Latent Diffusion Models, Scale Up Generation Quality, and Text-to-Image Generations

Presented by: Yue Yu[†]

†: Statistics Club, Indiana University

Tentative Schedule of Diffusion Model Series

- **10/10**: Overview of generative AI models in Computer Vision, including GANs (Goodfellow et al., 2014), VAEs (Kingma, 2013) and diffusion models (Sohl-Dickstein et al., 2015), and detailed introduction to GANs;
- **10/17**: Detailed introduction to Autoencoders;
- **10/24**: DDPM (Ho et al., 2020), the cornerstone paper about diffusion models, which enables diffusion models to produce high-quality, realistic samples and to be competitive with generative models like GANs and VAEs;

Tentative Schedule of Diffusion Model Series (Cont')

- **11/7**: DDIM (Song et al., 2020), which presents a method to speed up the sampling process in diffusion models without sacrificing much in terms of sample quality, and guided diffusion models (Dhariwal and Nichol, 2021), (Ho and Salimans, 2022), which enhances the flexibility and controllability of generated samples, and are often used in tasks requiring conditional generation, such as text-to-image synthesis;

- **11/21: Recent years' improvements and applications of diffusion models from different aspects, e.g., Latent Diffusion Models (Stable Diffusion, Rombach et al. (2022)) and its subsequent works like conditional control of text-to-image diffusions, namely, unCLIP (Ramesh et al., 2022) and Imagen (Saharia et al., 2022).**

Today's Presentation Outline

- Latent Diffusion Models;
- Scale Up Generation Quality;
- Text-to-Image Diffusion Models.
- Neural Network Architecture of Diffusion Models: Transformer.

Latent Diffusion Model (LDM)

- Latent diffusion model (LDM) (Rombach et al., 2022) runs the diffusion process in the latent space instead of pixel space, making training cost lower and inference speed faster.
- It is motivated by the observation that most bits of an image contribute to perceptual details, and the semantic and conceptual composition still remains after aggressive compression.
- LDM loosely decomposes the perceptual compression and semantic compression with generative modeling learning by:
  - First trimming off pixel-level redundancy with an autoencoder.
  - Then manipulating/generating semantic concepts with a diffusion process on the learned latent space.

Compression Tradeoff in Latent Diffusion Model (LDM)

