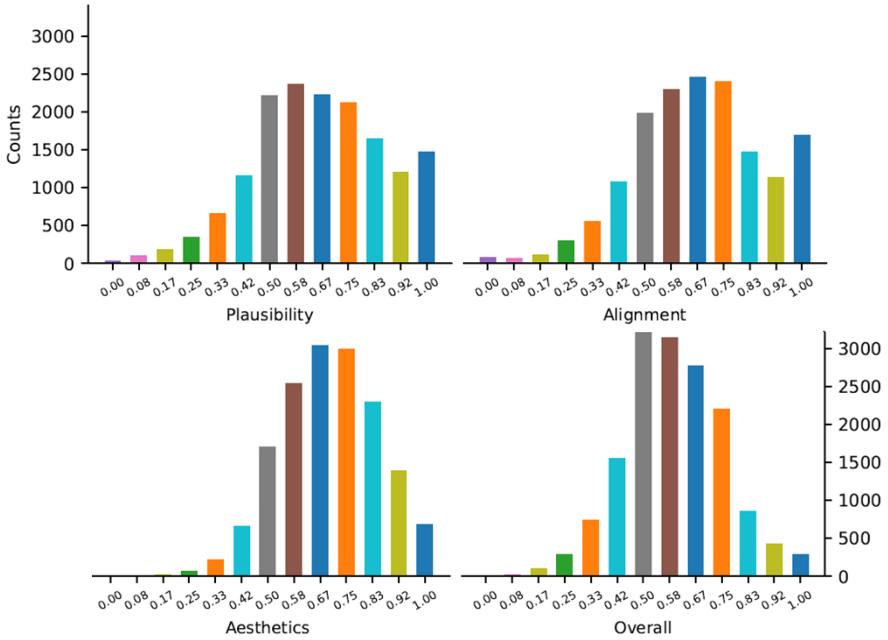


Figure 2. Histograms of the average scores of image-text pairs in the training set.

- To analyze the rating agreement among annotators for an image-text pair, we calculate the maximum difference among the scores:

$$\max_{diff} = \max(\text{scores}) - \min(\text{scores}),$$

where scores are the three score labels for an image-text pair.

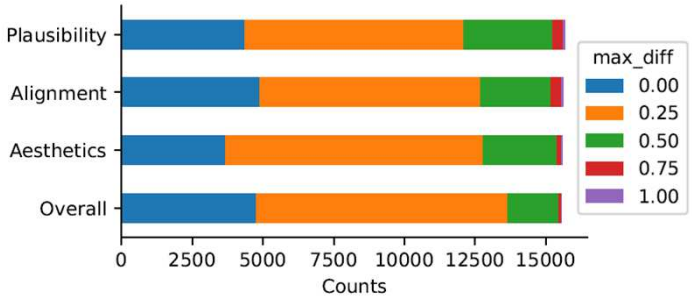


Figure 4. Counts of the samples with **maximum differences** of the scores in the training set.

- Around 25% of the samples have perfect annotator agreement and around 85% of the samples have good annotator agreement ($\max_{diff}$ is less than or equal to 0.25 after the standardization or 1 in the 5-point Likert scale).

Rich Human Feedback for Text-to-Image Generation

➢ Predicting rich human feedback

1) Model Architecture

- We adopt a **vision-language model** based on **ViT** [14] and **T5X** [40] models, inspired by the **Spotlight model architecture** [33], but modifying both the model and pretraining datasets to better suit our tasks.
- A **self-attention module** [49] among the concatenated image tokens and text tokens, similar to PaLI [7], as our tasks require **bidirectional information propagation**.
- The text information is propagated to image tokens for text misalignment score and heatmap prediction, while the vision information is propagated to text tokens for better vision-aware text encoding to decode the text misalignment sequence.
- To pretrain the model on more diverse images, we add the natural image captioning task on the **WebLI** dataset [7] to the pretraining task mixture.

