Visual Chain-of-Thought Diffusion Models

William Harvey
University of British Columbia
wsgh@cs.ubc.ca

Frank Wood
University of British Columbia
fwood@cs.ubc.ca

Abstract

Recent progress with conditional image diffusion models has been stunning, and this holds true whether we are speaking about models conditioned on a text description, a scene layout, or a sketch. Unconditional image diffusion models are also improving but lag behind, as do diffusion models which are conditioned on lower-dimensional features like class labels. We propose to close the gap between conditional and unconditional models using a two-stage sampling procedure. In the first stage we sample an embedding describing the semantic content of the image. In the second stage we sample the image conditioned on this embedding and then discard the embedding. Doing so lets us leverage the power of conditional diffusion models on the unconditional generation task, which we show improves FID by $25-50\%$ *compared to standard unconditional generation.*

1. Introduction

Recent text-to-image diffusion generative models (DGMs) have exhibited stunning sample quality [17] to the point that they are now being used to create art [13]. Further work has explored conditioning on scene layouts [22], segmentation masks [7, 22], or the appearance of a particular object [10]. We broadly lump these methods together as "conditional" DGMs to contrast them with "unconditional" image DGMs which sample an image without dependence on text or any other information. Relative to unconditional DGMs, conditional DGMs typically produce more realistic samples [1, 6, 7] and work better with few sampling steps [11]. Furthermore our results suggest that sample realism grows with "how much" information the DGM is conditioned on: as hinted at in Fig. 1 an image is likely to be more realistic if conditioned on being "an aerial photograph of a road between green fields" than if it is if simply conditioned on being "an aerial photograph."

This gap in performance is problematic. In the spirit of this workshop, imagine you have been tasked with sampling a dataset of synthetic aerial photos which will be used to train a computer vision system. A researcher doing so would currently have to either (a) make up a scene description before generating each dataset image, and ensure these cover the entirety of the desired distribution, or (b) accept the inferior image quality gleaned by conditioning just on each image being "an aerial photograph".

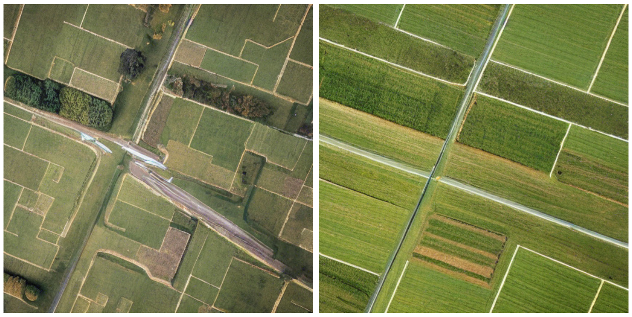

Figure 1. **Left:** Output from Stable Diffusion [16] prompted to produce "aerial photography". **Right:** Using a more detailed prompt[1] with the same random seed removes the "smudged" road artifact that appears on the left. VCDM builds on this observation.

To close this gap, we take inspiration from "chain of thought" reasoning [20] in large language models (LLMs). Consider using an LLM to answer a puzzle: *Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?* If the LLM is prompted to directly state the answer, it must perform all reasoning and computation in a single step. If instead it is prompted to explain its reasoning as it computes an answer, it can first conclude that the answer is given by the expression $5 + 2 \times 3$, and then output an answer *conditioned* on it arising from such an expression. Printing this expression in an intermediate step dramatically improves accuracy [20].

Let us imagine an image generative model along these lines. When prompted to sample "an aerial photograph", it may start by sampling a more detailed description: "an aerial photograph of a patchwork of small green fields

[1] We used the prompt "Aerial photography of a patchwork of small green fields separated by brown dirt tracks between them. A large tarmac road passes through the scene from left to right."

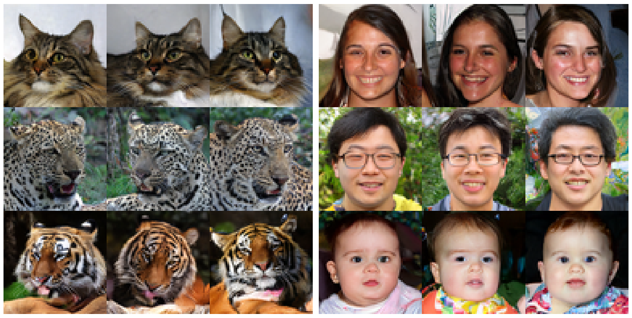

Figure 2. CLIP-conditional samples on AFHQ and FFHQ. Each row shows three samples conditioned on the same CLIP embedding.