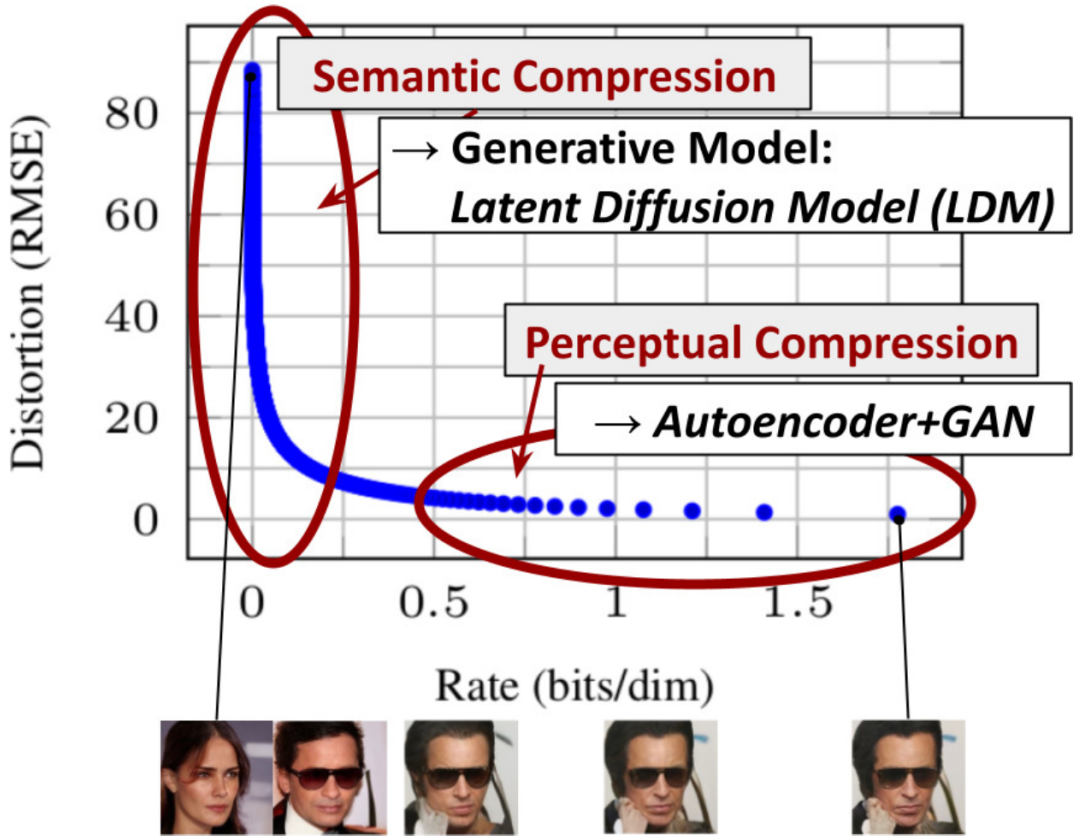

Figure: The plot for tradeoff between compression rate and distortion, illustrating two-stage compressions - perceptual and semantic compression. (Rombach et al., 2022)

Perceptual Compression in Latent Diffusion Models

- **Perceptual Compression Process:**
  - Uses an encoder $\mathcal{E}$ to compress the input image $\mathbf{x} \in \mathbb{R}^{H \times W \times 3}$ to a smaller latent matrix $\mathbf{z} = \mathcal{E}(\mathbf{x}) \in \mathbb{R}^{h \times w \times c}$.
  - Downsampling rate: $f = \frac{H}{h} = \frac{W}{w} = 2^m, m \in \mathbb{N}$.
  - A decoder $\mathcal{D}$ reconstructs the images from the latent matrix, $\tilde{\mathbf{x}} = \mathcal{D}(\mathbf{z})$.
- **Regularization Techniques:**
  - **KL-reg:** Adds a small KL penalty towards a standard normal distribution over the learned latent, similar to VAEs.
  - **VQ-reg:** Incorporates a vector quantization layer within the decoder, similar to VQ-VAE, but the quantization layer's effect is absorbed by the decoder.
- These techniques aim to avoid arbitrarily high variance in the latent spaces and improve the efficiency of the compression process.

Diffusion and Denoising with Cross-Attention

- **Latent matrix $\mathbf{z}$:** The diffusion and denoising processes operate in the latent space.
- **Denoising Model:**
  - A time-conditioned U-Net with cross-attention mechanisms.
  - Capable of handling flexible conditioning for image generation (e.g., class labels, semantic maps, blurred variants).
- **Cross-Attention Design:**
  - Integrates representations from different modalities via cross-attention.
  - Each conditioning input y is mapped by a domain-specific encoder τ_θ to an intermediate representation:

$$\tau_\theta(y) \in \mathbb{R}^{M \times d_\tau}.$$

Cross-Attention Mechanism

Attention Mechanism:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d}}\right) \cdot \mathbf{V}$$

where $\mathbf{Q} = \mathbf{W}_Q^{(i)} \cdot \phi_i(\mathbf{z}_i), \quad \mathbf{K} = \mathbf{W}_K^{(i)} \cdot \tau_\theta(y), \quad \mathbf{V} = \mathbf{W}_V^{(i)} \cdot$
$\tau_\theta(y), \mathbf{W}_Q^{(i)}, \mathbf{W}_K^{(i)}, \mathbf{W}_V^{(i)} \in \mathbb{R}^{d \times d_\epsilon}, \quad \phi_i(\mathbf{z}_i) \in \mathbb{R}^{N \times d_\epsilon}, \quad \tau_\theta(y) \in \mathbb{R}^{M \times d_\tau}.$

- $\mathbf{Q}$ (Query): Represents the input-dependent query vectors that extract relevant information from the key-value pairs. Computed from the latent vector $\phi_i(\mathbf{z}_i)$.

Cross-Attention Mechanism - Explained

- $\mathbf{K}$ (Key): Represents the encoded conditioning information ($\tau_\theta(y)$), which determines the relevance of each query to its corresponding values.
- $\mathbf{V}$ (Value): Represents the content or information stored in the conditioning representation ($\tau_\theta(y)$) that the attention mechanism retrieves.
- The attention score softmax $\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d}}\right)$ computes the alignment between $\mathbf{Q}$ and $\mathbf{K}$, guiding how much of $\mathbf{V}$ to attend to.
- Cross-attention enables the model to incorporate conditioning information (e.g., text or semantic maps) effectively, allowing diffusion models to generate contextually coherent and controllable outputs.

Latent Diffusion Model Summary

- **Pixel Space to Latent Space:**
  - Encoder $\mathcal{E}$ compresses pixel-level input $\mathbf{x}$ into a latent representation $\mathbf{z}$.
  - Decoder $\mathcal{D}$ reconstructs $\tilde{\mathbf{x}}$ from the latent $\mathbf{z}$.
- **Diffusion Process:**
  - Operates entirely in the latent space, reducing computational overhead.
  - Sequence of noise levels applied: $\mathbf{z}_T \rightarrow \mathbf{z}_{T-1} \rightarrow \ldots \rightarrow \mathbf{z}_0$.
- **Conditioning Information:**
  - Flexible inputs: Semantic maps, text, representations, images.
  - Encoded via τ_θ to align modalities with the latent space.

