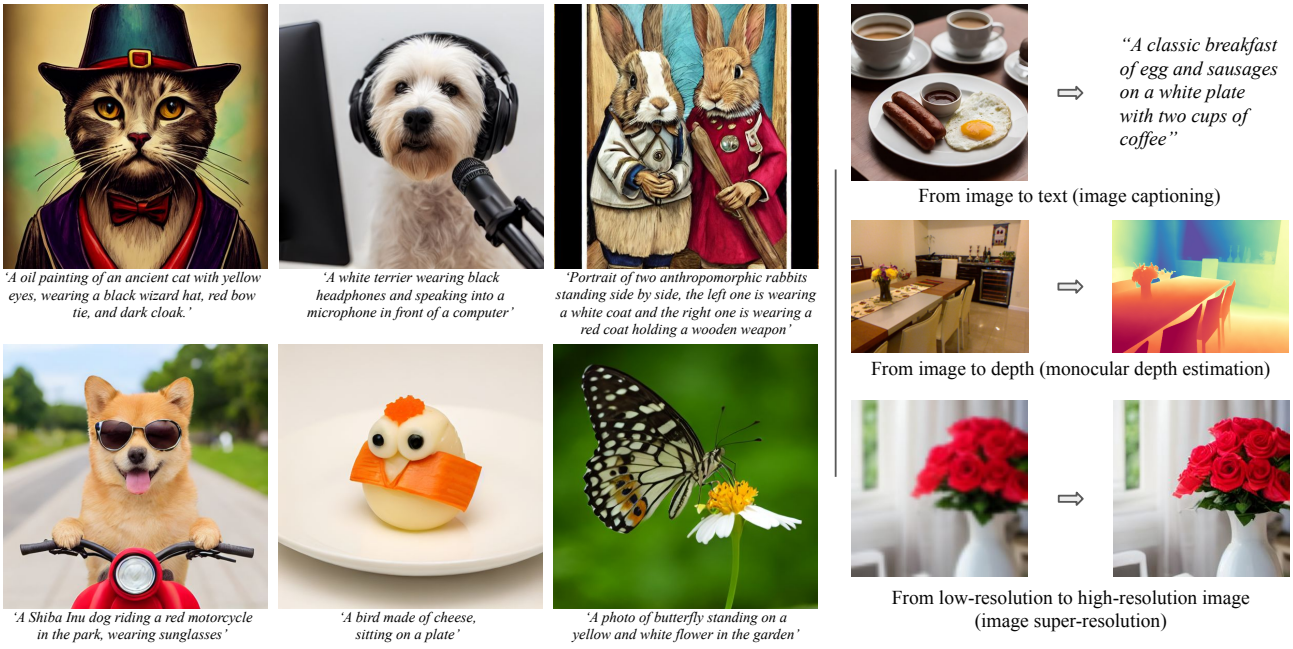
Flowing from Words to Pixels: A Noise-Free Framework for Cross-Modality Evolution

Qihao Liu[1,2] Xi Yin[1] Alan Yuille[2] Andrew Brown[1] Mannat Singh[1]
[1]GenAI, Meta [2]Johns Hopkins University

https://cross-flow.github.io/



(a) Directly evolving text into images for Text-to-Image generation

(b) CrossFlow for various tasks

Figure 1. **We propose CrossFlow, a general and simple framework that directly evolves one modality to another using flow matching with no additional conditioning**. This is enabled using a vanilla transformer without cross-attention, achieving comparable performance with state-of-the-art models on (a) text-to-image generation, and (b) various other tasks, without requiring task specific architectures.

Abstract

Diffusion models, and their generalization, flow matching, have had a remarkable impact on the field of media generation. Here, the conventional approach is to learn the complex mapping from a simple source distribution of Gaussian noise to the target media distribution. For cross-modal tasks such as text-to-image generation, this same mapping from noise to image is learnt whilst including a conditioning mechanism in the model. One key and thus far relatively unexplored feature of flow matching is that, unlike Diffusion models, they are not constrained for the source distribution to be noise. Hence, in this paper, we propose a paradigm shift, and ask the question of whether we can instead train flow matching models to learn a direct mapping from the distribution of one modality to the distribution of another, thus obviating the need for both the noise distribution and conditioning mechanism. We present a general and simple framework, CrossFlow, for cross-modal flow matching. We show the importance of applying Variational Encoders to the input data, and introduce a method to enable Classifier-free guidance. Surprisingly, for text-to-image, CrossFlow with a vanilla transformer without cross attention slightly outperforms standard flow matching, and we show that it scales better with training steps and model size, while also allowing for interesting latent arithmetic which results in semantically meaningful edits in the output space. To demonstrate the generalizability of our approach, we also show that CrossFlow is on par with or outperforms the state-of-the-art for various cross-modal / intra-modal mapping tasks, viz. image captioning, depth estimation, and image super-resolution. We hope this paper contributes to accelerating progress in cross-modal media generation.

1. Introduction

Diffusion models have achieved remarkable success in generating images [14, 55, 63, 65, 66], videos [6, 7, 30, 70], audio [37, 47], and 3D content [42, 59], revolutionizing the field of generative AI. Recently, flow matching [1, 44, 51] has been proposed as a generalization of diffusion models, where models are trained to find an optimal transport probability path between a source noise distribution and the target data distribution. This approach offers simpler, straight-line trajectories compared to the complex, curved trajectories in diffusion paths. As a result, it has been rapidly adopted in the latest state-of-the-art image and video generation models, including LDMs [18] and Movie Gen [58].

Both diffusion and flow-based models are typically trained to learn the mapping from noise to the target distribution. For cross-modal generation tasks such as text-to-image [8, 65], this same mapping from noise to the target modality distribution (*i.e*. the images) is learnt whilst adding a conditioning mechanism for the conditioning modality (*i.e*. the text) such as cross-attention. Unlike denoising diffusion models [29, 73], one relatively unexplored feature of flow matching models is that they are not constrained for the source distribution to be Gaussian noise; instead, the source distribution could be one that is correlated with the target distribution. Compared to noise, learning a mapping from such a distribution should intuitively be *easier* for the model because it has to learn shorter and more efficient probability paths. A question remains however as to what this correlated source distribution could be.

