北京大学 PEKING UNIVERSITY

VLIC: Vision-Language Models As Perceptual Judges for Human-Aligned Image Compression

Kyle Sargent[1], Ruiqi Gao[3], Philipp Henzler[2], Charles Herrmann[3],
Aleksander Holynski[3], Li Fei-Fei[1], Jiajun Wu[1], Jason Y. Zhang[2]

[1]Stanford University [2]Google Research [3]Google DeepMind

Presenter: Yu Cao

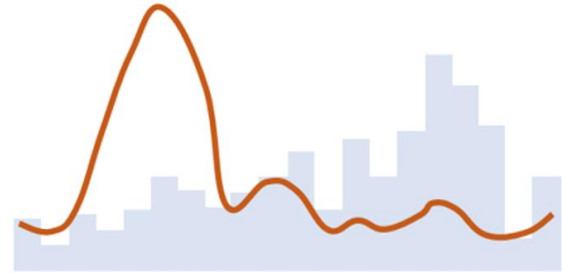
2026.1.11

Background: Image Compression

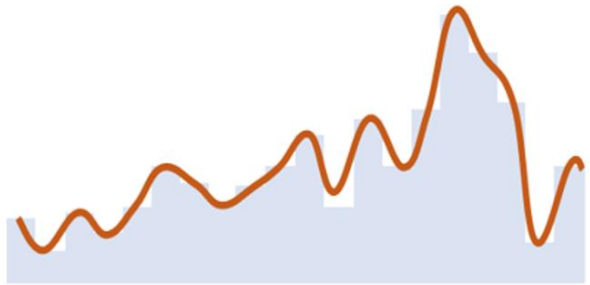
- **The Core of Image Compression**
  - Removal of Redundant Information
- **Measuring Image Redundancy Under Information Theory**
  - Large distribution deviation leads to high bit rates
  - Mutual Information:

$$I(X;Y) = H(X) + H(Y) - H(X|Y)$$

  - The goal of image compression can be transformed into:
    - Performing more accurate distribution estimation between predicted image and real image
    - Reducing the cross-entropy between two distributions



- Large deviation
- High bit rates

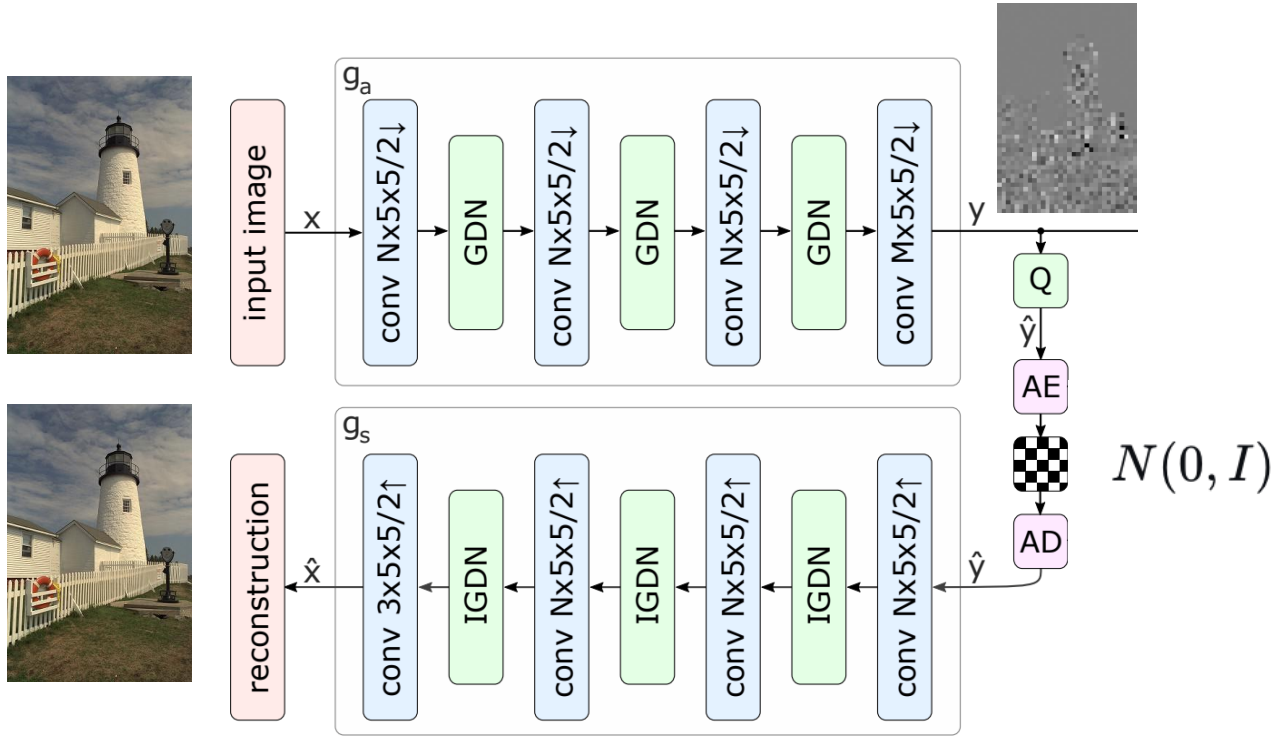


- Small deviation
- Low bit rates

Background: Learned Image Compression

- **Goal: Rate-Distortion Optimization**

$$R + \lambda \cdot D = \underbrace{\mathbb{E}_{\boldsymbol{x} \sim p_{\boldsymbol{x}}} \left[-\log_2 p_{\hat{\boldsymbol{y}}}(\hat{\boldsymbol{y}}) \right]}_{\text{rate (latents)}} + \lambda \cdot \underbrace{\mathbb{E}_{\boldsymbol{x} \sim p_{\boldsymbol{x}}} \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2^2}_{\text{distortion}}$$

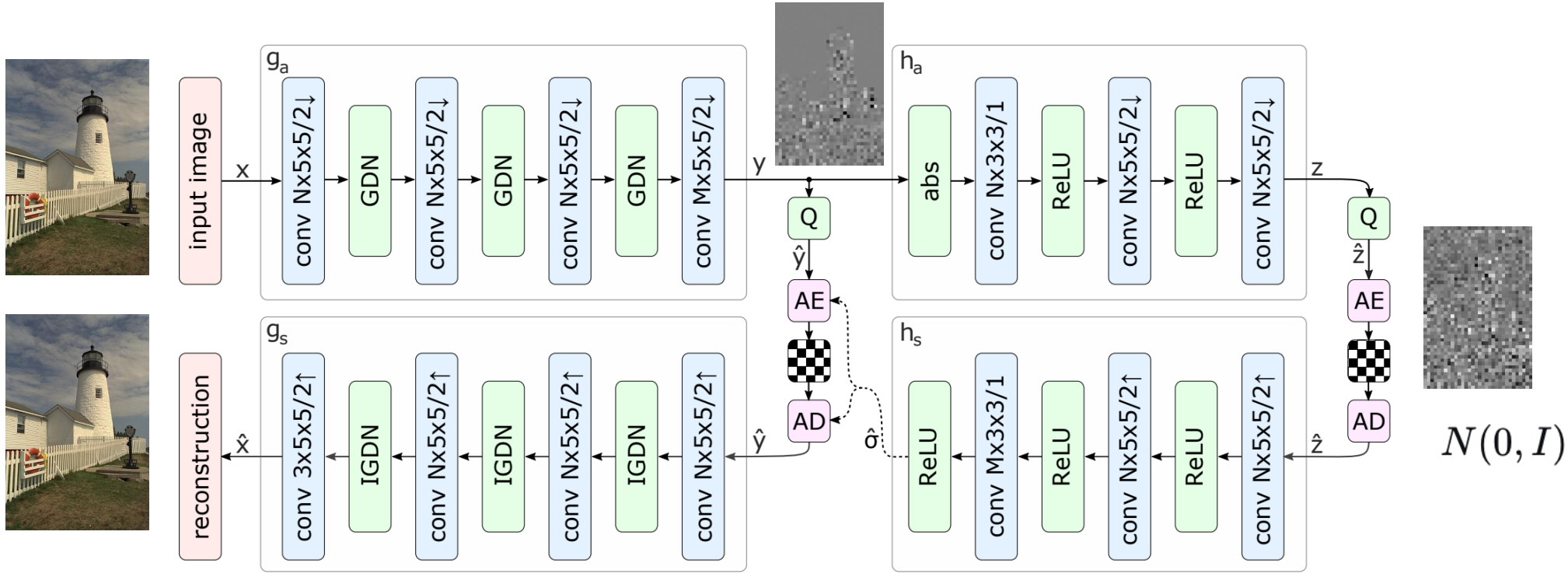


Ballé et al. Variational Image Compression with a Scale Hyperprior, ICLR 2018.

Background: Learned Image Compression

- **Goal: Rate-Distortion Optimization**

$$R + \lambda \cdot D = \underbrace{\mathbb{E}_{\boldsymbol{x} \sim p_{\boldsymbol{x}}}\left[-\log_2 p_{\hat{\boldsymbol{y}}}(\hat{\boldsymbol{y}})\right]}_{\text{rate (latents)}} + \underbrace{\mathbb{E}_{\boldsymbol{x} \sim p_{\boldsymbol{x}}}\left[-\log_2 p_{\hat{\boldsymbol{z}}}(\hat{\boldsymbol{z}})\right]}_{\text{rate (hyper-latents)}} + \lambda \cdot \underbrace{\mathbb{E}_{\boldsymbol{x} \sim p_{\boldsymbol{x}}} \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2^2}_{\text{distortion}}$$