Latent Diffusion Model Architecture

- **Denoising U-Net:**
  - Time-conditioned network removes noise from $\mathbf{z}_T$ to reconstruct latent $\mathbf{z}$.
  - Integrated with **cross-attention** modules:
    - Queries ($\mathbf{Q}$), Keys ($\mathbf{K}$), and Values ($\mathbf{V}$) computed from $\mathbf{z}$ and $\tau_\theta(y)$.
- **Key Features in Design:**
  - Skip connections for enhanced reconstruction fidelity.
  - Conditioning on diverse inputs for adaptable image synthesis.

Latent Diffusion Model Architecture Illustration

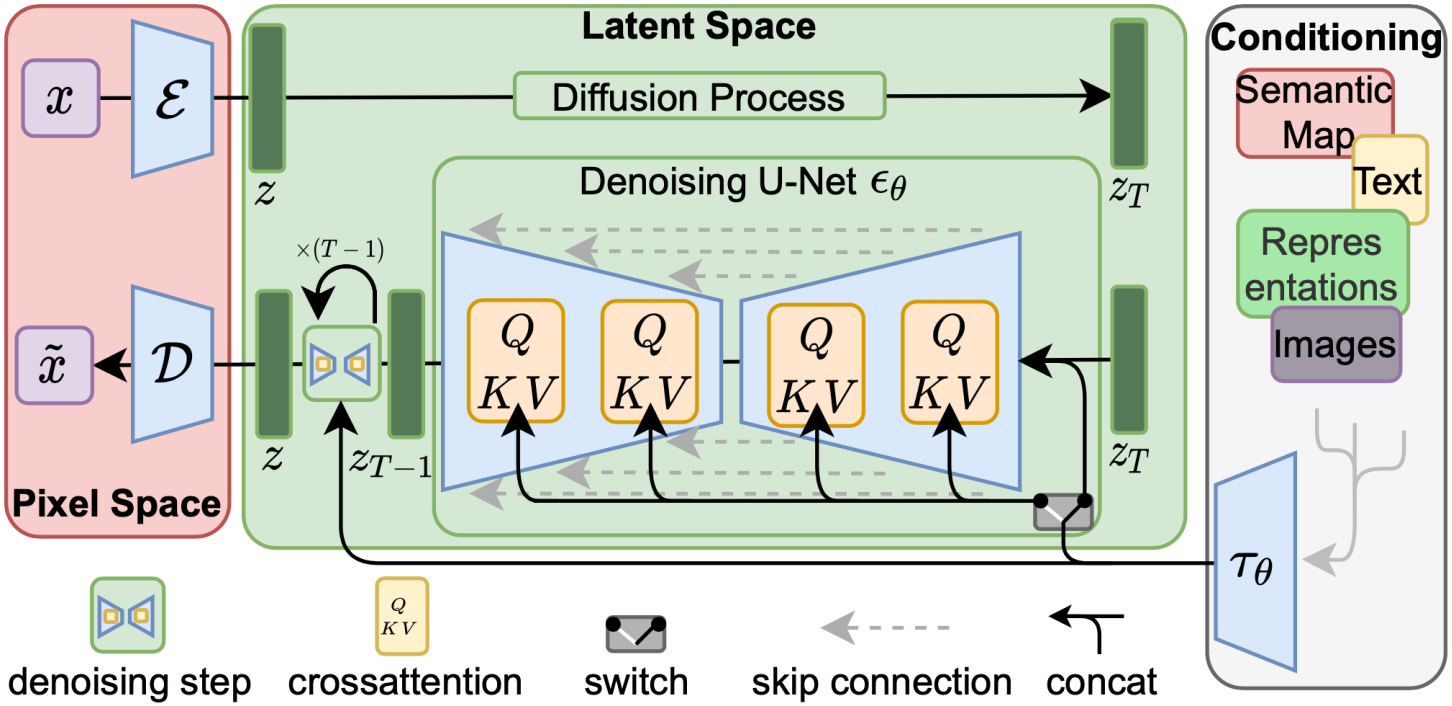

Figure: The architecture of the latent diffusion model (LDM).