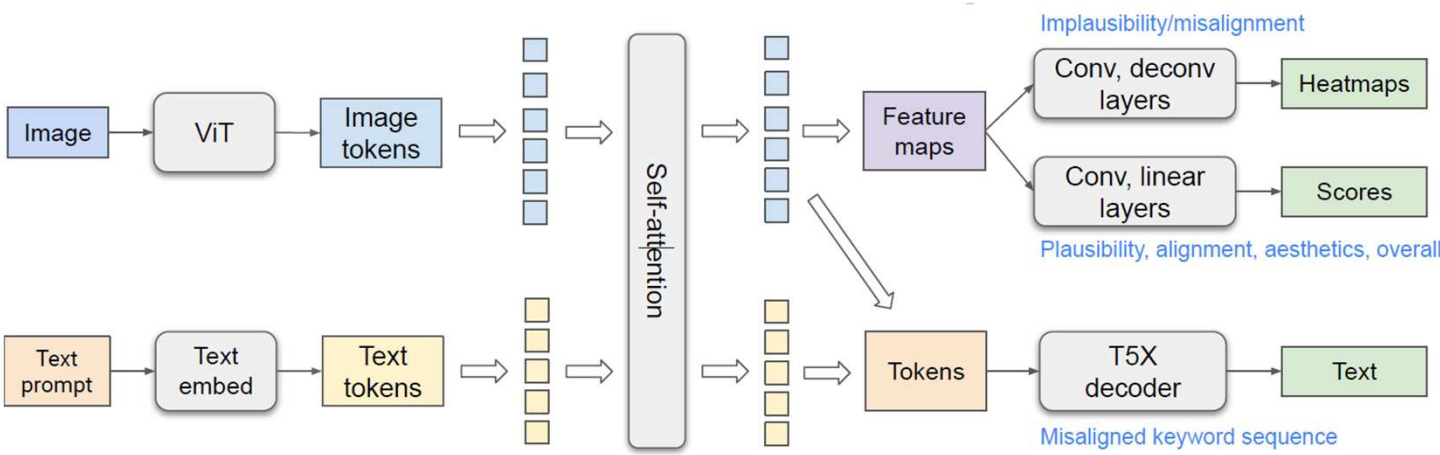


Figure 3. Architecture of our rich feedback model.

Our model consists of two streams of computation: one vision and one text stream.

We perform self-attention on the ViT-outputted image tokens and the Text-embed module-outputted text tokens to fuse the image and text information.

The vision tokens are reshaped into feature maps and mapped to heatmaps and scores. The vision and text tokens are sent to a Transformer decoder to generate a text sequence.

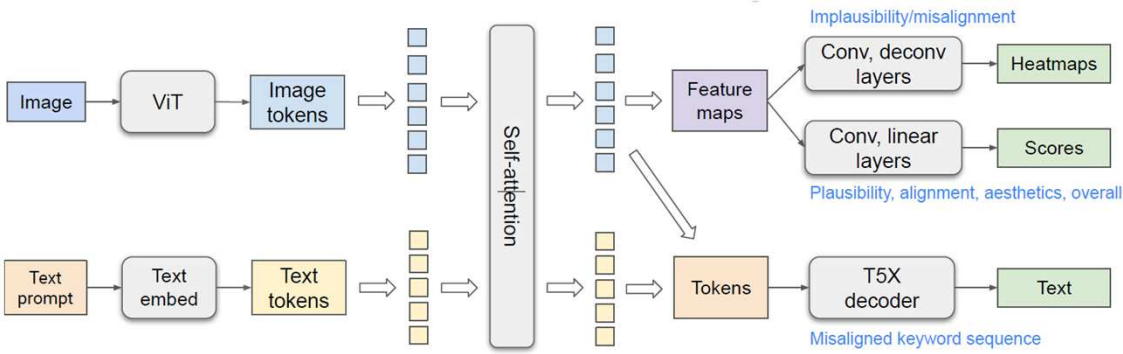
Rich Human Feedback for Text-to-Image Generation

➢ Predicting rich human feedback

1) Model Architecture

- The image tokens and embedded text tokens are concatenated and encoded by the Transformer self-attention encoder in T5X.

- On top of the encoded fused text and image tokens, we use three kinds of predictors to predict different outputs.



(1) **Predict Heatmap**

✓ the image tokens are reshaped into a feature map and sent through convolution layers, deconvolution layers, and sigmoid activation,

✓ outputs implausibility and misalignment heatmap

(2) **Predict Score**

✓ the feature map is sent through convolution layers, linear layers, and sigmoid activation

✓ resulting in scalars as fine-grained scores.

(3) **Predict the keyword misalignment sequence**

- the original prompt used to generate the image is used as text input to the model.
- A modified prompt is used as the prediction target for the T5X decoder.
  - ✓ The modified prompt has a special suffix ('_0') for each misaligned token, *e.g., a yellow_0_cat* if the generated image contains a black cat and the word yellow is misaligned with the image.
- During evaluation, we can extract the misaligned keywords using the special suffix.