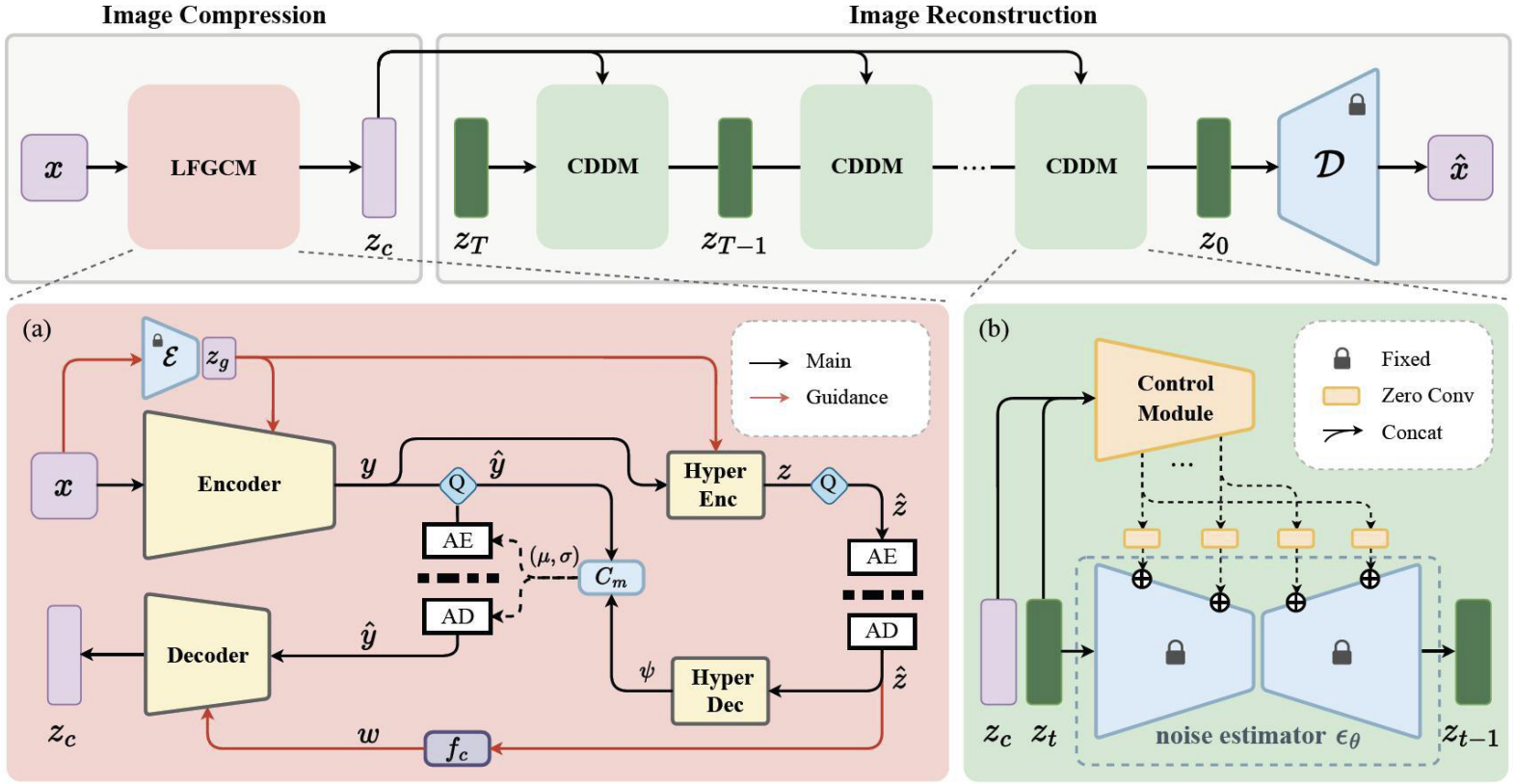


Ballé et al. Variational Image Compression with a Scale Hyperprior, ICLR 2018.

Background: Diffusion-based Image Compression

For Post Processing | **As Entropy Coders** | **As Autoencoders**

- **DiffEIC** - Toward Extreme Image Compression with Latent Feature Guidance and Diffusion Prior (TCSVT 2024)



- Reconstruct VAE latents
- Inject Control to Diffusion

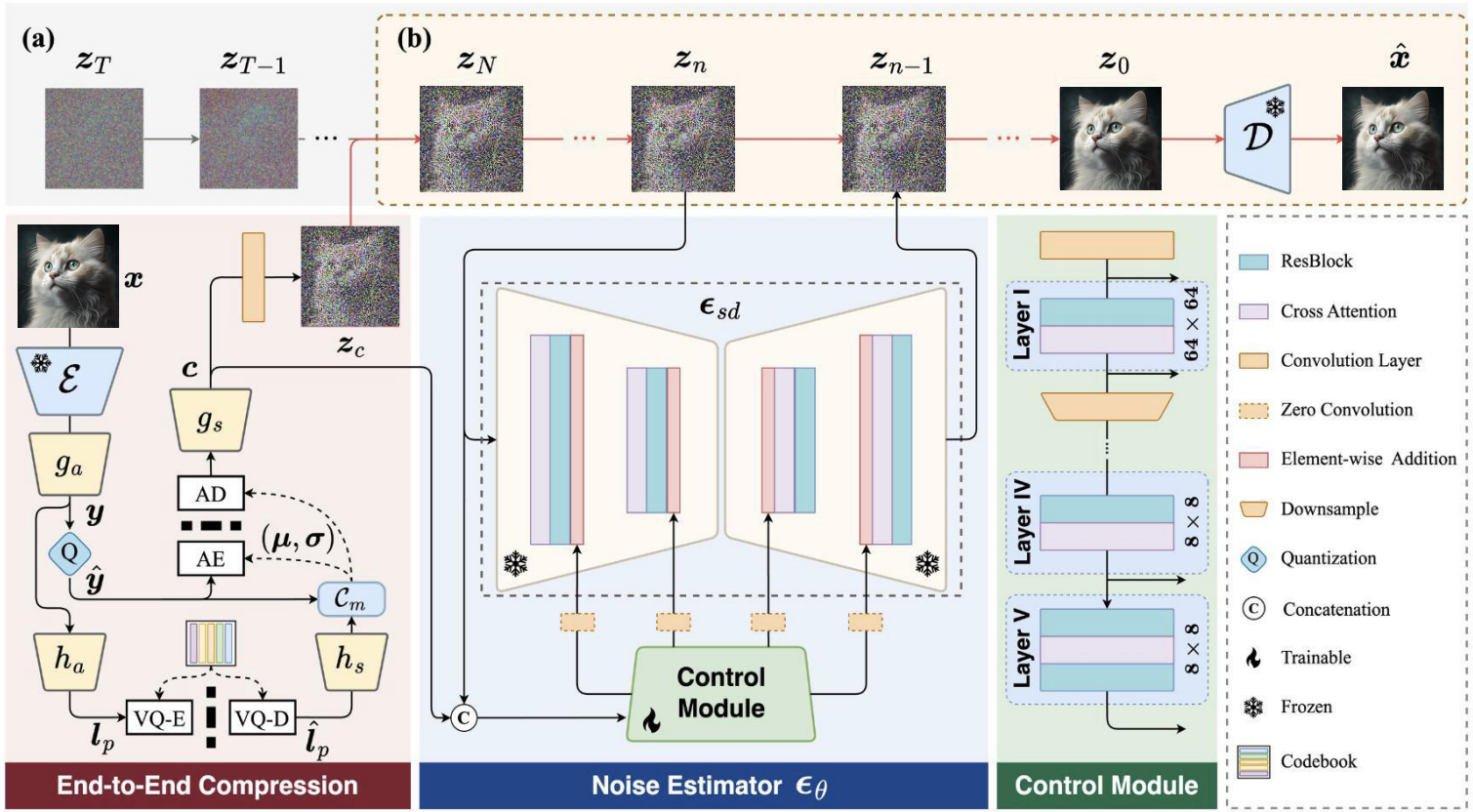
$$L_{\text{ne}} = \mathbb{E}_{z_0,c,t,\epsilon,z_c}\|\epsilon - \epsilon_\theta(z_t, c, t, z_c)\|^2,$$
$$L_{\text{rate}} = R(\hat{y}) + R(\hat{z}),$$
$$L_{\text{sa}} = \|z_c - E(x)\|^2,$$
$$L_{\text{total}} = \lambda L_{\text{rate}} + \lambda_{\text{sa}} L_{\text{sa}} + \lambda_{\text{ne}} L_{\text{ne}}$$

- **Problem:**
  - Generation start from noise
  - Content drift

Background: Diffusion-based Image Compression

For Post Processing | **As Entropy Coders** | **As Autoencoders**

- **RDEIC**: Accelerating Diffusion-Based Extreme Image Compression with Relay Residual Diffusion (TCSVT 2025)

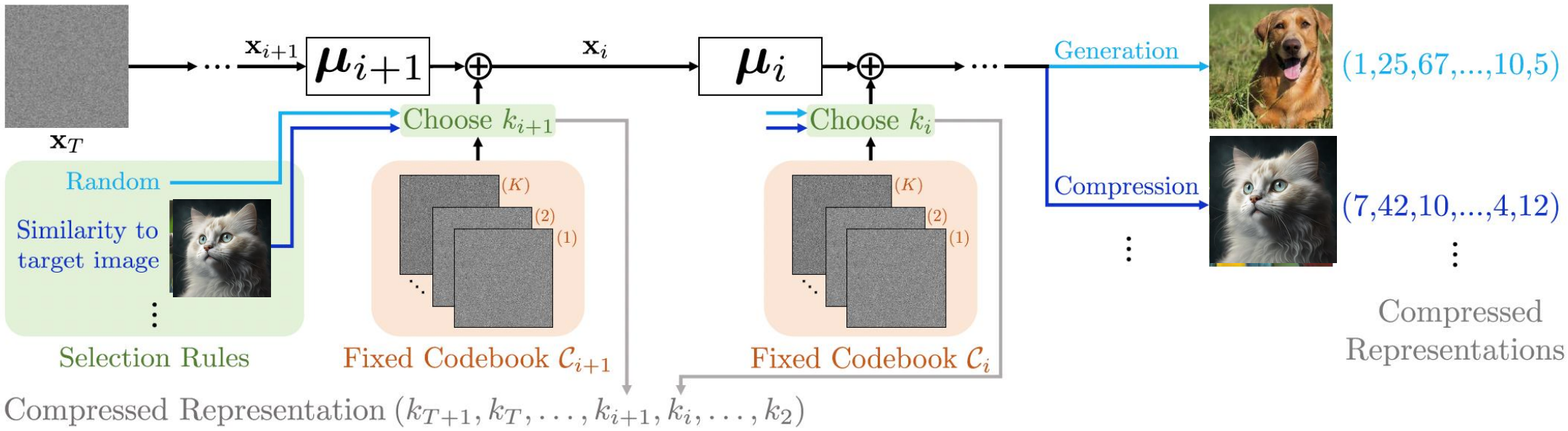


- Use VQ to quantize hyper **z**
- Use output of compression as the starting point of denoising
- First train on fixed timestep, then finetune on whole process
- **Problem of diffusion-based post-processing:**
  - Diffusion decoupled from R-D optimization
  - May introduce hallucinated details

Background: Diffusion-based Image Compression

For Post Processing | **As Entropy Coders** | **As Autoencoders**

- **DDCM** - Compressed Image Generation with Denoising Diffusion Codebook Models (ICML 2025)