[...]". Given this detailed description, it can leverage the full power of a conditional DGM to generate a high-quality image. Our approach follows these lines but, instead of operating on language, our intermediate space consists of a semantically-meaningful embedding from a pretrained CLIP embedder [14]. Specifically we train a DGM to model the distribution of CLIP embeddings of images in our dataset. From this we achieve improved unconditional image generation by first sampling a CLIP embedding and then feeding this CLIP embedding into a conditional image DGM. Note that, while this technique is related to text-conditional image generation, we are instead applying it to improved *unconditional* image generation. We call the resulting model a Visual Chain-of-Thought Diffusion Model (VCDM).

2. Background

Conditional DGMs We provide a high-level overview of conditional DGMs that is sufficient to understand our contributions, referring to Karras et al. [8] for a more complete description and derivation. A conditional image DGM [19] samples an image $\mathbf{x}$ given a conditioning input $\mathbf{y}$, where $\mathbf{y}$ can be, for example, a class label, a text description, or both of these in a tuple. We can recover an unconditional DGM by setting $\mathbf{y}$ to a null variable in the below. Given a dataset of $(\mathbf{x}, \mathbf{y})$ pairs sampled from $p_{\text{data}}(\cdot, \cdot)$, a conditional DGM $p_\theta(\mathbf{x}|\mathbf{y})$ is fit to approximate $p_{\text{data}}(\mathbf{x}|\mathbf{y})$. It is parameterized by a neural network $\hat{\mathbf{x}}_\theta(\cdot)$ trained to optimize

$$\mathbb{E}_{u(\sigma)p_\sigma(\mathbf{x}_\sigma|\mathbf{x},\sigma)p_{\text{data}}(\mathbf{x},\mathbf{y})}\left[\lambda(\sigma)||\mathbf{x} - \hat{\mathbf{x}}_\theta(\mathbf{x}_\sigma, \mathbf{y}, \sigma)||^2\right] \quad (1)$$

where $\mathbf{x}_\sigma \sim p_\sigma(\cdot|\mathbf{x}, \sigma)$ is a copy of $\mathbf{x}$ corrupted by Gaussian noise with standard deviation σ; $u(\sigma)$ is a broad distribution over noise standard deviations; and $\lambda(\sigma)$ is a weighting function. During inference, samples from $p_\theta(\mathbf{x}|\mathbf{y})$ are drawn via a stochastic differential equation with dynamics dependent on $\hat{\mathbf{x}}_\theta(\cdot)$.

CLIP embeddings CLIP (contrastive language-image pretraining) [14] consists of two neural networks, an image embedder $e_i(\cdot)$ and a text embedder $e_t(\cdot)$, trained on a large captioned-image dataset. Given an image $\mathbf{x}$ and a caption $\mathbf{y}$, the training objective encourages the cosine similarity between $e_i(\mathbf{x})$ and $e_t(\mathbf{y})$ to be large if $\mathbf{x}$ and $\mathbf{y}$ are a matching image-caption pair and small if not. The image embedder therefore learns to map from an image to a semantically-meaningful embedding capturing any features that may be included in a caption. We use a CLIP image embedder with the ViT-B/32 architecture and weights released by Radford et al. [14]. We can visualize the information captured by the CLIP embedding by showing the distribution of images produced by our conditional DGM given a single CLIP embedding; see Fig. 2.

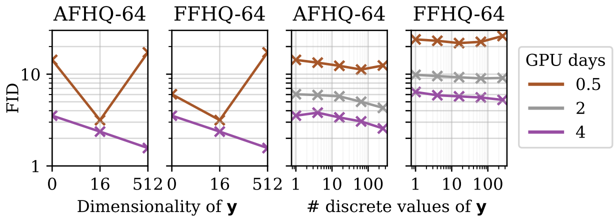

Figure 3. FID versus dimensionality of $\mathbf{y}$ on AFHQ [2] and FFHQ [9]. With small training budgets (brown line), it is harmful when $\mathbf{y}$ is too informative. With larger training budgets (purple line), it is helpful to make $\mathbf{y}$ much more high dimensional.