Interestingly, due to the information redundancy between different modalities arising from the same data point, for cross-modal generation tasks, the provided conditioning (*e.g*. the text in text-to-image) resembles such data that is correlated with the target distribution (*e.g*. the images). Hence, in this paper, we propose a paradigm shift for cross-modal generation, and ask the question of whether we can instead train flow matching models to learn a direct mapping from the distribution of one modality to the distribution of another, *hence obviating the need for both the noise distribution and any conditioning mechanism*.

Despite the exciting theoretical motivation, there are several key challenges in practice. First, both diffusion and flow-based models require the source and target distributions to be of the same shape; a requirement that is not satisfied for data distributions from different modalities. Secondly, state-of-the-art methods heavily rely on Classifier-free guidance (CFG) [28] for improved generation quality; a method that is not compatible with cross-modal flow matching due to the lack of a conditioning mechanism to turn on/off since the conditioning information instead lies *within* the source data. As a result, prior work [1, 25, 51] targets the simple setting of mapping between two similar intra-modal distributions, such as human faces to cat faces [51].

In this work, we present key architecture design solutions for overcoming these challenges: First, we employ a Variational Encoder for encoding the source modality data distribution to the same shape as the target modality, and show that the resulting regularization in the source distribution is essential for generation performance. Secondly, we enable CFG in cross-modal flow matching through the introduction of a binary conditioning indicator during training, and demonstrate the quantitative benefits of this approach compared to alternative CFG methods. We present CrossFlow; a general framework for mapping between two different modalities without the need for any conditioning mechanism or noise distribution. Typically, different cross-modal generation tasks require task-specific architectural and training modifications, but CrossFlow works for different tasks without any such changes.

Using the ubiquitous albeit challenging text-to-image (T2I) generation task as our primary setting, we show the significant result that CrossFlow outperforms commonly used flow matching baselines, given the same training data, model size, and training budget, all *without requiring any cross-attention layers*. CrossFlow exhibits improved scaling behavior over standard flow matching using cross-attention when scaling training steps or model size, and is also compatible with a variety of Large Language Models (LLMs), including CLIP [60], T5 [61], and Llama3 [16]. Additionally, we demonstrate that since our approach encodes the source distribution into a regularized continuous space with semantic structure, CrossFlow enables exciting new *latent arithmetic* for the text-to-image task, *e.g*., $\mathcal{L}$("A dog with a hat") + $\mathcal{L}$("Sunglasses") – $\mathcal{L}$("A hat") creates an image of a dog wearing sunglasses without a hat. Lastly, CrossFlow enables bi-directional mapping between modalities, allowing, for instance, the inversion of text-to-image models to serve as image-to-text (captioning) models.

We demonstrate the general-purpose nature of CrossFlow on various cross-modal/intra-modal tasks: image-to-text (image captioning), image-to-depth (depth estimation), and low-resolution to high-resolution image (super-resolution). CrossFlow achieves comparable or superior performance to various state-of-the-art methods on all three tasks, without requiring task specific architectures. For example, in image captioning, CrossFlow directly projects images into a textual latent space to generate captions, achieving state-of-the-art performance using only a simple text decoder that maps textual latents to discrete tokens. Results are shown in Fig. 1. We hope this paper contributes to accelerating the progress in cross-modal media generation.

2. Related Work

Diffusion models and rectified flow. Starting from Gaussian noise, diffusion [29, 71] and score-based [32, 72] generative models progressively approximate the reverse ODE

of a stochastic forward process to generate data. These models have driven significant advances across various domains, particularly in high-fidelity image [4, 14, 31, 49, 57], video [6, 7, 30, 58, 70], and 3D generation [42, 48, 50, 59]. Recently, rectified flow models [1, 44, 51], such as flow matching, have been proposed to improve the generative process by enabling a transport map between two distributions. They enable faster training and sampling by avoiding complex probability flow ODEs.

Directly bridging distributions. Flow Matching theoretically allows for arbitrary distributions as the source distribution, which can then be used for direct evolution. Various approaches have been proposed in this direction, such as InterFlow [1], α-blending [25], data-dependent coupling [3], and Schrödinger Bridge [12, 45, 46, 68, 74, 75, 81]. They provide important theoretical support for using ODE-based methods to bridge two arbitrary distributions. However, they are still limited to similar distributions from the same domain, such as image-to-image translation (*e.g.*, faces-to-faces [51, 81] or sketches-to-images [46]). As a step forward, CrossFlow focuses on learning the mapping between data distributions arising from even different modalities.

Text-to-image generation. Text-to-image generation [8, 11, 18, 55, 62, 63, 65, 66, 83] has rapidly advanced with diffusion and later flow matching models. This task bridges two critical and complex domains: language and vision. Existing methods typically integrate text encoders, such as LLMs, into diffusion models through additional conditioning mechanisms, with cross-attention being the most prevalent [18, 58]. However, these approaches increase model complexity and require extra parameters. We demonstrate that CrossFlow improves over standard flow matching with better scaling characteristics, and is comparable to prior work, despite a simpler architecture.

Cross-modal / intra-modal mapping. Various tasks can be framed as cross-modal/intra-modal mapping problems, including image captioning [20, 24, 38, 39, 54, 80, 82], depth estimation [5, 15, 35, 40, 41, 64, 79], and image super-resolution [19, 67]. However, due to the significant differences between modalities or distributions, previous methods have typically relied on task-specific designs. For example, Bit Diffusion [10] encodes text into binary bits and uses a diffusion model with self-conditioning for captioning. Flow-based super-resolution models, such as CFM [19], still require the low-resolution image as extra conditioning, and also add Gaussian noise to the input. In contrast, our CrossFlow uses the same unified framework across all these tasks without extra conditioning or noise.

3. Preliminaries

Flow Matching. We consider a generative model that defines a mapping between samples z_0 from a source distribution p_0 to samples z_1 of a target distribution p_1 via the ordinary differential equation (ODE): $dz_t = v_\theta(z_t, t)dt$. Here, v_θ is the velocity parameterized by the weights θ of a neural network, and $t \in [0, 1]$ is the time-step. Specifically, Flow Matching [1, 44, 51] defines the forward process as:

$$z_t = tz_1 + (1 - (1 - \sigma_{min})t)z_0 \tag{1}$$

and $\sigma_{min} = 10^{-5}$. Ground truth velocity is computed as:

$$\hat{v}_t = \frac{dz_t}{dt} = z_1 - (1 - \sigma_{min})z_0 \tag{2}$$

To achieve this, a network $v_\theta(z_t, t)$ is trained to predict velocity by minimizing the mean squared error (MSE) between its output and the target $\hat{v}_t$. This constructs a continuous path between z_0 and z_1 at any time-step $t \in [0, 1]$.