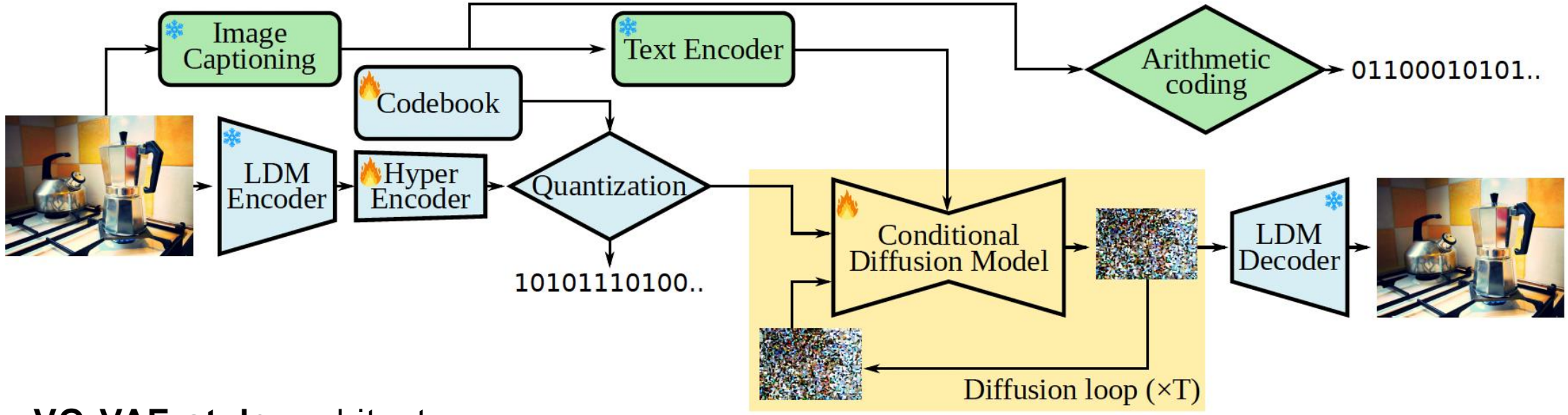


- Use discrete noise codebook to replace continuous sampling of Gaussian noise.
- During compression, select the optimal noise index step-by-step.
- For higher bit rates, use matching pursuit to linearly combine M codebook entries.

Background: Diffusion-based Image Compression

For Post Processing | **As Entropy Coders** | **As Autoencoders**

- **PerCo** - Towards Image Compression with Perfect Realism at Ultra-low Bitrates (ICLR 2024)

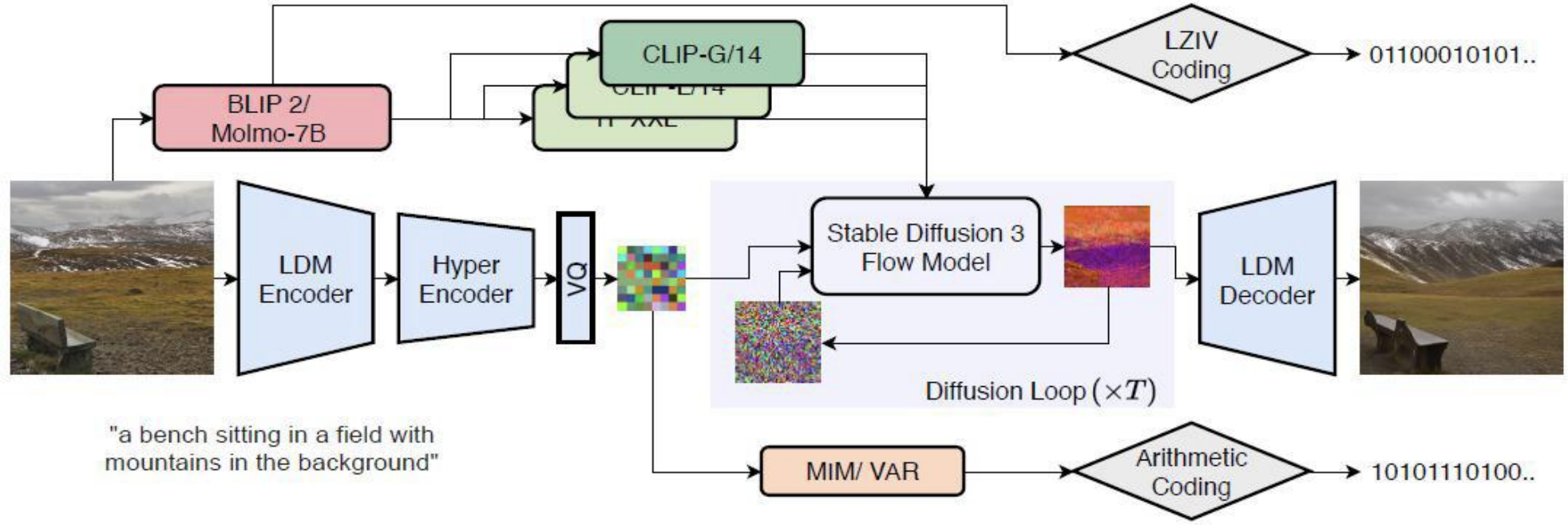


- **VQ-VAE-style** architecture
- **Diffusion model** serving as the decoder, providing a **strong prior.**
- **Compressed text caption** providing global semantic information.
- Concat quantized result and diffusion output to U-Net.

Background: Diffusion-based Image Compression

For Post Processing | **As Entropy Coders** | **As Autoencoders**

- **PerCoV2**: Improved Ultra-Low Bit-Rate Perceptual Image Compression with Implicit Hierarchical Masked Image Modeling (arxiv 2025)

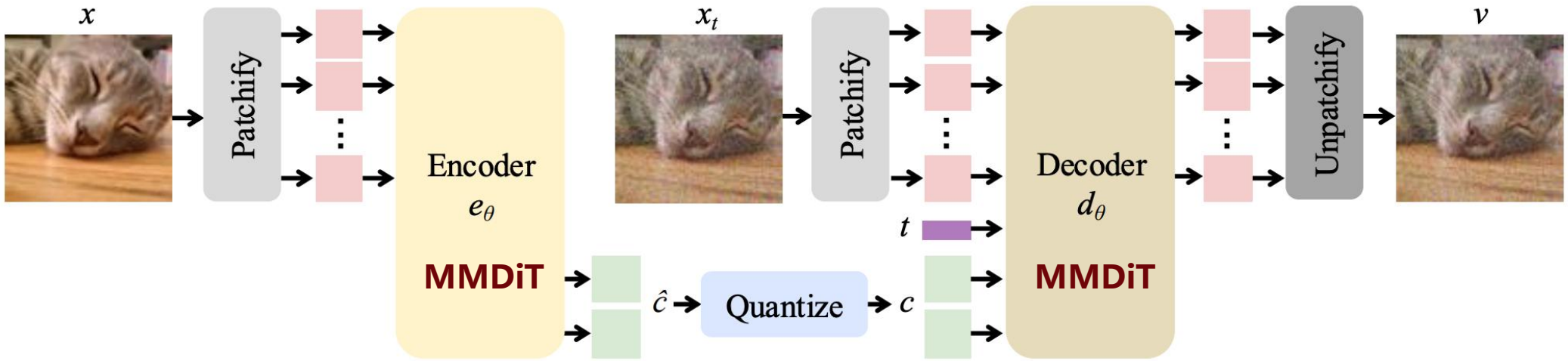


- Use Stable Diffusion 3 Flow Model for better image prior
- MIM (Masked Image Model) or VAR (Visual Autoregressive Model) to model hyper distribution

Background: Diffusion-based Image Compression

For Post Processing | **As Entropy Coders** | **As Autoencoders**

- **Flow to the Mode**: Mode-Seeking Diffusion Autoencoders for State-of-the-Art Image Tokenization (ICCV 2025) (Baseline of this work)



- Use **MMDiT** (adapted from FLUX) as decoder backbone
- $\hat{c}$ is quantized using **lookup-free quantization** $c = q(\hat{c}) = 2 \cdot \mathbb{1}[\hat{c} \geq 0] - 1.$
- Decoder is trained to model a velocity field $v = d_\theta(x_t, c, t)$

Background: Evaluation Metrics

Classical Methods | **NN Approximation** | **VLM for Evaluation**

- **PSNR (Peak Signal-to-Noise Ratio)**
  - based on MSE derivation
  - strictly measures pixel-level error

$$\mathrm{PSNR} = 10 \cdot \log_{10}\left(\frac{\mathrm{MAX}_I{}^2}{\mathrm{MSE}}\right)$$

$$\mathrm{MSE} = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2$$

- **SSIM (Structural Similarity Index Measure)**
  - Based on similarity of luminance, contrast, structure
  - Aligns more closely with human perception

$$\mathrm{SSIM}(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

- **Problem:** Mechanical, often contradict human judgement of visual quality

Background: Evaluation Metrics

Classical Methods | **NN Approximation** | **VLM for Evaluation**

- **LPIPS**
  - Use pretrained network to extract feature
  - Measures image similarity in deep feature space through L2 norm

$$d(x, x_0) = \sum_l \frac{1}{H_l W_l} \sum_{h,w} ||w_l \odot (\hat{y}^l_{hw} - \hat{y}^l_{0hw})||_2^2$$

- **DISTS**
  - Use pretrained network
  - Compute structure and texture similarity in feature space

$$l(\tilde{x}_j^{(i)}, \tilde{y}_j^{(i)}) = \frac{2\mu_{\tilde{x}_j}^{(i)} \mu_{\tilde{y}_j}^{(i)} + c_1}{\left(\mu_{\tilde{x}_j}^{(i)}\right)^2 + \left(\mu_{\tilde{y}_j}^{(i)}\right)^2 + c_1} \quad s(\tilde{x}_j^{(i)}, \tilde{y}_j^{(i)}) = \frac{2\sigma_{\tilde{x}_j \tilde{y}_j}^{(i)} + c_2}{\left(\sigma_{\tilde{x}_j}^{(i)}\right)^2 + \left(\sigma_{\tilde{y}_j}^{(i)}\right)^2 + c_2}$$