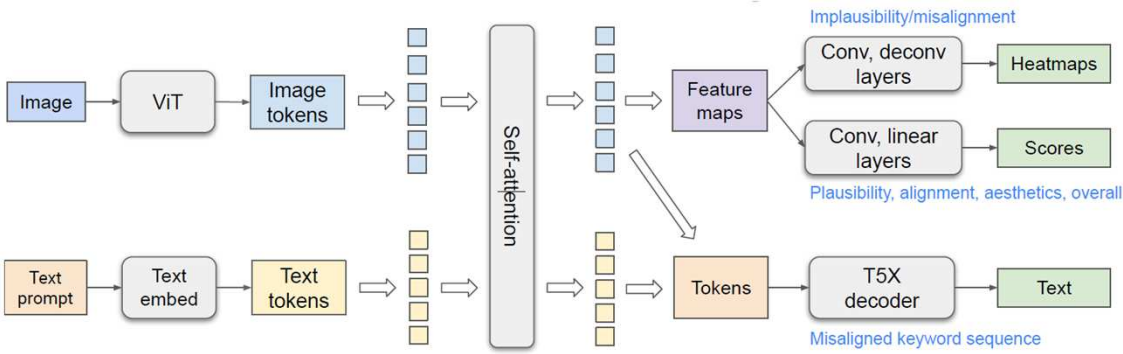
Rich Human Feedback for Text-to-Image Generation

➢ Predicting rich human feedback

2) Model Variants

[**Multi-head**]

- A straightforward way to predict multiple heatmaps and scores is to use multiple prediction heads, with one head for each score and heatmap type.
- Require seven prediction heads in total



[**Augmented prompt**]

- Another approach : Use a single head for each prediction type, i.e., three heads in total, for heatmap, score, and misalignment sequence, respectively.

- To inform the model of the fine-grained heatmap or score type, we **augment the prompt with the output type.**
  - ✓ We prepend a task string (e.g., ‘implausibility heatmap’) to the prompt for each particular task of one example and use the corresponding label as the training target.
  - ✓ During inference, by augmenting the prompt with the corresponding task string, the single heatmap (score) head can predict different heatmaps (scores).

Rich Human Feedback for Text-to-Image Generation

➢ Predicting rich human feedback

3) Model Optimization

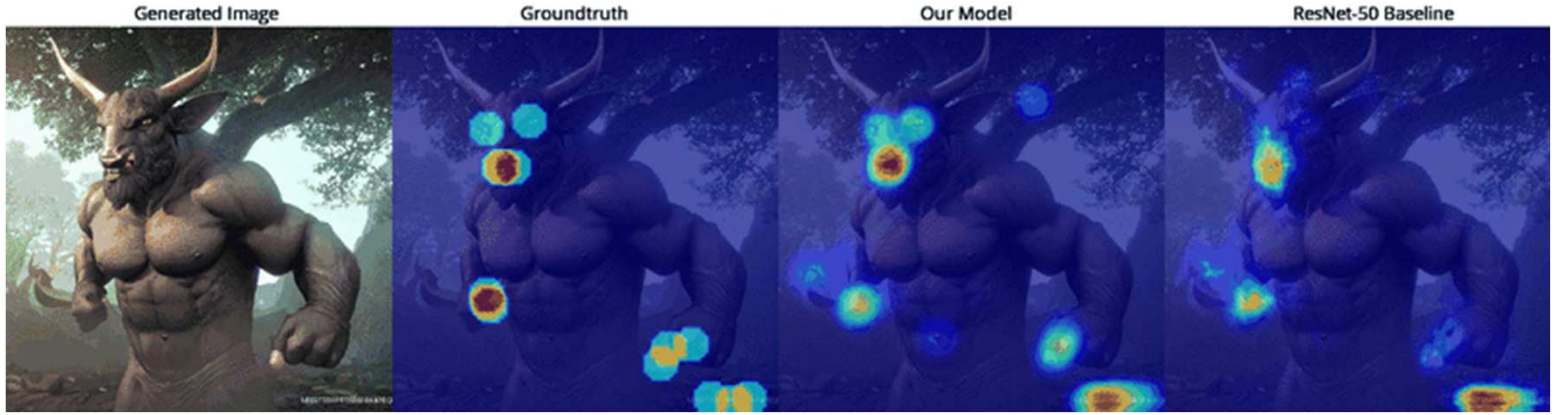
- Train the model with a pixel-wise MSE loss for the heatmap prediction and a MSE loss for the score prediction.
- Train with teacher-forcing cross-entropy loss for misalignment sequence prediction

- The final loss function is the weighted combination of the heatmap MSE loss, score MSE loss, and the sequence teacher-forcing cross-entropy loss.