Architecture of Different Diffusion Models

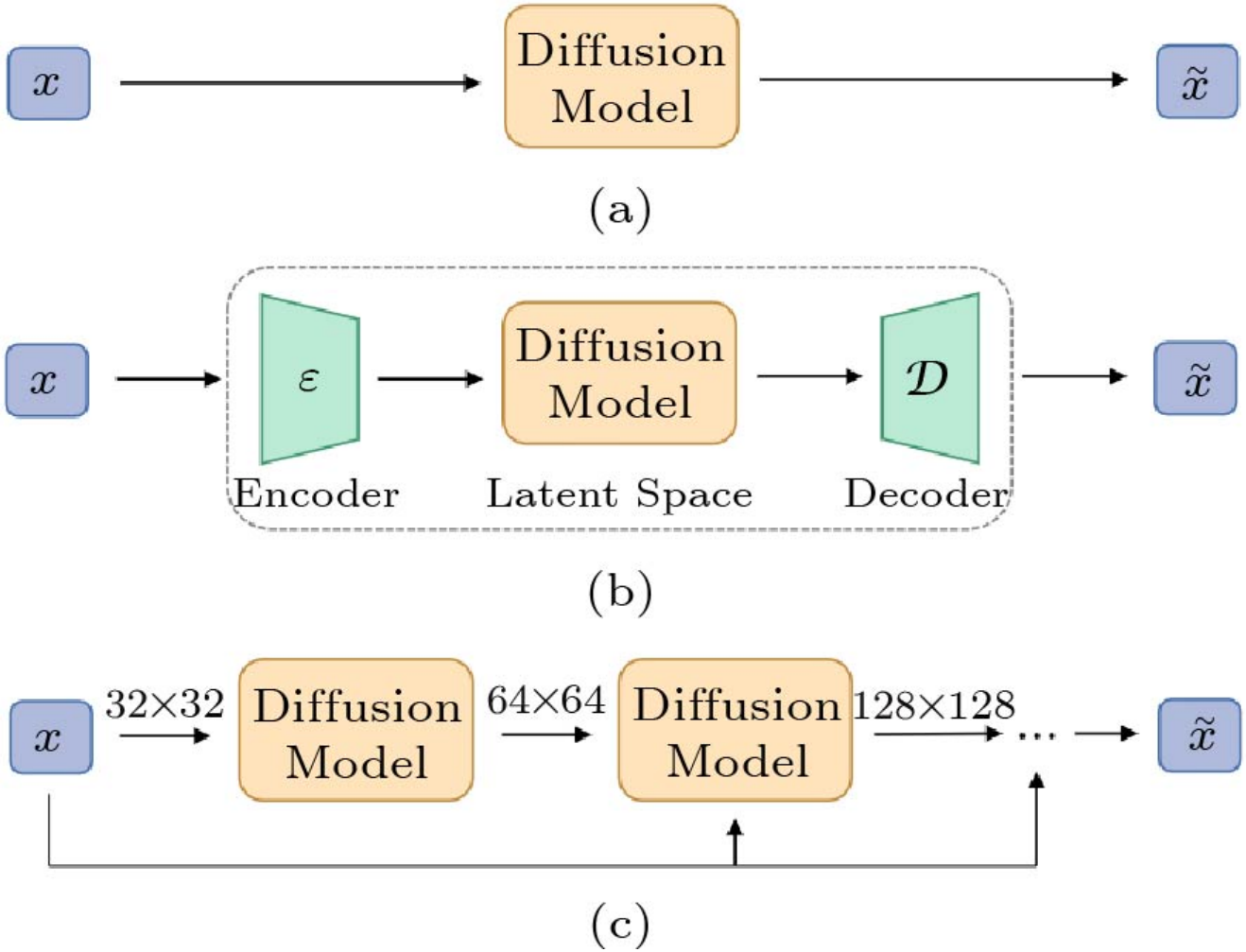

Figure: The architecture of regular, latent and upscale diffusion models.

Scale up Generation Resolution and Quality

- **Objective:** Generate high-quality images at high resolution using a pipeline of diffusion models.
- **Pipeline Design:** Ho et al. (2022) utilizes multiple diffusion models at increasing resolutions.
- **Noise Conditioning Augmentation:**
  - Applies strong data augmentation to the conditioning input $\mathbf{z}$ of each super-resolution model $p_\theta(\mathbf{x}|\mathbf{z})$.
  - Helps reduce compounding errors in the pipeline setup.
- **Model Architecture:** *U-net* is commonly used for high-resolution image generation.