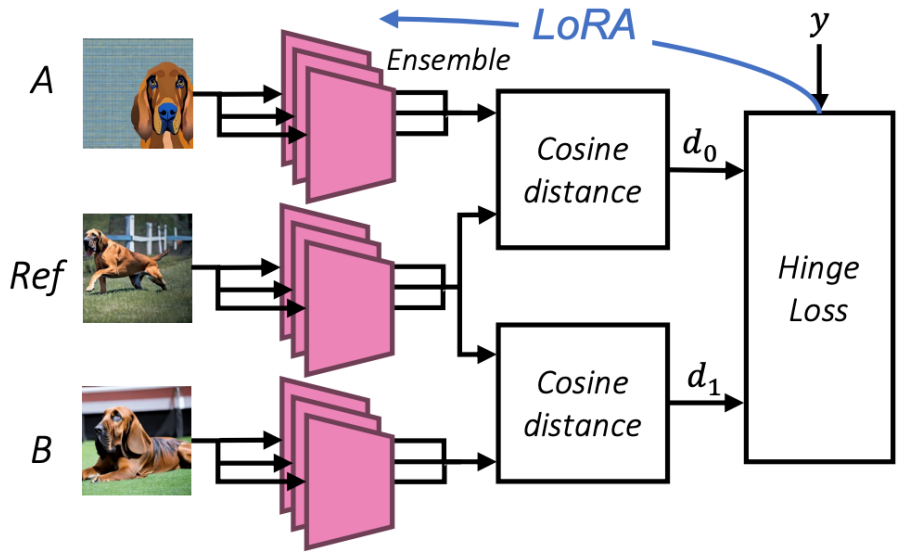
$$D(x, y; \alpha, \beta) = 1 - \sum_{i=0}^{m} \sum_{j=1}^{n_i} \left(\alpha_{ij} l(\tilde{x}_j^{(i)}, \tilde{y}_j^{(i)}) + \beta_{ij} s(\tilde{x}_j^{(i)}, \tilde{y}_j^{(i)}) \right)$$

Fu et al. DreamSim: Learning New Dimensions of Human Visual Similarity using Synthetic Data, NeurIPS 2023.

Background: Evaluation Metrics

Classical Methods | **NN Approximation** | **VLM for Evaluation**

- **DreamSim**
  - Add LoRA to ensemble (CLIP/DINO/OpenCLIP)
  - Use Cosine Distance to measure distance



- **Problem of NN Approximation**
  - Directly optimizing on those metrics can exploit their null-space
  - May not align with human preference

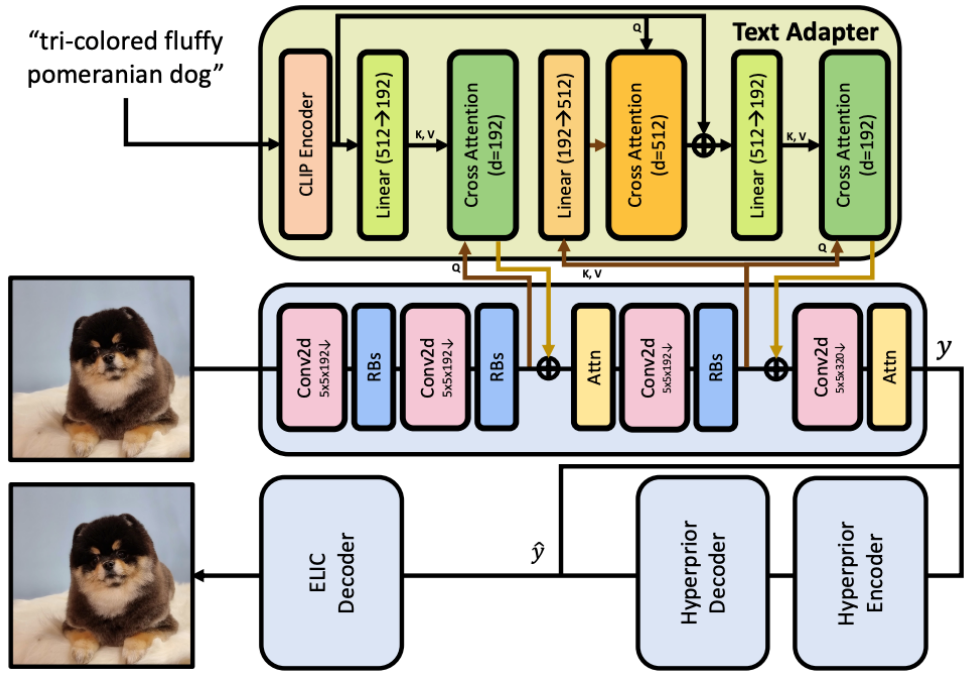
Background: Evaluation Metrics

Classical Methods | **NN Approximation** | **VLM for Evaluation**

- **Text-aligned Image Compression**
  - **E.g. TACO**
    - Utilize CLIP constructive loss for text alignment:

$$L_{\mathrm{j}}(x, \hat{x}, c) = \underbrace{L_{\mathrm{con}}(f_{\mathrm{I}}(\hat{x}), f_{\mathrm{T}}(c))}_{\text{relevance between text and image}} + \beta \cdot \underbrace{\|f_{\mathrm{I}}(x) - f_{\mathrm{I}}(\hat{x})\|_2}_{\text{CLIP embedding distance of images}}.$$

$$\mathcal{L}_{\text{clip}} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(S_{ii})}{\sum_{j=1}^{N}\exp(S_{ij})} \qquad S_{ij} = \tau \cdot \frac{f_T(c_i)^\top f_I(\hat{x}_j)}{\|f_T(c_i)\|\,\|f_I(\hat{x}_j)\|}$$

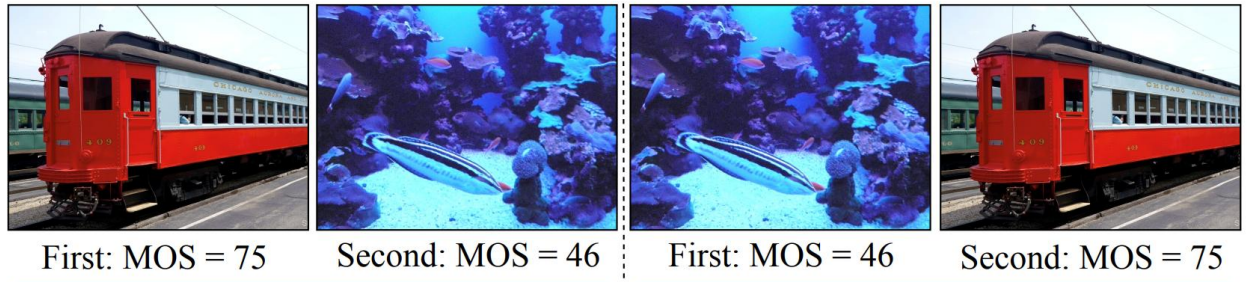


Lee et al. Neural Image Compression with Text-guided Encoding for both Pixel-level and Perceptual Fidelity, ICML 2024.

Background: Evaluation Metrics

Classical Methods | **NN Approximation** | **VLM for Evaluation**

- **2AFC Prompting** of Large Multimodal Models for Image Quality Assessment (TCSVT 2024)
  - Employ the two-alternative forced choice (2AFC) prompting
  - Measure the consistency, accuracy and correlation of evaluation result



Background: Diffusion DPO

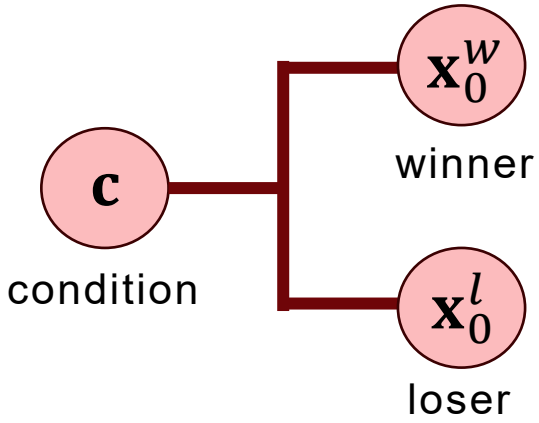
- **Diffusion Model Alignment Using Direct Preference Optimization (CVPR 2024)**
  - Reinforcement Learning from Human Feedback (**RLHF**)
    - Assume $\mathbf{x_0}$: generation result, $\mathbf{c}$: condition (text, image, etc.)
    - RLHF aims to optimize $p_\theta(\mathbf{x}_0|\mathbf{c})$, such that reward function $r(\mathbf{c}, \mathbf{x}_0)$ is maximized under the KL-Divergence constraint:

$$\max_{p_\theta} \underbrace{E_{\mathbf{c},\mathbf{x_0}}\left[r(\mathbf{c}, \mathbf{x_0})\right]}_{\text{Reward Item}} - \beta \underbrace{D_{\mathrm{KL}}\left[p_\theta(\mathbf{x_0}|\mathbf{c})||p_{\mathrm{ref}}(\mathbf{x_0}|\mathbf{c})\right]}_{\text{KL-Divergence Item}}$$

    - p_{ref} : reference distribution, β : hyperparameter

Background: Diffusion DPO

- **Diffusion Model Alignment Using Direct Preference Optimization (CVPR 2024)**
  - Reward Modeling: **Bradley-Terry (BT) model**
    - Assume access only to ranked pairs generated from same condition $\mathbf{c}$



Data: $(\mathbf{c}, \mathbf{x}_0^w, \mathbf{x}_0^l)$

$$p_{\mathrm{BT}}(\mathbf{x}_0^w \succ \mathbf{x}_0^l | \mathbf{c}) = \sigma\left(r(\mathbf{c}, \mathbf{x}_0^w) - r(\mathbf{c}, \mathbf{x}_0^l)\right)$$

σ : sigmoid function

r can be parameterized by Neural Network r_ϕ

- Maximum Likelihood Estimation

$$L_{\mathrm{BT}}(\phi) = -E_{\mathbf{c}, \mathbf{x}_0^w, \mathbf{x}_0^l}\left[\log \sigma\left(r_\phi(\mathbf{c}, \mathbf{x}_0^w) - r_\phi(\mathbf{c}, \mathbf{x}_0^l)\right)\right]$$

Background: Diffusion DPO

RLHF: $\max_{p_\theta} \underbrace{E_{\mathbf{c},\mathbf{x_0}}\left[r(\mathbf{c},\mathbf{x_0})\right]}_{\text{Reward Item}} - \underbrace{\beta D_{\mathrm{KL}}\left[p_\theta(\mathbf{x_0}|\mathbf{c})||p_{\mathrm{ref}}(\mathbf{x_0}|\mathbf{c})\right]}_{\text{KL-Divergence Item}}$