As discussed earlier, flow matching enables evolving a sample z_1 from an arbitrary source distribution p_0. But prior work [18, 58] typically starts from Gaussian noise $z_0 \sim \mathcal{N}(0, 1)$, and computing the velocity with additional condition c incorporated through various methods, *e.g.*, cross-attention [18, 58], channel-wise concatenation [23].

Classifier-free guidance. CFG [28] is a broadly used technique that enhances sample quality in *conditional* generative models by jointly training a single model on conditional and unconditional objectives. This is achieved through randomly dropping the condition c during training with a certain probability p. Sampling is performed by extrapolating between conditional and unconditional denoising $v_\theta(z_t, c)$ and $v_\theta(z_t)$ with a scaling factor ω:

$$\tilde{v}_\theta(z_t, c) = \omega v_\theta(z_t, c) + (1 - \omega)v_\theta(z_t) \tag{3}$$

It significantly improves the generation quality and fidelity by guiding the samples towards higher likelihood of the condition c, which plays a crucial role in state-of-the-art media generation models [8, 18, 58, 63].

4. CrossFlow

In this section, we discuss the various components of our approach: a Variational Encoder (VE) to encode the inputs in Sec. 4.1, using flow matching to evolve from the source to the target distribution in Sec. 4.2, and finally, applying CFG in this setting for improving quality and fidelity in Sec. 4.3.

4.1. Variational Encoder for Encoding Inputs

Flow matching models require the source distribution p_0 to have the same shape as the target distribution p_1. In particular, given an input x, we need to convert it to the source latent z_0, which has the same shape as the target latent z_1. An intuitive solution is to use an encoder $\mathcal{E}$ to convert x to z_0, *i.e.*, $z_0 = \mathcal{E}(x)$, which can preserve most of the input information as shown in the Supp. However, directly evolving from $\mathcal{E}(x)$ to z_1 is problematic. We find that it is essential

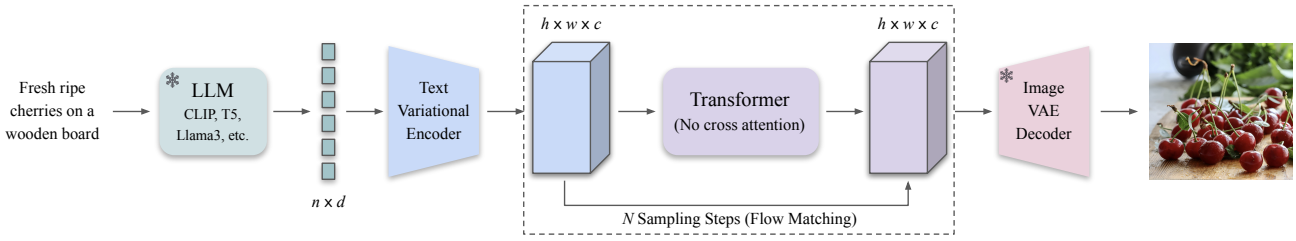


Figure 2. **CrossFlow Architecture.** CrossFlow enables direct evolution between two different modalities. Taking text-to-image generation as an example, our T2I model comprises two main components: a Text Variational Encoder and a standard flow matching model. At inference time, we utilize the Text Variational Encoder to extract the text latent $z_0 \in \mathbb{R}^{h \times w \times c}$ from text embedding $x \in \mathbb{R}^{n \times d}$ produced by any language model. Then we directly evolve this text latent into the image space to generate image latent $z_1 \in \mathbb{R}^{h \times w \times c}$.

to formulate z_0 as a regularized distribution for the source in order for flow matching to work well. To address this, we propose using a VE to convert x to z_0. Formally, instead of directly predicting z_0, we predict its mean $\bar{\mu}_{z_0}$ and variance $\bar{\sigma}_{z_0}$, and then sample the latent $z_0 \sim \mathcal{N}(\bar{\mu}_{z_0}, \bar{\sigma}^2_{z_0})$. This enables us to convert the given input x into latent z_0 with a regularized distribution, which can then be gradually evolved into the target distribution z_1 with flow matching.

The VE can be trained with a standard Variational Autoencoding objective (VAE) [36] comprising of an encoding loss and the KL-divergence loss. For the encoding loss, the VE is trained to minimize a loss between the output z_0 and a target $\hat{z}$. For a VAE this loss would be a reconstruction loss like MSE between the input x and the decoder $\mathcal{D}$'s output, $\texttt{MSE}(\mathcal{D}(z_0), x)$. But since we simply need a encoder and not an autoencoder, we don't restrict ourselves to a VAE.

4.2. Training CrossFlow

For each training sample, we start with an input-target pair (x, z_1). We apply the VE to x to encode it to a latent z_0 with the same shape as z_1. Next, we employ a transformer model v_θ trained for flow matching as per Equations 1 and 2. The VE can be trained prior to training v_θ or concurrently. We show in Sec. 5.2 that jointly training the Variational Encoder with flow matching results in improved performance.

Specifically, we jointly train the VE with the flow matching model using a sum of flow matching MSE loss L_{FM}, and the losses for Variational Encoder training (encoding loss L_{Enc} and KL-divergence loss L_{KL}):

$$\begin{aligned} L &= L_{FM} + L_{Enc} + \lambda L_{KL} \\ &= \texttt{MSE}(v_\theta(z_t, t), \hat{v}) + \texttt{Enc}(z_0, \hat{z}) \\ &\quad + \lambda \texttt{KL}(\mathcal{N}(\bar{\mu}_{z_0}, \bar{\sigma}^2_{z_0}) || \mathcal{N}(0, 1)) \end{aligned} \tag{4}$$

where λ is the KL-divergence loss weight. Eq. 4 outlines the general form of the loss function across tasks, where L_{enc} varies by task. Sec. 4.4 discusses text-to-image generation and choices for L_{Enc}. More details in the Supp.