3. Conditional vs. unconditional DGMs

What does it mean to say that conditional DGMs beat unconditional DGMs? A standard procedure to evaluate unconditional DGMs is to start by sampling a set of N images independently from the model: $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)} \sim p_\theta(\cdot)$. We can then compute the Fréchet Inception distance (FID) [5] between this set and the dataset. If the generative model matches the data distribution well, the FID will be low. For conditional DGMs the standard procedure has one extra step: we first independently sample $\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(N)} \sim p_{\text{data}}(\cdot)$. We then sample each image given the corresponding $\mathbf{y}^{(i)}$ as $\mathbf{x}^{(i)} \sim p_\theta(\cdot|\mathbf{y}^{(i)})$. Then, as in the unconditional case, we compute the FID between the set of images $\mathbf{x}_1, \ldots, \mathbf{x}_N$ and the dataset, without reference to $\mathbf{y}_1, \ldots, \mathbf{y}_N$. Even though it does not measure alignment between $\mathbf{x}, \mathbf{y}$ pairs, conditional DGMs beat comparable unconditional DGMs on this metric in many settings: class-conditional CIFAR-10 generation [8], segmentation-conditional generation [7], or bounding box-conditional generation [7].

Why do conditional DGMs beat unconditional DGMs? Conditional DGMS "see" more data during training than their unconditional counterparts because updates involves $\mathbf{y}$ as well as $\mathbf{x}$. Bao et al. [1], Hu et al. [7] prove that this is not the sole reason for their successes because the effect holds

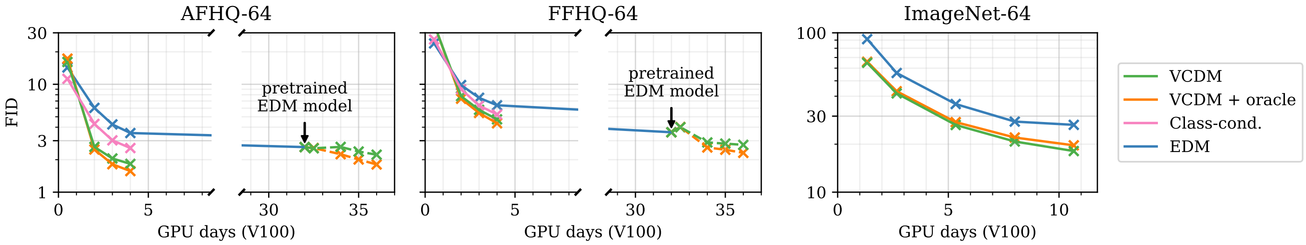

Figure 4. FID throughout training. We show results for each method trained from scratch and, on AFHQ and FFHQ, for finetuning a pretrained EDM model (which was trained for the equivalent of 32 GPU days). VCDM quickly outperforms EDM when trained from scratch and quickly improves on the pretrained model when used for finetuning.

up even when $\mathbf{y}$ is derived from an unconditional dataset through self-supervised learning. To our knowledge, the best explanation for their success is, as stated by Bao et al. [1], that conditional distributions typically have "fewer modes and [are] easier to fit than the original data distribution."

When do conditional DGMs beat unconditional DGMS? We present results in Fig. 3 to answer this question. We show FID scores for conditional DGMs trained to condition on embeddings of varying information content. We produce $\mathbf{y}$ by starting from the CLIP embedding of each image in our dataset and using either principal component analysis to reduce their dimensionality (left two panels) or K-means clustering to discretize them (right two panels) [7]. We see that, given a small training budget, it is best to condition on little information. With a larger training budget, performance appears to improve steadily as the dimensionality of $\mathbf{y}$ is expanded. We hypothesize that **(1)** conditioning on higher-dimensional $\mathbf{y}$ slows down training because it means that points close to any given value of $\mathbf{y}$ will be seen less frequently and **(2)** with a large enough compute budget, any $\mathbf{y}$ correlated with $\mathbf{x}$ will be useful to condition on. This suggests that, as compute budgets grow, making unconditional DGM performance match conditional DGM performance will be increasingly useful.