Cascaded Diffusion Illustration

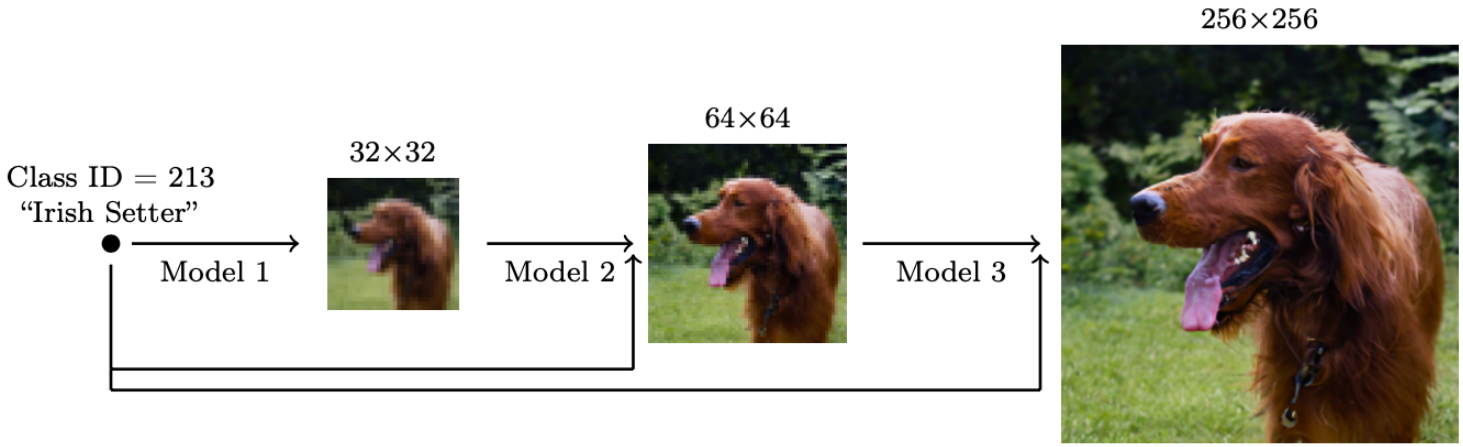

Figure: A cascaded pipeline of multiple diffusion models at increasing resolutions.

Gaussian Blur, Gaussian Noise

Gaussian Blur:

- A smoothing filter that reduces image detail and noise.
- Achieved by convolving the image with a Gaussian kernel.
- Helps in high-resolution models by removing sharp edges or high-frequency variations, ensuring smoother transitions.

Gaussian Noise:

- Random noise following a Gaussian (normal) distribution.
- Adds stochastic perturbations to an image, simulating real-world imperfections.
- Useful for regularizing low-resolution models, preventing overfitting.

Corrupted $\mathbf{z}_t$

- Refers to the intermediate latent variable at timestep t after applying a corruption process.
- Defined as $\mathbf{z}_t \sim q(\mathbf{x}_t|\mathbf{x}_0)$:
  - $q(\mathbf{x}_t|\mathbf{x}_0)$ is the forward diffusion process that gradually adds noise to $\mathbf{x}_0$ to generate $\mathbf{x}_t$.
- Represents a noised version of the original latent variable, which is later fed into the super-resolution model for refinement.

Effective Noise and Conditioning Augmentation

Effective Noise Strategies:

- Gaussian noise is applied at low resolution.
- Gaussian blur is applied at high resolution.

Conditioning Augmentation Techniques:

- **Truncated Conditioning Augmentation:**
  - Stops the diffusion process early at step $t > 0$ for low resolution.
- **Non-Truncated Conditioning Augmentation:**
  - Runs the full low-resolution reverse process until step $t = 0$.
  - Corrupts it by $\mathbf{z}_t \sim q(\mathbf{x}_t|\mathbf{x}_0)$.
  - Feeds the corrupted $\mathbf{z}_t$ into the super-resolution model.

Note: Conditioning noise is applied only during training, not inference.

The Two-Stage Diffusion Model: unCLIP - Overview

- unCLIP (Ramesh et al., 2022) leverages the pretrained CLIP model to generate high-quality, text-guided images.
- Inputs paired training data $(\mathbf{x}, \mathbf{y})$, where $\mathbf{x}$ is the image, and y is the corresponding caption.
- CLIP generates:
  - Text embedding $\mathbf{c}^t(y)$ from caption y.
  - Image embedding $\mathbf{c}^i(\mathbf{x})$ from image $\mathbf{x}$.

The Two-Stage Diffusion Model: unCLIP - Two Models

- **Prior Model:** $P(\mathbf{c}^i|y)$
  - Outputs CLIP image embedding $\mathbf{c}^i$ given the text y.
- **Decoder:** $P(\mathbf{x}|\mathbf{c}^i, [y])$
  - Generates image $\mathbf{x}$ using the CLIP image embedding $\mathbf{c}^i$ and optionally the text y.