Rich Human Feedback for Text-to-Image Generation

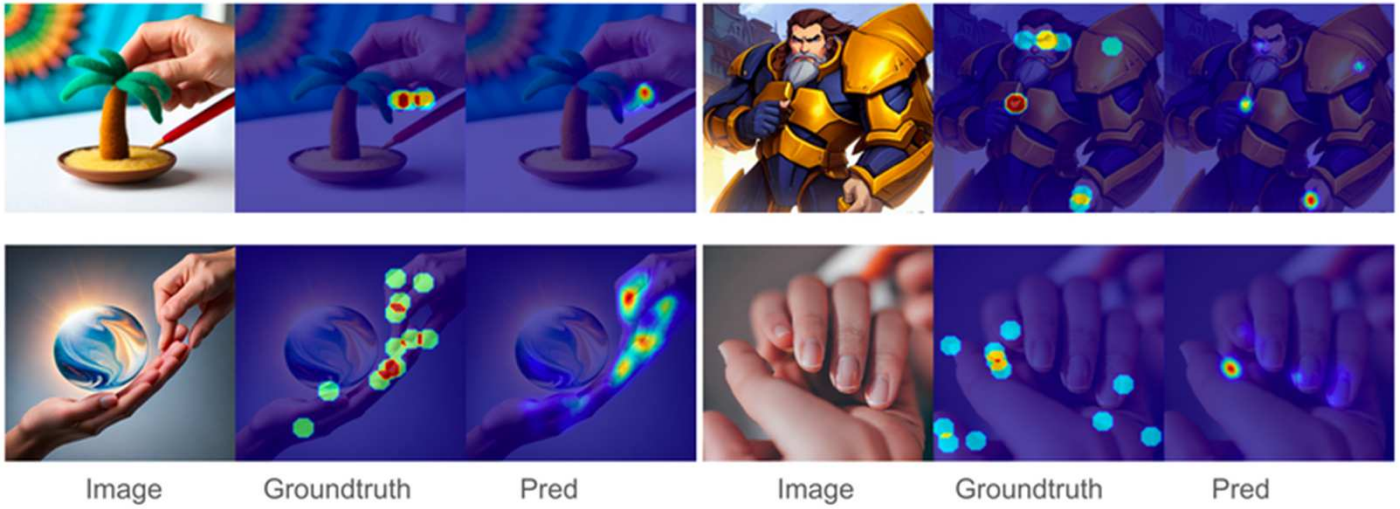
➢ **Experiments : Rich human feedback prediction**

https://research.google/blog/rich-human-feedback-for-text-to-image-generation/



Examples of **implausibility heatmaps**.

In Groundtruth heatmap, color represents how many annotators mark the region as implausible. Red/yellow/blue means 3/2/1 annotators mark the regions respectively. In prediction, color represents the signal strength (probability). The hotter one region is, the more probable the model predicts it as implausible.



Case study of **implausibility heatmaps** for human hands and fingers.

A case study on human hands (which are among the most common sources of error for generative models) and figures are shown below, which demonstrates that the model can successfully locate the artifacts for various cases. This indicates that the model learns the concept of good hands and fingers.

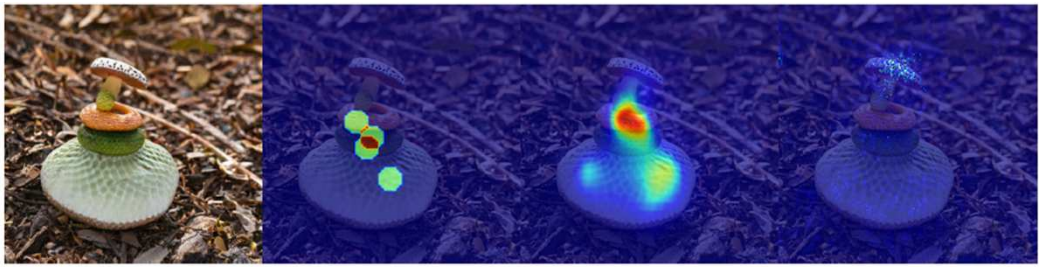
Rich Human Feedback for Text-to-Image Generation

➢ **Experiments : Rich human feedback prediction**

https://research.google/blog/rich-human-feedback-for-text-to-image-generation/

(a) Image (b) GT (c) Our model (d) ResNet-50

Figure 5. Examples of **implausibility heatmaps**. Prompt: *photo of a slim asian little girl ballerina with long hair wearing white tights running on a beach from behind nikon D5*



(a) Image (b) GT (c) Our model (d) CLIP gradient

Figure 6. Examples of **misalignment heatmaps**. Prompt: *A snake on a mushroom.*

Below is an example of our model prediction on misalignment heatmap.