4.3. Classifier-Free Guidance with an Indicator

CFG [28] has become the standard low-temperature sampling method for enhancing multi-modal alignment and improving quality. However, it can only be applied to generation methods that accept an additional conditioning input c, since the guidance signal relies on the difference between conditional and unconditional predictions $v_\theta(z_t, c)$ and $v_\theta(z_t)$. Recently, Autoguidance (AG) [34] has been introduced as a method to improve both conditional and unconditional generation, by guiding with a smaller, less-trained 'bad model'. However, it underperforms compared to standard CFG. AG also requires training a separate bad model, and its performance varies dramatically based on the choice of the bad model. While using an under-trained version of the same model narrows the search space, it affects performance and is impractical for large models due to the need to load two models during inference.

We instead aim to support CFG for CrossFlow, which is as accessible and performant as CFG is for standard flow matching. To enable CFG without the presence of an explicit conditioning input c, we introduce CFG with indicator. Specifically, our model is of the form $v_\theta(z_t, 1_c)$, where $1_c \in \{0, 1\}$ is an indicator to specify conditional *vs*. unconditional generation. The model evolves from z_0 to z_1 when $1_c = 1$, and from z_0 to z_1^{uc} when $1_c = 0$, where z_1^{uc} represents any sample from the target distribution p_1 other than z_1. During training, we employ two learnable parameters, g^c and g^{uc}, corresponding to conditional and unconditional generation, respectively. Depending on 1_c, the appropriate learnable parameter is concatenated with the transformer input tokens along sequence dimension. We randomly sample the indicator with an unconditional rate of 10%, as per standard practice. The insight behind the CFG indicator is similar to that of standard CFG. In this approach, $v_\theta(z_t, 1)$ is trained to map z_0 to a specific region of the target manifold, while $v_\theta(z_t, 0)$ is trained to map z_0 to the entire target manifold to generate arbitrary unrelated images.

4.4. Flowing from Text to Image

Now, we consider text-to-image generation as the archetypal task to leverage CrossFlow. We start with the input text embedding $x \in \mathbb{R}^{n \times d}$ with token length n and dimension d, and use our Text VE to extract the corresponding text latent $z_0 \sim \mathcal{N}(\bar{\mu}_x, \bar{\sigma}^2_x)$. While our approach is agnostic to

pixel *vs*. latent image generation, we consider image generation in the latent space for efficiency, and leverage a pre-trained VAE to obtain the image latent from the input image I, which serves as our target z_1. Then, we employ the vanilla flow matching [44] model to predict $v(z_t, t)$ between z_0 and z_1. The pipeline for performing text-to-image generation with CrossFlow is illustrated in Fig. 2. We discuss how to train a performant Text Variational Encoder next.

4.4.1. Text Variational Encoder

Training the Text VE is challenging, as this involves compressing the text embeddings to small latent space (*e.g.*, 77×768 CLIP tokens to $4 \times 32 \times 32$ image latents for 256px generation, $14.4\times$ compression). We explore various methods to train VEs for CrossFlow. The straightforward approach is to simply train a VAE with a MSE reconstruction loss. While this approach achieves very low reconstruction errors, we find that it does not capture semantic concepts well, leading to sub-optimal image generations.

Contrastive loss. We explore contrastive losses, which produce representations with strong semantic understanding when training on samples within the same modality [9, 56] and on different modality pairs [60]. To produce the contrastive targets for the VE, we either use the input text embedding x (text-text contrastive), or the paired image I for the text (image-text contrastive). Given the target, we employ a simple encoder to project it into a feature space with the same shape as z_0, resulting in a representation denoted as $\hat{z}$. We then encourage semantic similarity between z_0 and $\hat{z}$ using the contrastive CLIP loss [60]. During training, the batch-wise contrastive loss is computed as $L_{Enc} = \texttt{CLIP}(z_0, \hat{z})$. We ablate this choice in Sec. 5.2 and find that contrastive loss works significantly better than the VAE reconstruction loss, with the image-text loss working slightly better than the text-text loss.

5. Experiments

We first evaluate CrossFlow on text-to-image generation, demonstrate its scalability, and showcase some interesting applications with latent arithmetic in Sec. 5.1. Then, we ablate our main design decisions through ablation studies in Sec. 5.2. Finally, we further explore CrossFlow's performance on three distinct tasks: image captioning, monocular depth estimation, and image super-resolution in Sec. 5.3.

5.1. Text-to-Image Generation

Experimental setup. Scientifically comparing T2I models is challenging due to diverse training datasets, often including proprietary data, and varying training conditions. In addition, our method represents a new paradigm for utilizing diffusion models, distinct from previous T2I approaches. Therefore, we primarily compare our model with the widely used "standard flow matching baseline" that starts from noise and uses text cross-attention. For fairness, both CrossFlow and the baseline share the same codebase, training recipe, dataset, and budget. Unlike the baseline, which requires cross-attention after each self-attention layer, our model only relies on self-attention, reducing parameters per layer. To account for this, we adjust the number of layers to match model sizes. For both methods, we use a grid search to find the optimal CFG scale. We also compare CrossFlow with state-of-the-art T2I models to demonstrate that our approach is competitive with those established methods.

Method	#Params (B)	#Steps (K)	FID ↓	CLIP ↑
Standard FM (Baseline)	1.04	300	10.79	0.29
CrossFlow (Ours)	0.95	300	10.13	0.29

Table 1. **Comparison between our CrossFlow and standard flow matching with cross-attention.** Both models are trained with the same settings. We find that our model slightly outperforms standard flow matching baseline in terms of *zero-shot* FID-30K and achieves comparable performance on the CLIP score.

Architecture. Our model enables the use of vanilla Transformer [76] with self-attention layers and feed-forward layers. We use DiMR [49] as the flow matching backbone, a variant of Diffusion Transformer (DiT) [57] which replaces the parameter-heavy MLP in adaLN-Zero with a lightweight Time-Dependent Layer Normalization. For the Text VE, we apply stacked Transformer blocks, followed by a linear layer to project the output into the target shape.