The Two-Stage Diffusion Model: unCLIP - Conditional Generation

$$
\begin{aligned}
P(\mathbf{x}|y) &= P(\mathbf{x}, \mathbf{c}^i|y) \\
&= P(\mathbf{x}|\mathbf{c}^i, y)P(\mathbf{c}^i|[y])
\end{aligned}
$$

- The prior model $P(\mathbf{c}^i|y)$ enables the generation of intermediate CLIP image embeddings.
- The decoder $P(\mathbf{x}|\mathbf{c}^i, y)$ generates the final image based on these embeddings and optionally the original text y.
- Since $\mathbf{c}^i$ is deterministic given $\mathbf{x}$, this formulation enables efficient sampling for high-quality image generation.

unCLIP Illustration

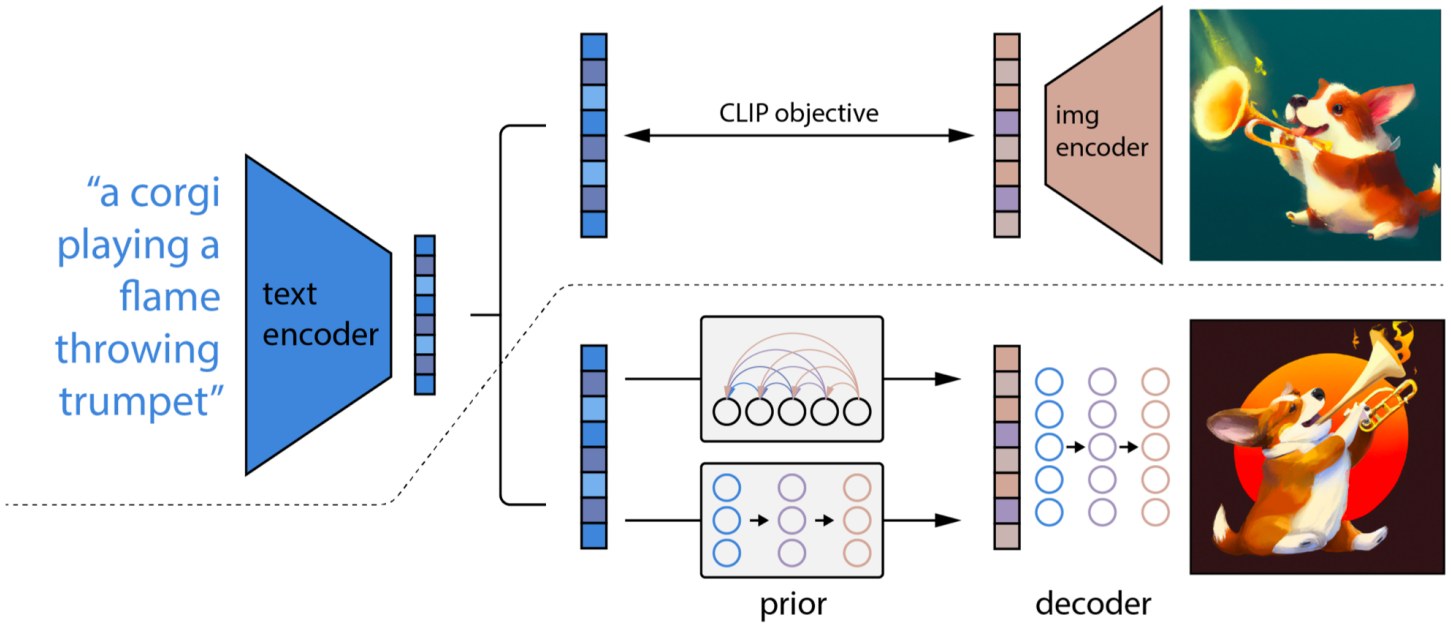

Figure: The architecture of unCLIP.

unCLIP: Two-Stage Image Generation Process

- **Stage 1: Text Embedding Generation:** Given text y, the CLIP model generates a text embedding $\mathbf{c}^t(y)$, enabling zero-shot image manipulation via text.
- **Stage 2: Image Generation:** A prior model $P(\mathbf{c}^i|y)$ refines $\mathbf{c}^t(y)$ to create an image prior $\mathbf{c}^i$. The decoder $P(\mathbf{x}|\mathbf{c}^i, [y])$ generates the image $\mathbf{x}$, conditioned on $\mathbf{c}^i$ and optionally y.
- **Key Features:** Combines semantic text representations with diffusion or autoregressive priors for high-quality, style-preserving image synthesis.