- **Diffusion Model Alignment Using Direct Preference Optimization (CVPR 2024)**
  - **DPO Objective Function**
    - Unique Global Optimal Solution:

$$p_\theta^*(\mathbf{x_0}|\mathbf{c}) = p_{\mathrm{ref}}(\mathbf{x_0}|\mathbf{c}) \exp(r(\mathbf{c},\mathbf{x_0})/\beta) / Z(\mathbf{c})$$

$$Z(\mathbf{c}) = \sum_{\mathbf{x_0}} p_{\mathrm{ref}}(\mathbf{x_0}|\mathbf{c}) \exp(r(\mathbf{c},\mathbf{x_0})/\beta)$$

    - Therefore, we can get the solution of $r(\mathbf{c},\mathbf{x_0})$:

$$r(\mathbf{c},\mathbf{x_0}) = \beta \log \frac{p_\theta^*(\mathbf{x_0}|\mathbf{c})}{p_{\mathrm{ref}}(\mathbf{x_0}|\mathbf{c})} + \beta \log Z(\mathbf{c})$$

    - with MLE of BT model, we derive:

$$L_{\mathrm{DPO}}(\theta) = -E_{\mathbf{c},\mathbf{x}_0^w,\mathbf{x}_0^l}\left[\log \sigma\left(\beta \log \frac{p_\theta(\mathbf{x}_0^w|\mathbf{c})}{p_{\mathrm{ref}}(\mathbf{x}_0^w|\mathbf{c})} - \beta \log \frac{p_\theta(\mathbf{x}_0^l|\mathbf{c})}{p_{\mathrm{ref}}(\mathbf{x}_0^l|\mathbf{c})}\right)\right]$$

Background: Diffusion DPO

$$L_{\text{DPO}}(\theta) = -E_{\mathbf{c},\mathbf{x}_0^w,\mathbf{x}_0^l}\left[\log\sigma\left(\beta\log\frac{p_\theta(\mathbf{x}_0^w|\mathbf{c})}{p_{\text{ref}}(\mathbf{x}_0^w|\mathbf{c})} - \beta\log\frac{p_\theta(\mathbf{x}_0^l|\mathbf{c})}{p_{\text{ref}}(\mathbf{x}_0^l|\mathbf{c})}\right)\right]$$

- **Diffusion Model Alignment Using Direct Preference Optimization (CVPR 2024)**
  - **DPO for Diffusion Models**
    - Reward on the whole diffusion chain $\mathbf{x}_{0:T}^w, \mathbf{x}_{0:T}^l$:

$$L_{\text{DDPO}}(\theta) = -E_{(\mathbf{x}_0^w,\mathbf{x}_0^l)\sim\mathbb{D}}\log\sigma\left(\beta E_{\mathbf{x}_{1:T}^w\sim p_\theta(\cdot|\mathbf{x}_0^w),\ x_{1:T}^l\sim p_\theta(\cdot|\mathbf{x}_0^l)}\left[\log\frac{p_\theta\left(\mathbf{x}_{0:T}^w\right)}{p_{\text{ref}}\left(\mathbf{x}_{0:T}^w\right)} - \log\frac{p_\theta\left(\mathbf{x}_{0:T}^l\right)}{p_{\text{ref}}\left(\mathbf{x}_{0:T}^l\right)}\right]\right)$$

    - **Markov process** $p_\theta(\mathbf{x_{0:T}}) = p(\mathbf{x_T})\prod_{t=1}^T p_\theta(\mathbf{x_{t-1}}\mid\mathbf{x_t})$
    - By Jensen inequality & convexity of log (referring to Appendix A of paper)

$$\begin{aligned}L_{\text{DDPO}}(\theta) \le -\mathbb{E}_{(\mathbf{x}_0^w,\mathbf{x}_0^l)\sim\mathcal{D},t\sim\mathcal{U}(0,T),\mathbf{x}_t^w\sim q(\mathbf{x}_t^w|\mathbf{x}_0^w),\mathbf{x}_t^l\sim q(\mathbf{x}_t^l|\mathbf{x}_0^l)}\log\sigma(-\beta T(&\\ +\mathbb{D}_{\text{KL}}(q(\mathbf{x}_{t-1}^w|\mathbf{x}_{0,t}^w)\|p_\theta(\mathbf{x}_{t-1}^w|\mathbf{x}_t^w)) - \mathbb{D}_{\text{KL}}(q(\mathbf{x}_{t-1}^w|\mathbf{x}_{0,t}^w)\|p_{\text{ref}}(\mathbf{x}_{t-1}^w|\mathbf{x}_t^w)) &\quad\text{Winner}\\ -\mathbb{D}_{\text{KL}}(q(\mathbf{x}_{t-1}^l|\mathbf{x}_{0,t}^l)\|p_\theta(\mathbf{x}_{t-1}^l|\mathbf{x}_t^l)) + \mathbb{D}_{\text{KL}}(q(\mathbf{x}_{t-1}^l|\mathbf{x}_{0,t}^l)\|p_{\text{ref}}(\mathbf{x}_{t-1}^l|\mathbf{x}_t^l))). &\quad\text{Loser}\end{aligned}$$

Background: Diffusion DPO

$$L_{\text{DPO}}(\theta) = -E_{\mathbf{c},\mathbf{x}_0^w,\mathbf{x}_0^l}\left[\log\sigma\left(\beta\log\frac{p_\theta(\mathbf{x}_0^w|\mathbf{c})}{p_{\text{ref}}(\mathbf{x}_0^w|\mathbf{c})} - \beta\log\frac{p_\theta(\mathbf{x}_0^l|\mathbf{c})}{p_{\text{ref}}(\mathbf{x}_0^l|\mathbf{c})}\right)\right]$$

- **Diffusion Model Alignment Using Direct Preference Optimization (CVPR 2024)**
  - **DPO for Diffusion Models**
    - Using derivations in DDPM, we get:

$$\frac{\beta_t^2}{2\sigma_t^2\alpha_t(1-\bar{\alpha}_t)}$$

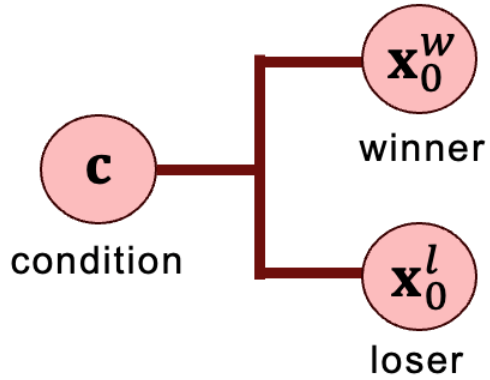
$$L(\theta) = -\mathbb{E}_{(\mathbf{x}_0^w,\mathbf{x}_0^l)\sim\mathcal{D},t\sim\mathcal{U}(0,T),\mathbf{x}_t^w\sim q(\mathbf{x}_t^w|\mathbf{x}_0^w),\mathbf{x}_t^l\sim q(\mathbf{x}_t^l|\mathbf{x}_0^l)}\log\sigma\left(-\beta T\omega(\lambda_t)\right.($$
$$\|\epsilon^w - \epsilon_\theta(\mathbf{x}_t^w,t)\|_2^2 - \|\epsilon^w - \epsilon_{\text{ref}}(\mathbf{x}_t^w,t)\|_2^2 - \left(\|\epsilon^l - \epsilon_\theta(\mathbf{x}_t^l,t)\|_2^2 - \|\epsilon^l - \epsilon_{\text{ref}}(\mathbf{x}_t^l,t)\|_2^2\right)\Big)$$

    - $\omega(\lambda_t)$: Weight of loss function in DDPM (constant)
    - $\epsilon_\theta(\mathbf{x}_t,t)$: The predicted noise
    - KL-Divergence ➡ L2-distance of noise

Background: Diffusion DPO

- **Diffusion Model Alignment Using Direct Preference Optimization (CVPR 2024)**
  - **Summary**
    - Triplet Data: $(\mathbf{c}, \mathbf{x}_0^w, \mathbf{x}_0^l)$ (Bradley-Terry Model)
    - As long as we get those winner and loser,

    we can perform finetuning using the following Loss Function:



$$L(\theta) = -\mathbb{E}_{(\mathbf{x}_0^w,\mathbf{x}_0^l)\sim\mathcal{D},t\sim\mathcal{U}(0,T),\mathbf{x}_t^w\sim q(\mathbf{x}_t^w|\mathbf{x}_0^w),\mathbf{x}_t^l\sim q(\mathbf{x}_t^l|\mathbf{x}_0^l)} \log\sigma\left(-\beta T\omega(\lambda_t)\right($$
$$\|\epsilon^w - \epsilon_\theta(\mathbf{x}_t^w, t)\|_2^2 - \|\epsilon^w - \epsilon_{\text{ref}}(\mathbf{x}_t^w, t)\|_2^2 - \left(\|\epsilon^l - \epsilon_\theta(\mathbf{x}_t^l, t)\|_2^2 - \|\epsilon^l - \epsilon_{\text{ref}}(\mathbf{x}_t^l, t)\|_2^2\right)\Big)$$