Training details. We use a proprietary dataset with about 350M image-text pairs to train both CrossFlow and our ablations. Our text encoder is based on CLIP [60] with a fixed sequence length of 77 text tokens. We use a pre-trained and frozen VAE from LDM [65] to extract image latents. Logit-normal sampling [18] is used to bias the training timesteps. All T2I models are trained using the same settings: an image resolution of 256×256, a batch size of 1024, a base learning rate of 1×10^{-4} with 5000 warm-up steps, and an AdamW optimizer [53] with $\beta_1 = \beta_2 = 0.9$ and a weight decay of 0.03, and a KL loss weight of $\lambda = 1 \times 10^{-4}$. We train our largest model (0.95B) on 256×256 for 600K iterations, then finetune it on 512×512 for an additional 240K iterations for higher resolution generation.

Evaluation metrics. We evaluate all models on the COCO validation set [43] and report FID [27] and CLIP score [26, 60]. Following previous works, we report *zero-shot* FID-30K, where 30K prompts are randomly sampled from the validation set, and the generated images are compared to reference images from the full validation set. Additionally, we also evaluate our models on GenEval benchmark as it exhibits strong alignment with human judgment [22].

5.1.1. CrossFlow *vs*. Standard Flow Matching

We compare our CrossFlow with widely used cross-attention baseline in Tab. 1. Both models are trained and tested under the same settings. The results show that Cross-

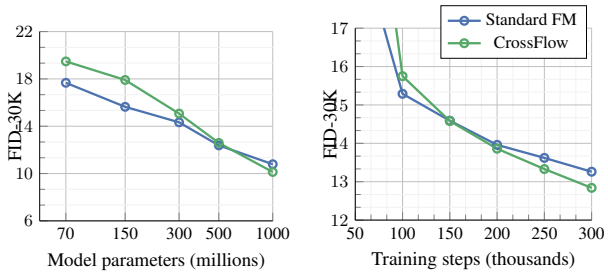


Figure 3. **Performance *vs*. Model Parameters and Iterations.** We compare the baseline of starting from noise with text cross-attention with CrossFlow, while controlling for data, model size and training steps. *Left*: Larger models are able to exploit the cross-modality connection better. *Right*: CrossFlow needs more steps to converge, but converges to better final performance. Overall, CrossFlow scales better than the baseline and can serve as the framework for future media generation models.

Method	#Params.	FID-30K ↓ *zero-shot*	GenEval ↑ score
DALL·E [62]	12.0B	27.50	-
GLIDE [55]	5.0B	12.24	-
LDM [65]	1.4B	12.63	-
DALL·E 2 [63]	6.5B	10.39	0.52
LDMv1.5 [65]	0.9B	9.62	0.43
Imagen [66]	3.0B	7.27	-
RAPHAEL [77]	3.0B	6.61	-
PixArt-α [8]	0.6B	7.32	0.48
LDMv3 (512^2) [18]	8.0B	-	0.68
CrossFlow	0.95B	9.63	0.55
CrossFlow (Sin-Cos)	0.95B	8.95	0.57

Table 2. **Comparison to recent T2I models.** For GenEval, we report overall scores here and task-specific results in the Supp. CrossFlow achieves comparable results to state-of-the-art models by evolving text directly into images. CrossFlow (Sin-Cos) replaces simple linear flow matching with sin-cos matching [2].

Flow achieves comparable performance, with slightly better *zero-shot* FID-30K compared with widely used flow matching baselines with cross-attention.

Scaling characteristics. We investigate the scalability of CrossFlow in Fig. 3 and compare it with standard flow matching. We train both approaches across 5 different model sizes, ranging from 70M to 1B parameters, with the same training settings, for 300K iterations. At smaller scales, CrossFlow underperforms the baseline, likely due to the lack of sufficient parameters to model the complex relationships between two modalities. But excitingly, as the model size increases, the *zero-shot* FID-30K improves more for our approach. Next, we evaluate the effect of varying the training iterations. We notice similarly that CrossFlow improves more as we increase training iterations.

While CrossFlow initially underperforms standard flow matching at small scales, increasing the model size and training iterations improves it significantly, even enabling it to surpass standard flow matching. We attribute this to the fact that CrossFlow generates images by directly evolving from the source distribution where different sub-regions correspond to different semantics. In contrast, standard flow matching may generate the same semantics from the entire source distribution, while exploiting the inductive biases afforded by text cross-attention. Ultimately, this works in favor of CrossFlow, as the learnt cross-modal paths and fewer inductive biases result in improved scaling characteristics with both model size and training iterations.

5.1.2. State-of-the-art Comparison

Finally, we compare CrossFlow with state-of-the-art text-to-image models and report results in Tab. 2. We additionally explore sin-cos matching [2] and find it improves over vanilla linear flow matching. We achieve a *zero-shot* FID-30K of 8.95 on COCO, and a GenEval score of 0.57, demonstrating performance comparable with the state-of-the-art. Note that our model uses only 630 A100 GPU-days for training, whereas other methods like DALL·E 2 [63] typically require thousands of A100 GPU days. These results suggest that CrossFlow is a simple and promising direction for state-of-the-art media generation.

5.1.3. Arithmetic Operations in Latent Space

Unlike previous diffusion or flow matching models, CrossFlow offers a unique property: arithmetic operations in the input latent space translate to similar operations in the output space. This is made possible since CrossFlow transforms the source space (*i.e*., the text latent space for T2I) into a regularized continuous space, where a uniform representation shape is shared across all texts. We showcase two examples of this, latent interpolation, and latent arithmetic. For latent interpolation, we use the Text Variational Encoder to generate text latents from two different text inputs, and then interpolate between them to produce images. As shown in Fig. 4, CrossFlow enables visually smooth linear interpolations, even between disparate prompts. Next, we showcase arithmetic operations in Fig. 5, in which we apply addition and subtraction in the text latent space, and find that the resulting images exhibit corresponding semantic modifications to the original image. This shows that CrossFlow formulates meaningful and well-structured semantic paths between the source and target distributions, providing additional capabilities and more control over standard flow matching approaches. See the Supp for further details.

5.2. Ablation Study

We conduct various ablation experiments to verify the effectiveness of the proposed designs in Tab. 3.

Variational Encoder *vs*. standard encoder. Compared to a standard encoder or even adding Guassian noise like CFM [19], a Variational Encoder significantly improves the generation quality, with significant gains in the FID. This shows that forming a regularized distribution for the source domain is a crucial step for cross-modal flow matching.