Imagen: Large Language Model for Text-to-Image

- **Model Overview** Imagen (Saharia et al., 2022) replaces CLIP with a pre-trained large language model (T5-XXL) for text encoding in image generation.
- **Advantages of T5-XXL** Larger model size improves image quality and text-image alignment. Performs comparably to CLIP on MS-COCO but outperforms in human evaluations on DrawBench (11-category prompt set).
- **Key Insight** High-capacity text encoders like T5-XXL enhance semantic understanding, crucial for detailed and accurate image synthesis.

Thresholding in Classifier-Free Guidance

- **Trade-off:** Increasing guidance scale w enhances image-text alignment but compromises image fidelity due to train-test mismatch. Since training data $\mathbf{x}$ is restricted to $[-1, 1]$, test data must also follow.

- **Thresholding Strategies:**
  - **Static Thresholding:** Clip $\mathbf{x}$ predictions to $[-1, 1]$.
  - **Dynamic Thresholding:** At each sampling step, compute s as a percentile of absolute pixel values. If $s > 1$, clip $\mathbf{x}$ to $[-s, s]$ and scale by dividing by s.

Efficient U-Net in Imagen

- **Parameter Shifting:** Shift model parameters from high-resolution blocks to low-resolution blocks by adding more residual locks at lower resolutions.
- **Scaling Skip Connections:** Scale skip connections by $\frac{1}{\sqrt{2}}$ for better efficiency.
- **Reordering Operations:** Reverse the order of downsampling and upsampling operations:
  - Move downsampling before convolutions.
  - Move upsampling after convolutions to speed up forward passes.
- **Key Findings:** Noise conditioning augmentation, dynamic thresholding, and efficient U-Net are critical for image quality. However, scaling the text encoder size is more impactful than increasing U-Net size.

Imagen Illustration

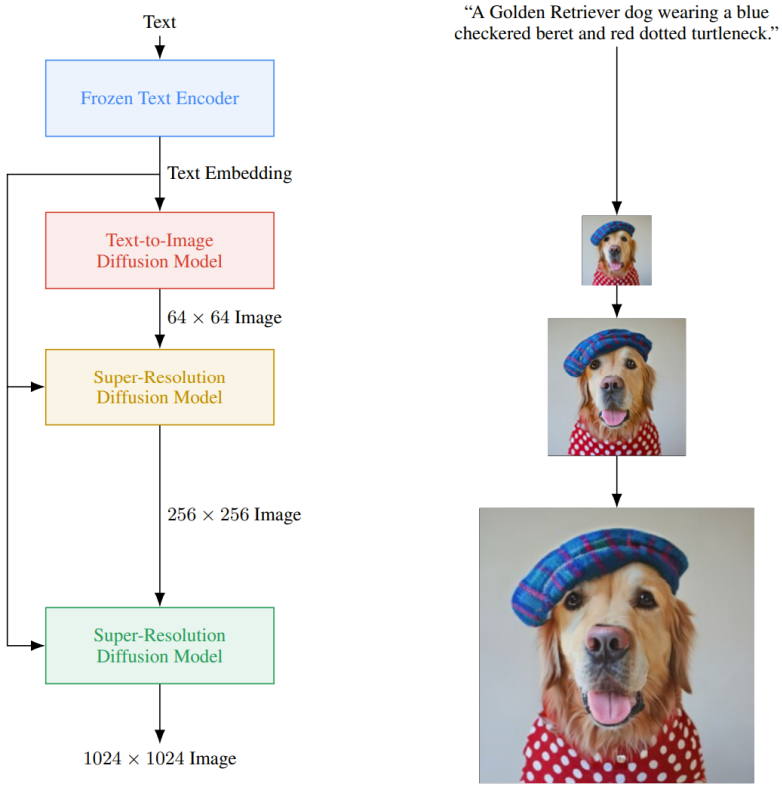

Figure: The Diffusion Transformer (DiT) architecture.

Diffusion Transformer (DiT)

Diffusion Transformer (DiT): Proposed by Peebles and Xie (2023), operates on latent patches, using the design principles of Latent Diffusion Models (LDM).

1. Takes the latent representation of an input $\mathbf{z}$ as input to DiT.
2. **Patchifying:** Converts the noise latent of size $I \times I \times C$ into patches of size p, forming a sequence of patches of size $(I/p)^2$.
3. Processes this sequence of tokens through Transformer blocks.
   - Three designs for generation conditioning on contextual information like timestep t or class label c.
   - **adaLN (Adaptive Layer Norm)-Zero:** Proven most effective compared to in-context conditioning and cross-attention block.

Diffusion Transformer (DiT) (Cont')

4. **adaLN-Zero Implementation:**
   - Scale and shift parameters γ and β are regressed from the sum of embedding vectors of t and c.
   - Dimension-wise scaling parameter α is regressed and applied before residual connections in the DiT block.