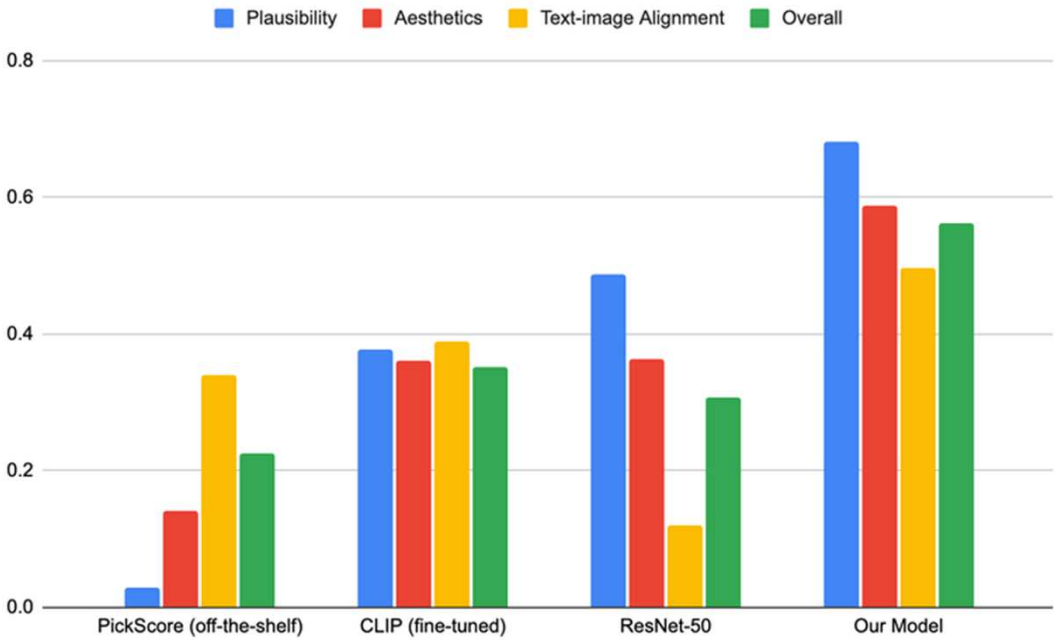
The top of the mushroom is labeled as a misalignment region as there is no snake generated. We can see our model can predict the misalignment region accurately in this example.

Rich Human Feedback for Text-to-Image Generation

➢ Experiments : Rich human feedback prediction

https://research.google/blog/rich-human-feedback-for-text-to-image-generation/

- Below we show a comparison of our model performance on score predictions vs. baseline methods. Examples for **score prediction** can be found in our paper.



Spearman correlation of score prediction results on the test set.

- An example of a generated image and the predicted human feedback, demonstrating that RAHF can serve as an evaluation tool for T2I generation, with automatic interpretation.

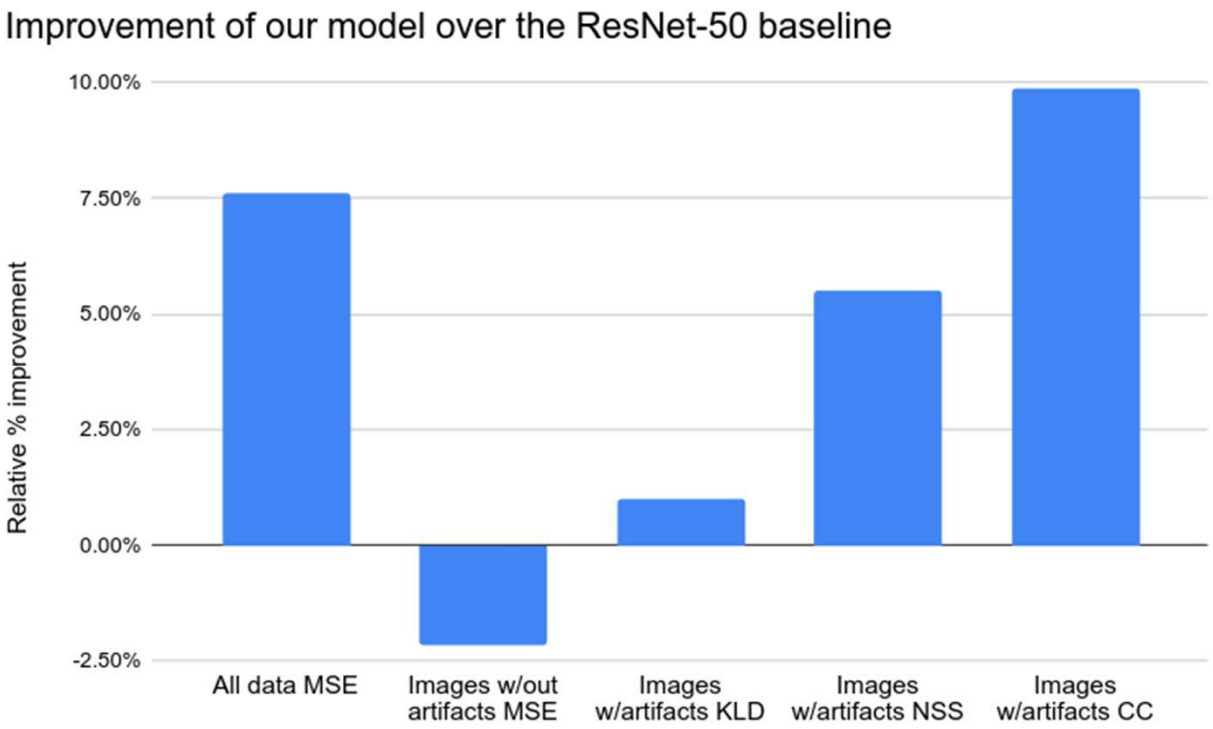
Example of one generated image and the predicted rich human feedback by RAHF.