Figure 4. **CrossFlow provides visually smooth interpolations in the latent space.** We show images generated by linear interpolation between the first (left) and second (right) text latents. CrossFlow enables visually smooth transformations of object direction, composite colors, shapes, background scenes, and even object categories. Please zoom in for better visualization. For brevity, we display only 7 interpolating images here; additional interpolating images can be found in the Supp.

Figure 5. **CrossFlow allows arithmetic in text latent space.** Using the Text Variational Encoder (VE), we first map the input text into the latent space z_0. Arithmetic operations are then performed in this latent space, and the resulting latent representation is used to generate the corresponding image. The latent code z_0 used to generate each image is provided at the bottom.

Text encoder	FID ↓	CLIP ↑
Encoder	66.65	0.20
Encoder + noise	59.91	0.21
Variational Encoder	40.78	0.23

(a) **Variational Encoder** *

Loss	FID ↓	CLIP ↑
T-T Recon.	40.78	0.23
T-T Contrast.	34.67	0.24
I-T Contrast.	33.41	0.24

(b) **Text VE loss** *

Method	FID ↓	CLIP ↑
No guidance	33.41	0.24
AG	26.36	0.25
CFG indicator	24.33	0.26

(c) **CFG with indicator**

Model	FID ↓	CLIP ↑
CLIP (0.4B)	24.33	0.26
T5-XXL (11B)	22.28	0.27
Llama3 (7B)	21.20	0.27

(d) **Language Model**

Train strategy	FID ↓	CLIP ↑
2-stage separate training	32.55	0.24
Joint training	24.33	0.26
2-stage w/ joint finetuning	23.79	0.26

(e) **Training strategy**

Table 3. **Ablation study** on Text Variational Encoder, training objective, CFG, language models, and training strategy. We conduct ablation study on our smallest model (70M), reporting *zero-shot* FID-10K and CLIP scores. Final settings used for CrossFlow are underlined. AG: Autoguidance. *: results without applying CFG.

Joint training *vs*. two-stage training. We consider three training strategies: (1) jointly training the VE and flow matching from scratch, (2) training the VE first and then training flow matching with a fixed VE, and (3) training the VE first and then training the flow matching while jointly fine-tuning VE. We observe that it is important to update the VE when training the flow matching, either through joint training from scratch, or finetuning the VE jointly with flow matching. Initializing with a pre-trained VE and then jointly training improves convergence speed by about 35%, but we opt to jointly train both models from scratch on account of the simplicity, and for fair comparisons with baselines.

CFG indicator. We evaluate the performance of our model when leveraging our proposed CFG indicator techinuqe. We also evaluate Autoguidance (AG) [34], which utilizes two models for inference – we use an under-trained version of the same model as the bad model, while using a grid-search to find the best under-trained checkpoint. While AG improves FID and also image-text CLIP alignment slightly, our CFG indicator works better than AG in terms of both FID and CLIP alignment while only using a single model trained with standard CFG settings. Qualitatively, our approach produces much higher fidelity images compared to both alternatives, as shared in the Supp.

Text VE loss. We explore reconstruction and contrastive objectives for the encoder loss L_{Enc} when training the text VE. We find that contrastive loss, which enhances semantic understanding, significantly outperforms reconstruction loss on input text embeddings. Moreover, image-text contrastive loss slightly surpasses text-text contrastive loss.

Effect of different language models. We evaluate CrossFlow with various language models trained with different objectives. Specifically, we evaluate CLIP [60] (contrastive image-text), T5-XXL's encoder [61] (encoder-decoder), Llama3-7B [16] (decoder-only). We use 77 tokens for

Method	B@4 ↑	M ↑	R ↑	C ↑	S ↑
MNIC [20]	30.9	27.5	55.6	108.1	21.0
MIR [38]	32.5	27.2	-	109.5	20.6
NAIC-CMAL [24]	35.3	27.3	56.9	115.5	20.8
SATIC [82]	32.9	27.0	-	111.0	20.5
SCD-Net [54]	37.3	28.1	58.0	118.0	21.6
CrossFlow-T2I (Ours)	33.1	27.0	56.4	111.2	20.3
CrossFlow (Ours)	36.4	27.8	57.1	116.2	20.4

Table 4. **Image captioning on COCO Karpathy split.** CrossFlow directly evolves from image to text, achieving comparable performance to state-of-the-art models on image captioning. For a fair comparison, we consider non-autoregressive methods that are trained without CIDEr optimization. CrossFlow-T2I achieves captioning by simply inverting our text-to-image CrossFlow model.

all language models, resulting in text embeddings of size 77×768, 77×4096, 77×4096, respectively. We train a separate Text VE for each language model, projecting the text embeddings into the target image latent shape ($4 \times 32 \times 32$). CrossFlow works well with all language models regardless of their training objectives and embedding sizes. As expected, our performance improves with better text representations. Due to compute restrictions however, we use the light-weight CLIP model for our main experiments.

5.3. CrossFlow for Various Tasks

We further evaluate CrossFlow on three distinct tasks that involve cross-modal / intra-modal evolution. We present the main results and key findings here, while additional details and qualitative results can be found in the Appendix.

Image to text (captioning). We first consider the task of image captioning. To achieve this, we train a new Text Variational Encoder on the captioning dataset to extract text latents from text tokens, and a separate text decoder with a reconstruction loss to convert text latents back into tokens. CrossFlow is then trained to map from the image latent space to the text latent space. Following previous work, we use the Karpathy split [33] of COCO dataset [43] for training and testing. In addition, we can also leverage the bi-directional flow property, and simply fine-tune our text-to-image CrossFlow model on COCO and use its inversion for captioning. We report results in Tab. 4. CrossFlow enables direct evolution from image space to text space for image captioning, achieving state-of-the-art performance.

Image to depth (depth estimation). For monocular depth estimation, we train CrossFlow in pixel space. Specifically, we use a recontruction loss to train the Image Variational Encoder to map the original image into the shape of a depth map, followed by the flow matching model which generates the final depth maps. We train and evaluate our model on KITTI [21] (Eigen split [17]) and NYUv2 [69] (official split) for outdoor and indoor scenarios, respectively. As shown in Tab. 5, our model achieves comparable performance to state-of-the-art methods on both datasets. Notably, DiffusionDepth [15] utilizes Swin Transformer [52] and specific designs such as Multi-Scale Aggregation and Monocular Conditioned Denoising Block. In contrast, our model achieves similar performance without any additional enhancements, demonstrating the efficiency and effectiveness of CrossFlow in mapping from images to depth.