5. The Transformer decoder outputs:
   - Noise predictions.
   - Output diagonal covariance prediction.

Diffusion Transformer (DiT) Architecturre

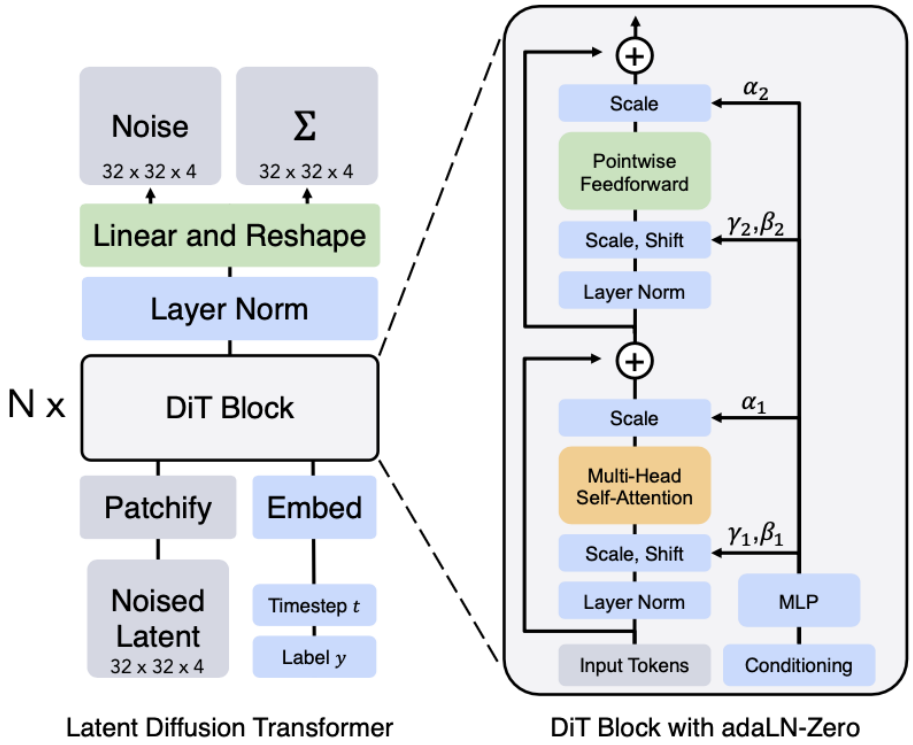

Figure: Diffusion Transformer Architecture. Transformer architecture can be easily scaled up and it is well known for that. This is one of the biggest benefits of DiT as its performance scales up with more compute and larger DiT models are more computational efficient according to the experiments.

Summary: Pros and Cons of Diffusion Models

- **Pros:**
  - Balance tractability and flexibility in generative modeling.
  - Tractable models: Analytically evaluated and cheaply fit data (e.g., Gaussian or Laplace).
  - Flexible models: Fit arbitrary structures in data but are computationally expensive.
  - Diffusion models combine tractability and flexibility effectively.
- **Cons:**
  - Depend on long Markov chains for sampling, making the process time- and compute-intensive.
  - Recent advancements improve efficiency, but sampling remains slower compared to GANs.

Future Research Directions for Diffusion Models

- **Efficiency Improvements:**
  - Reducing the number of diffusion steps without sacrificing generation quality.
  - Developing faster sampling methods to compete with GANs in terms of speed.
- **Conditioning and Control:**
  - Enhancing control over image attributes during generation (e.g., spatial precision).
  - Improving multimodal conditioning for text-to-image and video generation.
- **Scalability:**
  - Scaling to higher resolutions and larger datasets while maintaining tractability.
  - Exploring model performance on complex multimodal tasks.

Future Research Directions for Diffusion Models (Cont')

Applications:

- Expanding applications in fields like drug discovery, protein folding, and physics simulations.
- Integration with downstream tasks such as reinforcement learning or decision-making.

Theoretical Insights:

- Investigating the theoretical foundations of diffusion processes and their relation to other generative models.
- Establishing guarantees for stability and convergence in training and sampling.
- Enhance models' robustness using theories in reinforcement learning.

Questions?

Thank you!

References I

P. Dhariwal and A. Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

J. Ho and T. Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

J. Ho, C. Saharia, W. Chan, D. J. Fleet, M. Norouzi, and T. Salimans. Cascaded diffusion models for high fidelity image generation. *Journal of Machine Learning Research*, 23(47):1–33, 2022.

D. P. Kingma. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

References II

W. Peebles and S. Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023.

A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35: 36479–36494, 2022.

References III

J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015.

J. Song, C. Meng, and S. Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.