Rich Human Feedback for Text-to-Image Generation

https://research.google/blog/rich-human-feedback-for-text-to-image-generation/

➢ Experiments : Rich human feedback prediction

- As a baseline comparison, we fine-tuned a ResNet-50 with our RichHF-18K data. Quantitative analysis below shows that our **model performs better than the baseline on most metrics for implausibility heatmap prediction**.



Implausibility heatmap prediction results on the test set.

The figure shows improvement of most metrics by our model compared with the baseline model.

MSE: mean square error

KLD: KL divergence

NSS: Normalized Scanpath Saliency

CC: correlation coefficient.

What Do Different Evaluation Metrics Tell Us About Saliency Models?, IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 41, Issue: 3, March 2019.

Rich Human Feedback for Text-to-Image Generation

➢ Experiments

	Plausibility		Aesthetics		Text-image Alignment		Overall	
	PLCC ↑	SRCC ↑	PLCC ↑	SRCC ↑	PLCC ↑	SRCC ↑	PLCC ↑	SRCC ↑
ResNet-50	0.495	0.487	0.370	0.363	0.108	0.119	0.337	0.308
PickScore (off-the-shelf)	0.0098	0.0280	0.131	0.140	0.346	0.340	0.202	0.226
CLIP (off-the-shelf)	–	–	–	–	0.185	0.130	–	–
CLIP (fine-tuned)	0.390	0.378	0.357	0.360	0.398	0.390	0.353	0.352
Our Model (multi-head)	0.666	0.654	**0.605**	**0.591**	**0.487**	**0.500**	**0.582**	0.561
Our Model (augmented prompt)	**0.693**	**0.681**	0.600	0.589	0.474	0.496	0.580	**0.562**

Table 1. Score prediction results on the test set.

- For score prediction tasks, the **Pearson linear correlation coefficient (PLCC)** and **Spearman rank correlation coefficient (SRCC),**
- For heatmap prediction tasks, MSE on all samples and on those with empty ground truth, respectively, and report saliency heatmap evaluation metrics like NSS/KLD/AUC-Judd/SIM/CC [5] for the samples with non-empty ground truth.

	All data	$GT = 0$	$GT > 0$				
	MSE ↓	MSE ↓	CC ↑	KLD ↓	SIM ↑	NSS ↑	AUC-Judd ↑
ResNet-50	0.00996	**0.00093**	0.506	1.669	0.338	2.924	0.909
Ours (multi-head)	0.01216	0.00141	0.425	1.971	0.302	2.330	0.877
Ours (augmented prompt)	**0.00920**	0.00095	**0.556**	**1.652**	**0.409**	**3.085**	**0.913**

Table 2. Implausibility heatmap prediction results on the test set. $GT = 0$ refers to empty implausibility heatmap, *i.e.*, no artifacts/implausibility (69 out of 995 test samples are empty), for ground truth. $GT > 0$ refers to heatmaps with artifacts/implausibility, for ground truth.

➢ Experiments

	All data	$GT = 0$	$GT > 0$				
	MSE ↓	MSE ↓	CC ↑	KLD ↓	SIM ↑	NSS ↑	AUC-Judd ↑
CLIP gradient	0.00817	0.00551	0.015	3.844	0.041	0.143	0.643
Our Model (multi-head)	**0.00303**	0.00015	0.206	**2.932**	0.093	1.335	0.838
Our Model (augmented prompt)	0.00304	**0.00006**	**0.212**	2.933	**0.106**	**1.411**	**0.841**

Table 3. Text misalignment heatmap prediction results on the test set. $GT = 0$ refers to empty misalignment heatmap, *i.e.*, no misalignment (144 out of 995 test samples are empty), for ground truth. $GT > 0$ refers to heatmaps with misalignment, for ground truth.