Methods

- **Overview**

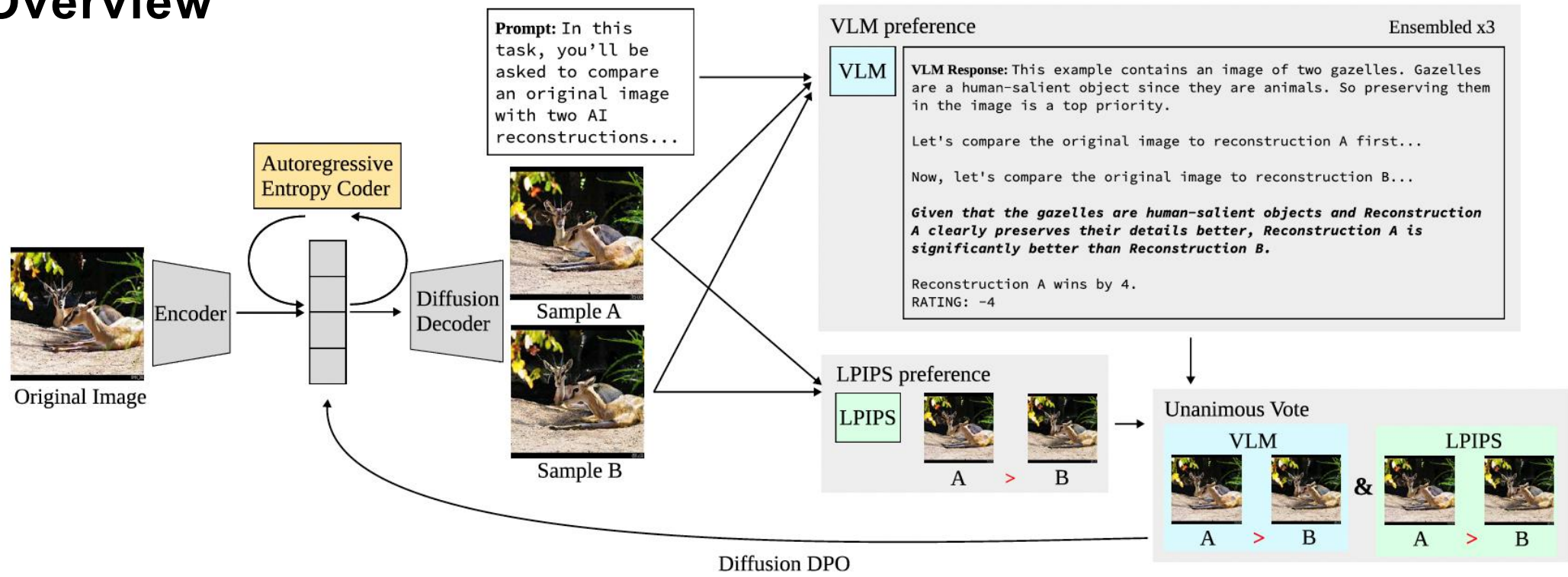
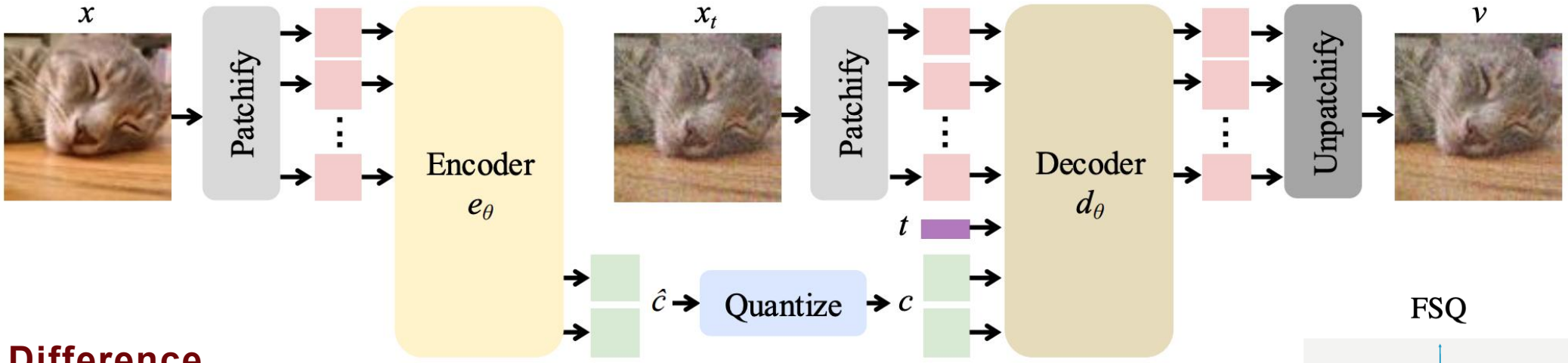


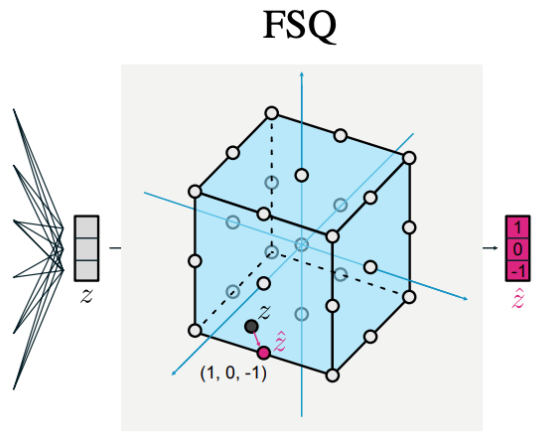
Figure 3. **Method.** An original image is encoded to a one-dimensional discrete latent code via an encoder. The discrete code is entropy coded by an auto-regressive language model. The diffusion decoder samples two reconstructions conditioned on the latent code, which are ranked via a VLM. The resulting preference is used to train the full diffusion autoencoder via Diffusion DPO [47].

Methods

- **Encoder & Decoder Architecture**
  - Architecture: Same as **Flow to the Mode**

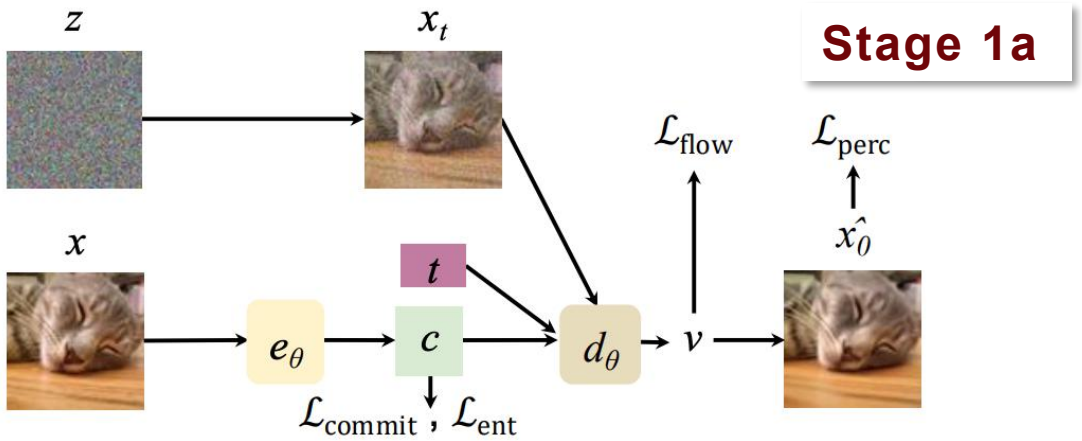


- **Difference**
  - Use **Finite Scalar Quantization** (FSQ) instead of lookup-free quantization: $\mathrm{round}(f(z))$
  - Separately train a hyper network to compress the discrete tokens from FSQ



Methods

- **First Stage Training (**Same as **Flow to the Mode)**



Set fixed timestep

flow matching loss

$$\mathcal{L}_{\text{flow}} = \mathbb{E}\left[\left\|x - z - d_\theta(x_t, q(e_\theta(x)), t)\right\|_2^2\right].$$

perceptual loss

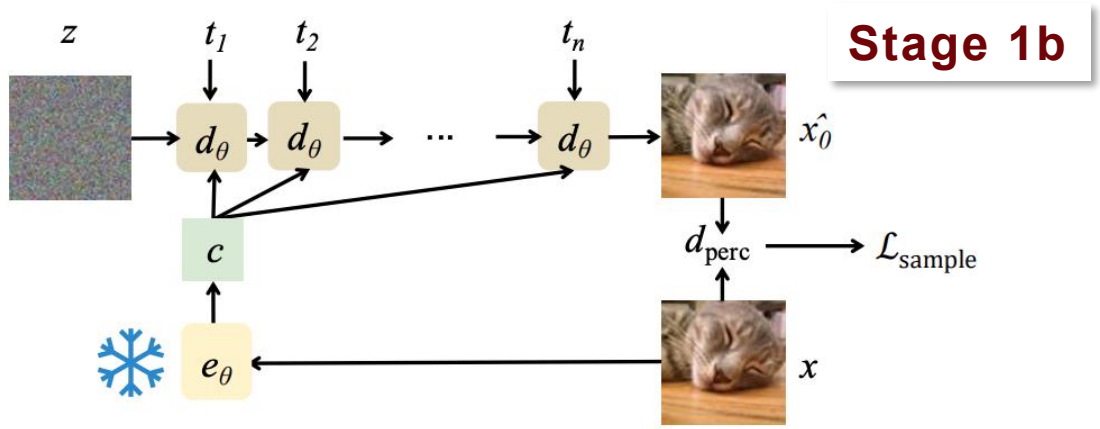
$$\mathcal{L}_{\text{perc}} = \mathbb{E}\left[d_{\text{perc}}(x, x_t + t d_\theta(x_t, q(e_\theta(x)), t))\right].$$

token reconstruction loss

$$\mathcal{L}_{\text{commit}} = \mathbb{E}\left[\left\|\hat{c} - q(\hat{c})\right\|_2^2\right]$$

token entropy loss

$$\mathcal{L}_{\text{ent}} = \mathbb{E}\left[H(q(\hat{c})\right] - H(\mathbb{E}\left[q(\hat{c})\right])$$



Backpropagate through the whole chain

flow matching loss

$$\mathcal{L}_{\text{flow}} = \mathbb{E}\left[\left\|x - z - d_\theta(x_t, q(e_\theta(x)), t)\right\|_2^2\right].$$

perceptual loss

$$\mathcal{L}_{\text{sample}} = \mathbb{E}\left[d_{\text{perc}}\left(x, d_{t_n} \circ d_{t_{n-1}} \circ \cdots \circ d_{t_1}(z)\right)\right]$$

Methods

- **Second Stage Training (With Diffusion DPO)**
  - Utilize the Diffusion DPO we have derived before:

$$L(\theta) = L_{\text{DDPO}}(\theta) + \lambda_{\text{Flow}} L_{\text{Flow}}(\theta)$$

$$L_{\text{DDPO}}(\theta) = -\mathbb{E} \log \sigma \left(-\beta \omega(\lambda_t)(\Delta^w - \Delta^l)\right)$$

Diffusion DPO Item

$$\Delta^w = \|\boldsymbol{\epsilon}^w - \boldsymbol{\epsilon}_\theta(\hat{\mathbf{x}}_t^w, \mathbf{x}, t)\|_2^2 - \|\boldsymbol{\epsilon}^w - \boldsymbol{\epsilon}_{\text{ref}}(\hat{\mathbf{x}}_t^w, \mathbf{x}, t)\|_2^2$$

$$\Delta^l = \left\|\boldsymbol{\epsilon}^l - \boldsymbol{\epsilon}_\theta(\hat{\mathbf{x}}_t^l, \mathbf{x}, t)\right\|_2^2 - \left\|\boldsymbol{\epsilon}^l - \boldsymbol{\epsilon}_{\text{ref}}(\hat{\mathbf{x}}_t^l, \mathbf{x}, t)\right\|_2^2$$

$$L_{\text{Flow}}(\theta) = \mathbb{E}_{\epsilon,x,t}(\|\mathbf{v} - \mathbf{v}_\theta(\mathbf{x}, \mathbf{x}_t, t)\|_2^2)$$