Method	KITTI		NYUv2	
	AbsRel (↓)	δ_1 (↑)	AbsRel (↓)	δ_1 (↑)
TransDepth [78]	0.064	0.956	0.106	0.900
AdaBins [5]	0.058	0.964	0.103	0.903
DepthFormer [40]	0.052	0.975	0.096	0.921
BinsFormer [41]	0.052	0.974	0.094	0.925
DiffusionDepth [15]	0.050	0.977	0.085	0.939
CrossFlow (Ours)	0.053	0.973	0.094	0.928

Table 5. **Monocular depth estimation on KITTI and NYUv2.** CrossFlow enables direct mapping from image to depth, achieving comparable performance to state-of-the-art models.

Method	FID ↓	IS ↑	PSNR ↑	SSIM ↑
Reference	1.9	240.8	-	-
Regression	15.2	121.1	27.9	0.801
SR3 [67]	5.2	180.1	26.4	0.762
Flow Matching [44]	3.4	200.8	24.7	0.747
CrossFlow (Ours)	3.0	207.2	25.6	0.764

Table 6. **Image super-resolution on the ImageNet validation set.** Our direct mapping method achieves better performance.

Low-resolution to high-resolution (super-resolution). We compare CrossFlow with the standard flow-matching super-resolution method, which upsamples the low-resolution image, concatenates it with input noise as conditioning, and then processes it through the neural network. In contrast, we directly evolve the upsampled low-resolution image into a high-resolution image, without additional concatenation conditioning. We also compare against SR3 [67] which uses diffusion models for super-resolution. Following previous work [44, 67], we train and evaluate our model on ImageNet [13] for $64 \times 64 \rightarrow 256 \times 256$ super-resolution, and provide results in Tab. 6. Our method achieves better results compared to the standard flow matching and SR3, indicating that CrossFlow can also effectively evolve between similar distributions while achieving superior performance.

6. Conclusion

In this paper, we proposed CrossFlow, a simple and general framework for cross-modal flow matching that works well across a variety of tasks without requiring task specific architectural modifications. It outperforms conventional flow matching, while also enabling new capabilities such as latent arithmetic. We showcase that CrossFlow is a promising approach for the future thanks to its better scalablity. We hope our approach helps pave the way towards further research and applications of cross-modal flow matching.

Acknowledgements. We sincerely appreciate Ricky Chen and Saketh Rambhatla for their valuable discussions. AY acknowledges support from the ONR N000142412696.

References

[1] Michael S Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *ICLR*, 2023. 2, 3

[2] Michael S Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *arXiv preprint arXiv:2303.08797*, 2023. 6

[3] Michael S Albergo, Mark Goldstein, Nicholas M Boffi, Rajesh Ranganath, and Eric Vanden-Eijnden. Stochastic interpolants with data-dependent couplings. *arXiv preprint arXiv:2310.03725*, 2023. 3

[4] Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All are worth words: A vit backbone for diffusion models. In *CVPR*, 2023. 3

[5] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Adabins: Depth estimation using adaptive bins. In *CVPR*, 2021. 3, 8

[6] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 2, 3

[7] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *CVPR*, 2023. 2, 3

[8] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. Pixart-α : Fast training of diffusion transformer for photorealistic text-to-image synthesis. *arXiv preprint arXiv:2310.00426*, 2023. 2, 3, 6

[9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 5

[10] Ting Chen, Ruixiang Zhang, and Geoffrey Hinton. Analog bits: Generating discrete data using diffusion models with self-conditioning. *arXiv preprint arXiv:2208.04202*, 2022. 3

[11] Xiaoliang Dai, Ji Hou, Chih-Yao Ma, Sam Tsai, Jialiang Wang, Rui Wang, Peizhao Zhang, Simon Vandenhende, Xiaofang Wang, Abhimanyu Dubey, et al. Emu: Enhancing image generation models using photogenic needles in a haystack. *arXiv preprint arXiv:2309.15807*, 2023. 3

[12] Valentin De Bortoli, Guan-Horng Liu, Tianrong Chen, Evangelos A Theodorou, and Weilie Nie. Augmented bridge matching. *arXiv preprint arXiv:2311.06978*, 2023. 3

[13] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 8

[14] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. In *NeurIPS*, 2021. 2, 3

[15] Yiqun Duan, Xianda Guo, and Zheng Zhu. Diffusiondepth: Diffusion denoising approach for monocular depth estimation. *arXiv preprint arXiv:2303.05021*, 2023. 3, 8

[16] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. 2, 7

[17] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *NeurIPS*, 2014. 8

[18] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *ICML*, 2024. 2, 3, 5, 6

[19] Johannes S Fischer, Ming Gui, Pingchuan Ma, Nick Stracke, Stefan A Baumann, and Björn Ommer. Boosting latent diffusion with flow matching. *arXiv preprint arXiv:2312.07360*, 2023. 3, 6

[20] Junlong Gao, Xi Meng, Shiqi Wang, Xia Li, Shanshe Wang, Siwei Ma, and Wen Gao. Masked non-autoregressive image captioning. *arXiv preprint arXiv:1906.00717*, 2019. 3, 8

[21] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 2013. 8

[22] Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. In *NeurIPS*, 2024. 5

[23] Rohit Girdhar, Mannat Singh, Andrew Brown, Quentin Duval, Samaneh Azadi, Sai Saketh Rambhatla, Akbar Shah, Xi Yin, Devi Parikh, and Ishan Misra. Emu video: Factorizing text-to-video generation by explicit image conditioning. *arXiv preprint arXiv:2311.10709*, 2023. 3

[24] Longteng Guo, Jing Liu, Xinxin Zhu, Xingjian He, Jie Jiang, and Hanqing Lu. Non-autoregressive image captioning with counterfactuals-critical multi-agent learning. *arXiv preprint arXiv:2005.04690*, 2020. 3, 8

[25] Eric Heitz, Laurent Belcour, and Thomas Chambon. Iterative α-(de) blending: A minimalist deterministic diffusion model. In *SIGGRAPH*, 2023. 2, 3