- For the misaligned keyword sequence prediction, we adopt the token-level precision, recall, and F1-score. Specifically, the precision/recall/F1 scores are computed for the misaligned keywords over all the samples.

	Precision	Recall	F1 Score
Multi-head	**62.9**	33.0	43.3
Augmented prompt	61.3	**34.1**	**43.9**

Table 4. Text misalignment prediction results on the test set.

Preference	$\gg$	$>$	$\approx$	$<$	$\ll$
Percentage	21.5%	30.33%	31.33%	12.67%	4.17%

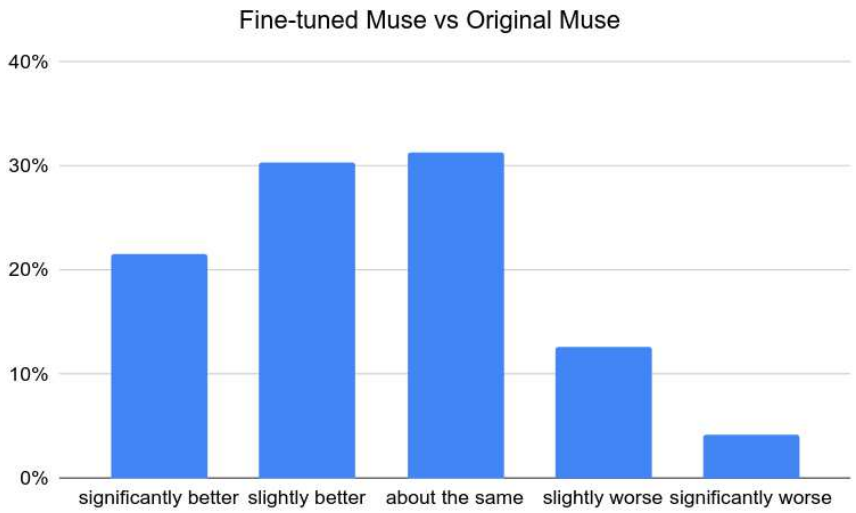
Table 5. **Human Evaluation Results: Finetuned Muse vs original Muse model preference**: Percentage of examples where finetuned Muse is significantly better ($\gg$), slightly better ($>$), about the same ($\approx$), slightly worse ($<$), significantly worse ($\ll$) than original Muse. Data was collected from 6 individuals in a randomized survey.

Rich Human Feedback for Text-to-Image Generation

➢ Experiments : Learning from rich human feedback

https://research.google/blog/rich-human-feedback-for-text-to-image-generation/

- The predicted rich human feedback (e.g., scores and heatmaps) can be used to improve image generation.
- One way is by **fine-tuning generative models with predicted scores**. To do this, we start by creating a high-quality dataset by filtering the **Muse** model results with RAHF-predicted scores. The Muse model is then fine-tuned with this dataset, via **LoRA** fine-tuning method. Side-by-side evaluation showed that Muse fine-tuned with RAHF plausibility scores possesses significantly fewer artifacts than the original Muse, as shown by the examples and results below.



- Muse: Text-To-Image Generation via Masked Generative Transformers, Huiwen Chang et al.; Google Research
- LoRA: Low-Rank Adaptation of Large Language Models, Edward J. Hu et al.; Microsoft

Muse generated images before and after fine-tuning with examples filtered by plausibility scores.

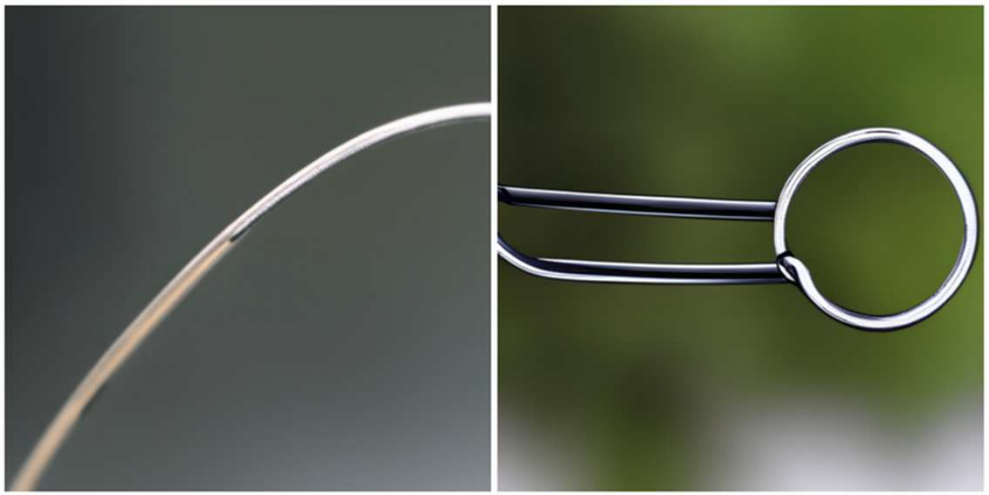
Rich Human Feedback for Text-to-Image Generation