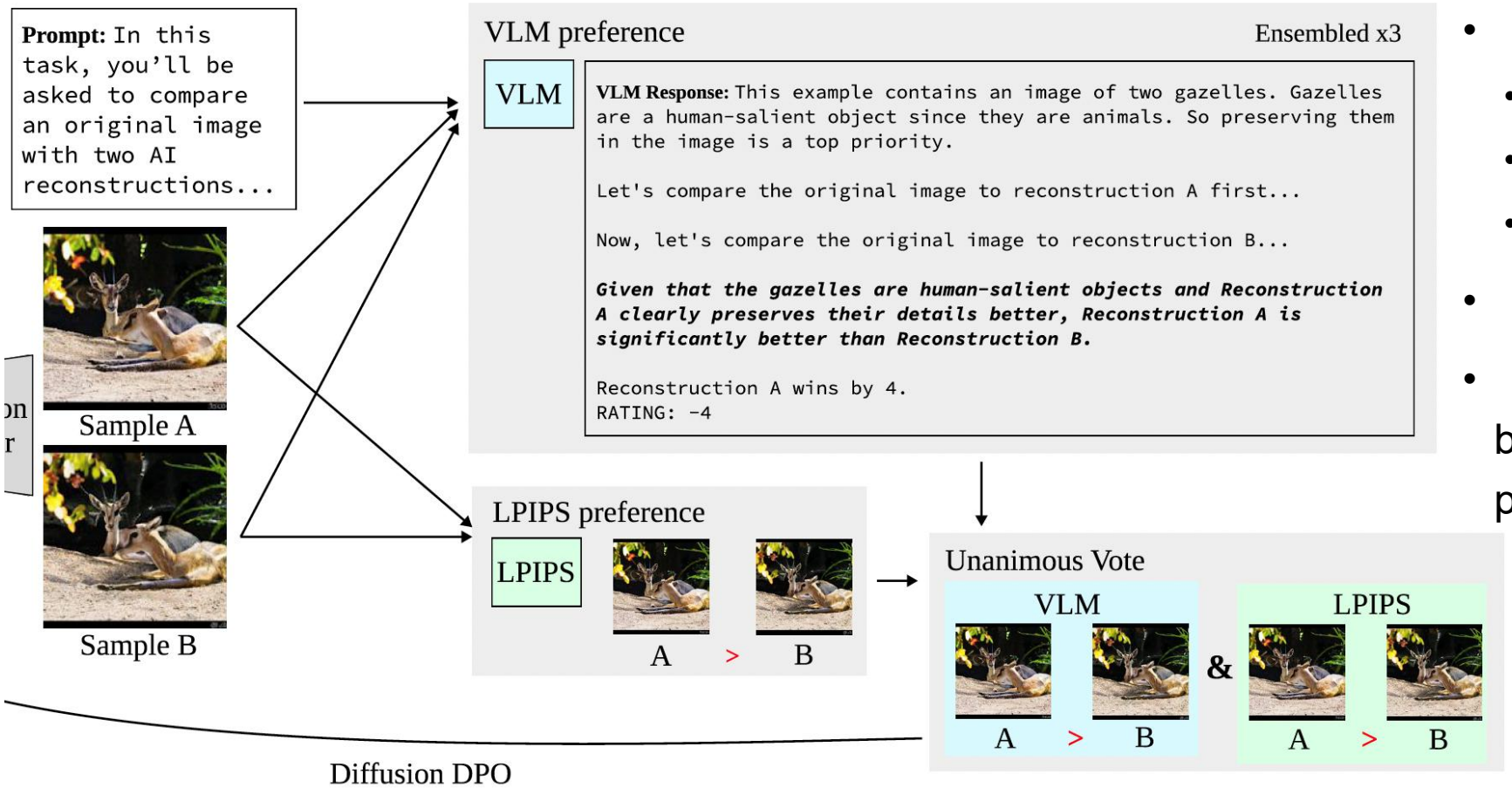
Flow Matching Item

- For training stability
- $\boldsymbol{v} = \boldsymbol{\epsilon} + \boldsymbol{x}$ the flow matching velocity
- v-parameterization is chosen

  - Train with the encoder **unfrozen** for slightly better performance

Methods

- **VLM Reward Computation**
  - Use Gemini 2.5-Flash to judge compressed images



- Pass three images:
  - Original Image $\mathbf{x}$
  - Reconstruction $\hat{\mathbf{x}}_0^A$
  - Reconstruction $\hat{\mathbf{x}}_0^B$
- Pass a detailed instruction
- Ask VLM to give a rate between -5 and 5, and provide detailed reasoning

Methods

- **VLM Reward Computation**
  - Several mitigation strategies to improve the reliability
    - **Rate in two-orders**

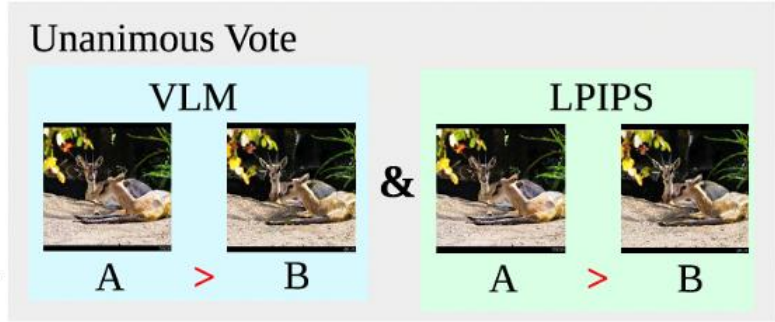
$$r^i_{B,0} = -r^i_{A,0} = \mathrm{VLM}(\boldsymbol{x}, \hat{\boldsymbol{x}}^{\boldsymbol{A}}_{\boldsymbol{0}}, \hat{\boldsymbol{x}}^{\boldsymbol{B}}_{\boldsymbol{0}}, i) \qquad r^i_{A,1} = -r^i_{B,1} = \mathrm{VLM}(\boldsymbol{x}, \hat{\boldsymbol{x}}^{\boldsymbol{B}}_{\boldsymbol{0}}, \hat{\boldsymbol{x}}^{\boldsymbol{A}}_{\boldsymbol{0}}, i)$$

$$r^i_A = \mathrm{sign}(r^i_{A,0} + r^i_{A,1}), \quad r^i_B = \mathrm{sign}(r^i_{B,0} + r^i_{B,1}).$$

    - **Average over n random seeds**

$$r_A = \sum_{i=1}^{n} r^i_A, \quad r_B = \sum_{i=1}^{n} r^i_B$$

Ensembled x3



    - **Ensemble VLM reward with LPIPS**
      - Require LPIPS and the VLM to produce a unanimous judgment
      - If they disagree, the sample is discarded
  - Use the reward to determine winner and loser

Experiments

- **Prompting Details**
  - **(Detailed Task Description)**

In this task, you will be asked to compare an original image with two AI reconstructions of that image, Reconstruction A followed by Reconstruction B. You will see triplets of images like (original, A, B).

  - **(Score Interpretation)**

You will give a relative rating of the two images. This is **between -5 and 5**, inclusive.

Higher scores mean that Reconstruction B is relatively better.

  - **(Give a ranking example)**

So if you give a score of -2, you think A is kind of better, and if you give a score of +5, you think B is obviously significantly better. **Be sparing with the higher magnitude scores** - you should expect that most triplets you see won't have an obviously better reconstruction.

Experiments

- **Prompting Details**

(continue)

- **(Rating Criteria and Priority)**

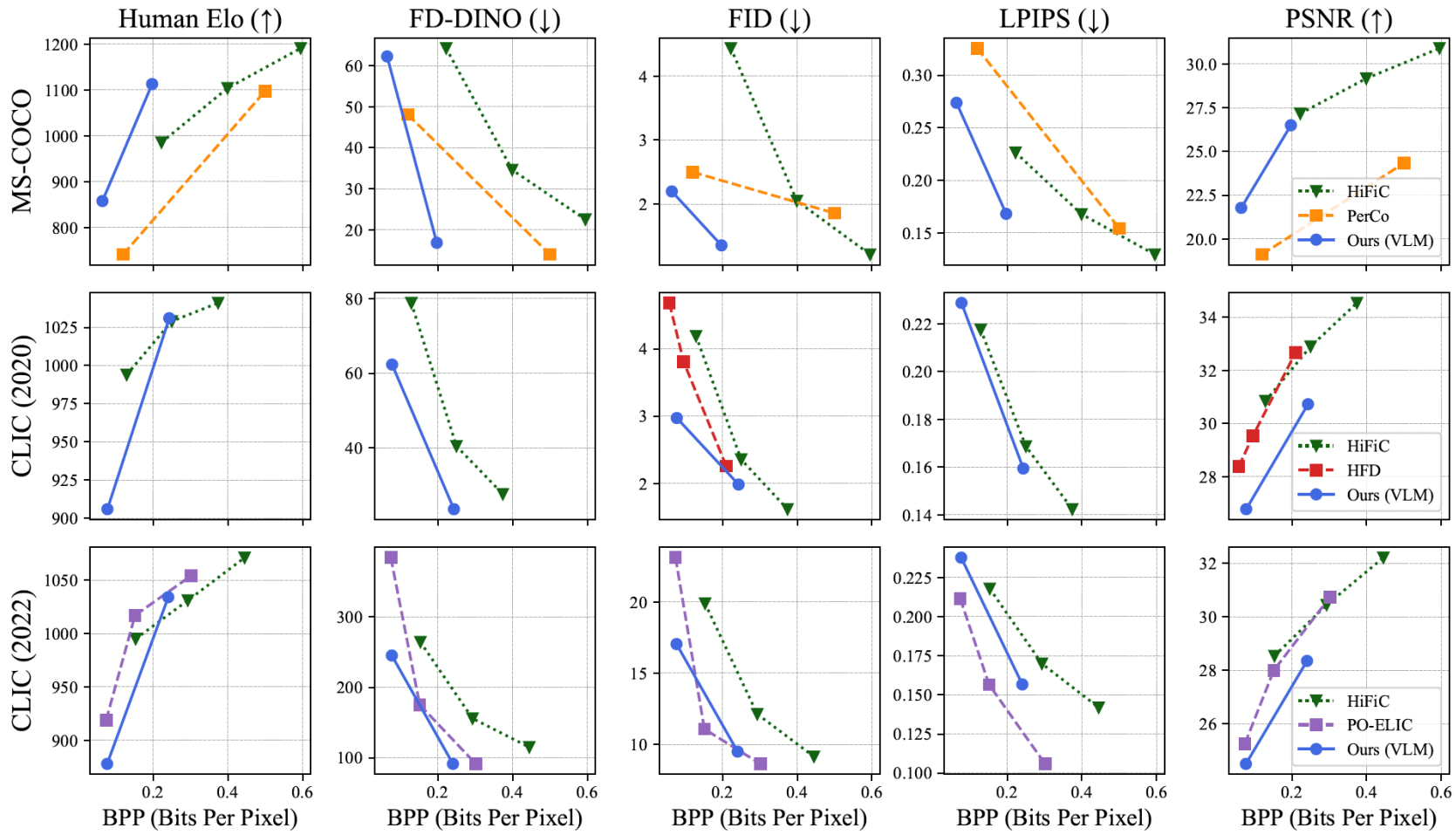
In your ratings, make sure to **prioritize differences that are semantically important to a human observer**. If a distortion changes the meaning of an image to a human observer, then it's more significant than if a distortion changes the texture of an image. Your response should **conclude with "RATING: X"**, where X is your rating, i.e. -2 or 3.