[26] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. *arXiv preprint arXiv:2104.08718*, 2021. 5

[27] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurIPS*, 2017. 5

[28] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 2, 3, 4

[29] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 2

[30] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 2, 3

[31] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *JMLR*, 2022. 3

[32] Aapo Hyvärinen and Peter Dayan. Estimation of non-normalized statistical models by score matching. *JMLR*, 2005. 2

[33] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015. 8

[34] Tero Karras, Miika Aittala, Tuomas Kynkäänniemi, Jaakko Lehtinen, Timo Aila, and Samuli Laine. Guiding a diffusion model with a bad version of itself. *arXiv preprint arXiv:2406.02507*, 2024. 4, 7

[35] Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *CVPR*, 2024. 3

[36] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 4

[37] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. *arXiv preprint arXiv:2009.09761*, 2020. 2

[38] Jason Lee, Elman Mansimov, and Kyunghyun Cho. Deterministic non-autoregressive neural sequence modeling by iterative refinement. *arXiv preprint arXiv:1802.06901*, 2018. 3, 8

[39] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. 3

[40] Zhenyu Li, Zehui Chen, Xianming Liu, and Junjun Jiang. Depthformer: Exploiting long-range correlation and local information for accurate monocular depth estimation. *Machine Intelligence Research*, 2023. 3, 8

[41] Zhenyu Li, Xuyang Wang, Xianming Liu, and Junjun Jiang. Binsformer: Revisiting adaptive bins for monocular depth estimation. *IEEE Transactions on Image Processing*, 2024. 3, 8

[42] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *CVPR*, 2023. 2, 3

[43] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 5, 8

[44] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *ICLR*, 2022. 2, 3, 5, 8

[45] Guan-Horng Liu, Yaron Lipman, Maximilian Nickel, Brian Karrer, Evangelos A Theodorou, and Ricky TQ Chen. Generalized schr\" odinger bridge matching. *arXiv preprint arXiv:2310.02233*, 2023. 3

[46] Guan-Horng Liu, Arash Vahdat, De-An Huang, Evangelos A Theodorou, Weili Nie, and Anima Anandkumar. Image-to-image schr\" odinger bridge. *arXiv preprint arXiv:2302.05872*, 2023. 3

[47] Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D Plumbley. Audioldm: Text-to-audio generation with latent diffusion models. *arXiv preprint arXiv:2301.12503*, 2023. 2

[48] Minghua Liu, Chao Xu, Haian Jin, Linghao Chen, Mukund Varma T, Zexiang Xu, and Hao Su. One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. *Advances in Neural Information Processing Systems*, 36:22226–22246, 2023. 3

[49] Qihao Liu, Zhanpeng Zeng, Ju He, Qihang Yu, Xiaohui Shen, and Liang-Chieh Chen. Alleviating distortion in image generation via multi-resolution diffusion models. *arXiv preprint arXiv:2406.09416*, 2024. 3, 5

[50] Qihao Liu, Yi Zhang, Song Bai, Adam Kortylewski, and Alan Yuille. Direct-3d: Learning direct text-to-3d generation on massive noisy 3d data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6881–6891, 2024. 3

[51] Xingchao Liu, Chengyue Gong, and qiang liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *ICLR*, 2023. 2, 3

[52] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 8

[53] I Loshchilov. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 5

[54] Jianjie Luo, Yehao Li, Yingwei Pan, Ting Yao, Jianlin Feng, Hongyang Chao, and Tao Mei. Semantic-conditional diffusion networks for image captioning. In *CVPR*, 2023. 3, 8

[55] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 2, 3, 6

[56] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 5

[57] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023. 3, 5

[58] Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024. 2, 3

[59] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 2, 3

[60] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 2, 5, 7

[61] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, 2020. 2, 7

[62] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *ICML*, 2021. 3, 6

[63] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 2, 3, 6

[64] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *TPAMI*, 2020. 3

[65] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2, 3, 5, 6

[66] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. In *NeurIPS*, 2022. 2, 3, 6

[67] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *TPAMI*, 2022. 3, 8

[68] Yuyang Shi, Valentin De Bortoli, Andrew Campbell, and Arnaud Doucet. Diffusion schrödinger bridge matching. In *NeurIPS*, 2024. 3

[69] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 8

[70] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022. 2, 3

[71] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2015. 2

[72] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *NeurIPS*, 2019. 2

[73] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *ICLR*, 2021. 2

[74] Zhicong Tang, Tiankai Hang, Shuyang Gu, Dong Chen, and Baining Guo. Simplified diffusion schr\" odinger bridge. *arXiv preprint arXiv:2403.14623*, 2024. 3

[75] Alexander Tong, Nikolay Malkin, Kilian Fatras, Lazar Atanackovic, Yanlei Zhang, Guillaume Huguet, Guy Wolf, and Yoshua Bengio. Simulation-free schr\" odinger bridges via score and flow matching. *arXiv preprint arXiv:2307.03672*, 2023. 3

[76] A Vaswani. Attention is all you need. In *NeurIPS*, 2017. 5

[77] Zeyue Xue, Guanglu Song, Qiushan Guo, Boxiao Liu, Zhuofan Zong, Yu Liu, and Ping Luo. Raphael: Text-to-image generation via large mixture of diffusion paths. In *NeurIPS*, 2024. 6

[78] Guanglei Yang, Hao Tang, Mingli Ding, Nicu Sebe, and Elisa Ricci. Transformer-based attention networks for continuous pixel-wise prediction. In *ICCV*, 2021. 8

[79] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *CVPR*, 2024. 3

[80] Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. Image captioning with semantic attention. In *CVPR*, 2016. 3

[81] Linqi Zhou, Aaron Lou, Samar Khanna, and Stefano Ermon. Denoising diffusion bridge models. *arXiv preprint arXiv:2309.16948*, 2023. 3

[82] Yuanen Zhou, Yong Zhang, Zhenzhen Hu, and Meng Wang. Semi-autoregressive transformer for image captioning. In *ICCV*, 2021. 3, 8

[83] Yufan Zhou, Bingchen Liu, Yizhe Zhu, Xiao Yang, Changyou Chen, and Jinhui Xu. Shifted diffusion for text-to-image generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10157–10166, 2023. 3