https://research.google/blog/rich-human-feedback-for-text-to-image-generation/

➢ Experiments : Learning from rich human feedback

- Moreover, below we show an example of **using the RAHF aesthetic score as Classifier Guidance** to the **Latent Diffusion model**, demonstrating that each of the fine-grained scores can improve different aspects of the generative model/results.

- Classifier Guidance : Universal Guidance for Diffusion Models, https://arxiv.org/pdf/2302.07121
- Latent Diffusion Model : High-Resolution Image Synthesis with Latent Diffusion Models, https://arxiv.org/abs/2112.10752



(a) w/o guidance (b) with guidance (aesthetics score)

An example of using the **RAHF aesthetic score** as **Classifier Guidance to the Latent Diffusion model**. Prompt: "a macro lens closeup of a paperclip".

(c) w/o guidance (d) with guidance (overall score)

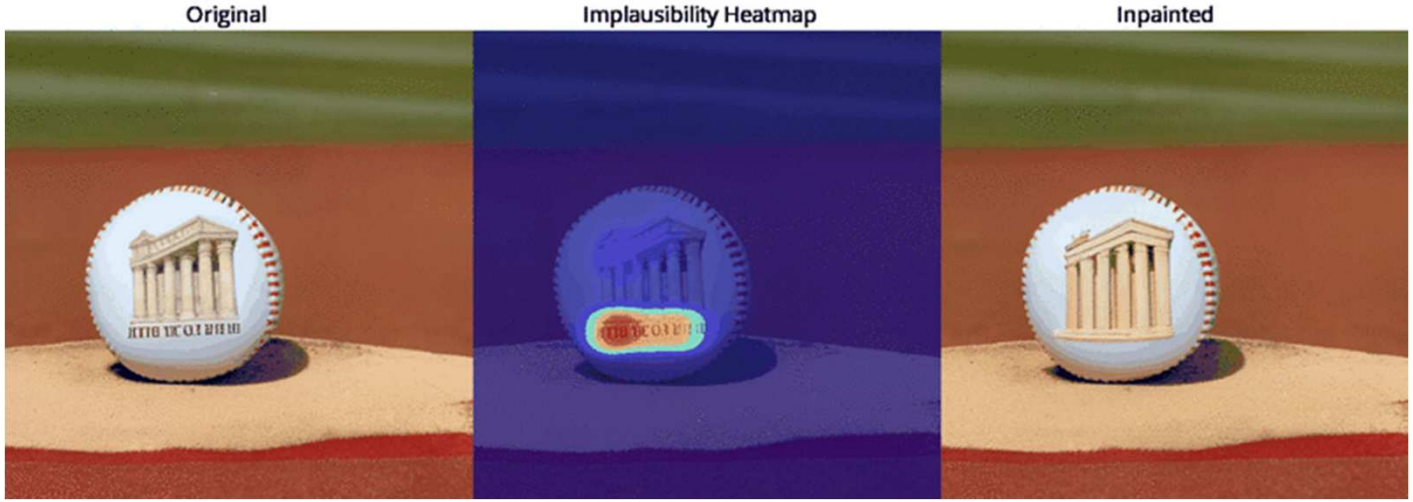
An example of using the **RAHF overall score** as **Classifier Guidance to the Latent Diffusion model**. Prompt: "kitten sushi stained glass window sunset fog"

Rich Human Feedback for Text-to-Image Generation

➢ Experiments : Learning from rich human feedback

https://research.google/blog/rich-human-feedback-for-text-to-image-generation/

- We also demonstrate that our model's predicted heatmaps and scores can be used to perform **region inpainting to improve the quality of generated images**.
- For each image, we first predict implausibility heatmaps, then create a mask by processing the heatmap (using **thresholding** and **dilating**). **Muse inpainting** is applied within the masked region to generate new images that match the text prompt.
- Multiple images are generated, and the final image is chosen by the highest predicted plausibility score by our RAHF.
- Below we show more plausible images with fewer artifacts generated after inpainting.



Region inpainting with Muse generative model. From left to right, the three figures are: original images with artifacts from Muse, predicted implausibility heatmaps from our model, and new images from Muse region inpainting with the mask, respectively.