- **(Specific Requirements)**

Now here is the image triplet that you need to rate. Make sure to **provide a lot more reasoning** and make sure to carefully look at each of the three images and provide meticulous visual justifications based on evidence from each image! **Remember, negative scores mean that A is better, and positive scores mean that B is better.** **(Emphasize again)**

Experiments

- **Quantitative Results**



- **FD-DINO**: Fréchet Distance with DINO Backbone
- **Human Elo**: computed via large-scale user studies
- Compare to HiFiC, it performs better on perceptual metrics

Experiments

- **Qualitative Results**
  - More accurate reconstruction of text, faces, architectural details, etc.

Original

Ours: 0.196 bpp

HiFiC: 0.199 bpp

PerCo: 0.5 bpp

Experiments

- **Qualitative Results**
  - More accurate reconstruction of text, faces, architectural details, etc.

Original

Ours: 0.198 bpp

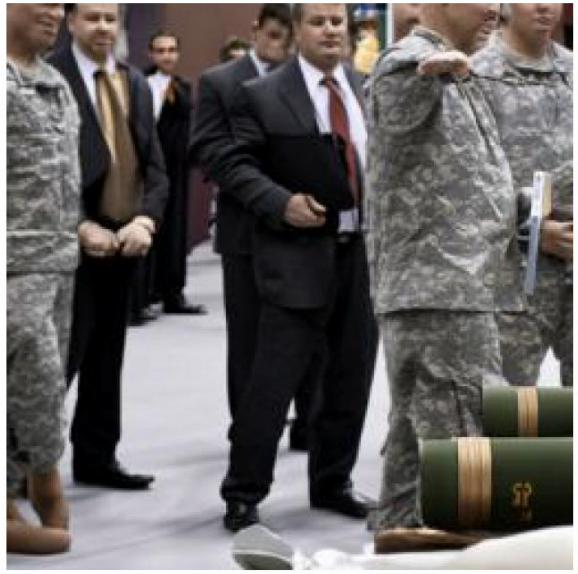
HiFiC: 0.287 bpp

PerCo: 0.12 bpp

Experiments

- **Qualitative Results**
  - More accurate reconstruction of text, faces, architectural details, etc.

Original



Ours: 0.203 bpp

HiFiC: 0.281 bpp

PerCo: 0.12 bpp

Experiments

- **Qualitative Results**

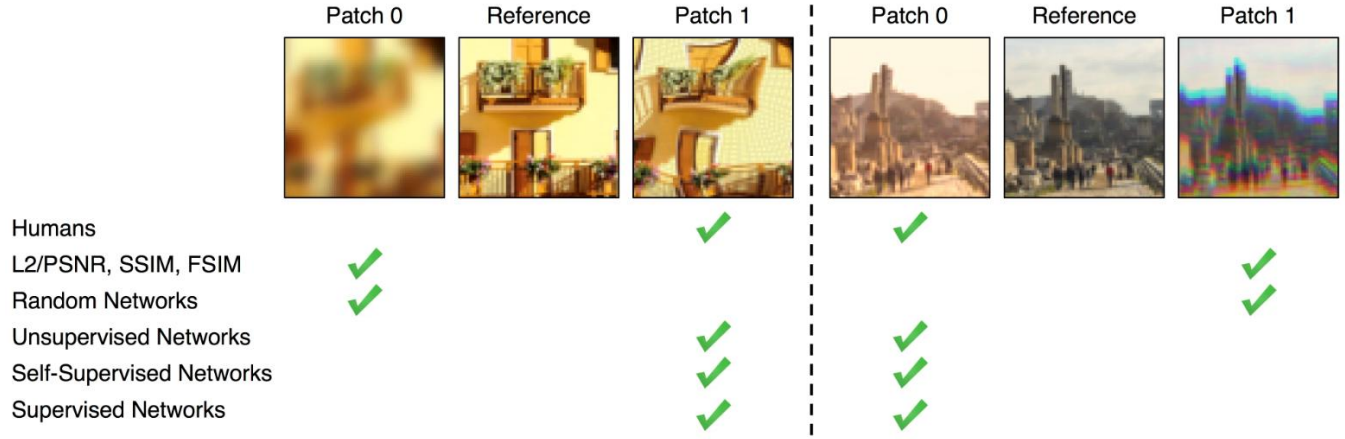
More accurate reconstruction of text, faces, architectural details, etc.

Experiments: Verification

- **Replicating human judgments**
  - Use BAPPS (Berkeley-Adobe Perceptual Patch Similarity (BAPPS) dataset and own collected Compressed Images dataset
  - Gemini 2.5-Flash can replicate human perceptual judgments

Accuracy	BAPPS-Val	Compressed Images
Human	73.99	72.15
LPIPS	69.56	92.32
DreamSim	68.13†	-
VLM	69.44	83.80

Human 2AFC Benchmarks



Examples of BAPPS Dataset

Experiments: Ablation Study

- **Importance of the VLM**
  - Compared with post-training only with LPIPS
  - Post-training provides gain

MS-COCO	Human Elo ↑	FD-DINO ↓	FID ↓	LPIPS ↓	PSNR ↑
Ours (0.07bpp)	**858**	**62.25**	**2.20**	**0.274**	**21.78**
– LPIPS post-training only	838	63.63	2.33	**0.274**	21.77
Ours (0.21bpp)	**1112**	**16.83**	1.35	**0.168**	26.50
– LPIPS post-training only	1103	16.96	**1.30**	0.169	**26.54**

Experiments: Ablation Study

- **Self-ensembling**
  - Performance increases as the number of VLM random seeds is increased
  - Final choice: 3 seeds for balancing performance and efficiency
- **Ensembling with LPIPS**

MS-COCO	FD-DINO ↓	FID ↓	LPIPS ↓	PSNR ↑
Ours (VLIC)	67.83	2.31	**0.278**	**21.68**
– No ensemble w/ LPIPS	**67.68**	**2.10**	0.280	21.29
– No post-training	82.31	2.40	0.300	21.27
– No self-ensembling	68.36	2.15	0.280	21.53

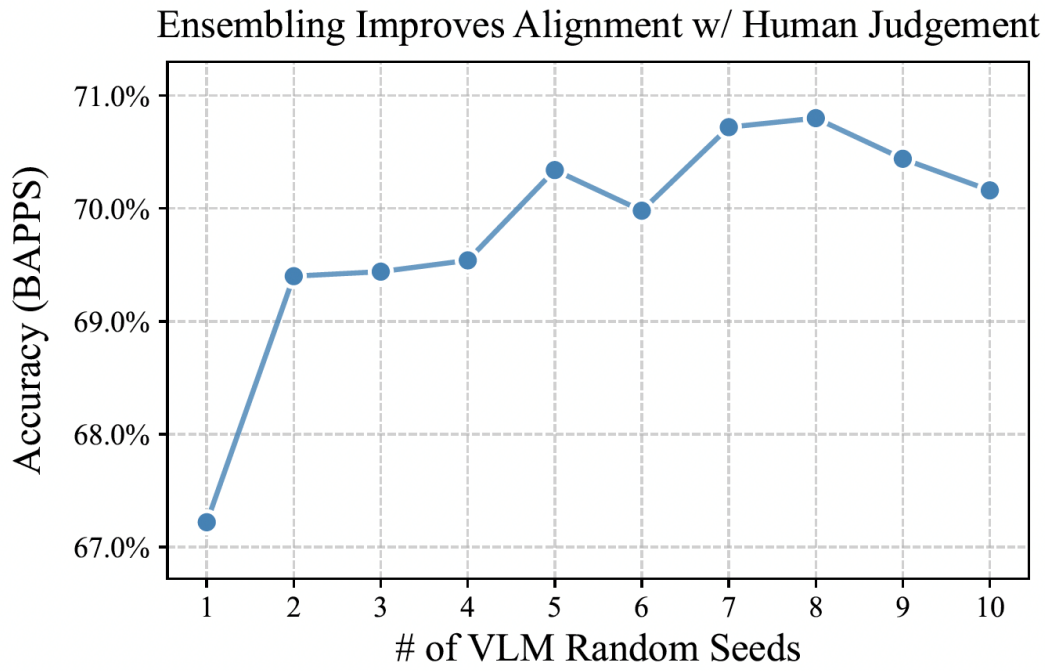


Figure 5. **Scaling self-ensembling.** The VLM becomes more predictive of human judgment on BAPPS [23] as test-time compute (number of VLM seeds) is scaled.

Limitations

- **Ranking hallucination**
  - Misjudging when images are highly similar
  - Fails to be self-consistent
- **More expensive to compute**
  - Each generation must be fed to the LLM for evaluation
- **Adds additional latency**
  - shared by other diffusion-based approaches

Reconstruction 1 | Original | Reconstruction 2

VLM prompted with (Original, 1, 2): **"... Overall, Reconstruction B is better because it preserves the whiskers better than Reconstruction A and also produces a more accurate color for the bridge of the nose and has less blurring for the floral pattern. It is a slight improvement over Reconstruction A …"**

VLM prompted with (Original, 2, 1): **"... In reconstruction B, we have less smudging on the textures of the flower, which are less defined in Reconstruction A, so this is another key win for Reconstruction B. Reconstruction B has less discoloration, so it is better. Therefore, Reconstruction B is superior …"**

Takeaways

- **Off-the-shelf VLMs have learned a visual prior highly correlated with human perception**
  - Utilizing VLM for post-processing and reinforcement learning is beneficial
  - VLM evaluation and training can be conducted in parallel
- **The noisiness and uncertainty of VLM should be considered**
  - Prompt engineering
  - Average over multiple random seeds
  - Use existing metrics to verify VLM judgement

北京大学
PEKING UNIVERSITY

Thanks for listening!