4. Method

We have established that conditioning on CLIP embeddings improves DGMs. We now introduce VCDM which leverages this phenomenon to benefit the unconditional setting (in which the user does not wish to specify any input to condition on) and the "lightly-conditional" setting in which the input is low-dimensional, e.g. a class-label. We will denote any such additional input $\mathbf{a}$ (letting $\mathbf{a}$ be a null variable in the unconditional setting) and from now on always use $\mathbf{y} := e_i(\mathbf{x})$ to refer to a CLIP embedding. VCDM approximates the target distribution $p_{\text{data}}(\mathbf{x}|\mathbf{a})$ as

$$
\begin{aligned}
p_{\text{data}}(\mathbf{x}|\mathbf{a}) &= \mathbb{E}_{p_{\text{data}}(\mathbf{y}|\mathbf{a})}\left[p_{\text{data}}(\mathbf{x}|\mathbf{y},\mathbf{a})\right] && (2)\\
&\approx \mathbb{E}_{p_{\phi}(\mathbf{y}|\mathbf{a})}\left[p_{\theta}(\mathbf{x}|\mathbf{y},\mathbf{a})\right] && (3)
\end{aligned}
$$

where $p_\phi(\mathbf{y}|\mathbf{a})$ is a second DGM modeling the CLIP embeddings. We can sample from this distribution by sampling $\mathbf{y} \sim p_\phi(\cdot|\mathbf{a})$ and then leveraging the conditional image DGM to sample $\mathbf{x} \sim p_\theta(\cdot|\mathbf{y},\mathbf{a})$ before discarding $\mathbf{y}$. From now on we will call $p_\theta(\mathbf{x}|\mathbf{y},\mathbf{a})$ the *conditional image model* and $p_\phi(\mathbf{y}|\mathbf{a})$ the *auxiliary model*. In our experiments the auxiliary model uses a small architecture relative to the conditional image model and so adds little extra cost.[2]

Auxiliary model Our auxiliary model is a conditional DGM targeting $p_{\text{data}}(\mathbf{y}|\mathbf{a})$, where $\mathbf{y}$ is a 512-dimensional CLIP embedding. We follow the architectural choice of Ramesh et al. [15] and use a DGM with a transformer architecture. It takes as input a series of 512-dimensional input tokens: an embedding of σ; an embedding of $\mathbf{a}$ if this is not null; an embedding of $\mathbf{a}_\sigma$; and a learned query. These are passed through six transformer layers and then the output corresponding to the learned query token is used as the output. Like Ramesh et al. [15], we parameterize the DGM to output an estimate of the denoised $\mathbf{a}$ instead of estimating the added noise as is more common in the diffusion literature. On AFHQ and FFHQ we find that data augmentation is helpful to prevent the auxiliary model overfitting. We perform augmentations (including rotation, flipping and color jitter) in image space and feed the augmented image through $e_i(\cdot)$ to obtain an augmented CLIP embedding. Following Karras et al. [8], we pass a label describing the augmentation into the transformer as an additional input token so that we can condition on there being no augmentation at test-time.

Conditional image model Our diffusion process hyperparameters and samplers build on those of Karras et al. [8]. For AFHQ and FFHQ, we use the U-Net architecture originally proposed by Song et al. [18]. For ImageNet, we use the slightly larger U-Net architecture proposed by Dhariwal and Nichol [4]. We match the data augmentation scheme to be the same as that of Karras et al. [8] on each dataset. There are

[2] For our ImageNet experiments, sampling from our auxiliary model takes 35ms per batch item. Sampling from our image model takes 862ms and so VCDM has inference time only 4% greater than our baselines.

established conditional variants of both architectures [4, 8], which incorporate $\mathbf{y}$ via a learned linear projection that is added to the embedding of the noise standard deviation σ. Our conditional image model needs to additionally incorporate $\mathbf{a}$; we can do so by simply concatenating it to $\mathbf{y}$ and learning a projection for the resulting vector.

5. Experiments

We experiment on three datasets: AFHQ [2], FFHQ [9] and ImageNet [3], all at 64×64 resolution. We target unconditional generation for AFHQ and FFHQ, and class-conditional generation for ImageNet. As well as training networks from scratch on each dataset, we report results with the model checkpoints released by Karras et al. [8] on AFHQ and FFHQ, which we finetune to be conditional on the CLIP embedding. To do so, we simply add a learnable linear projection of the CLIP embedding and initialize its weights to zero. Figure 4 reports the FID on each setting and dataset throughout the training of the conditional image model.[3] In each case, the auxiliary model is trained for one day on one V100 GPU. We compare VCDM to three other approaches: **EDM** [9] is a standard DGM directly modeling $p_{\text{data}}(\mathbf{x}|\mathbf{a})$. **VCDM with oracle** uses our conditional image model but uses the ground-truth $\mathbf{y}$ for each test $\mathbf{a}$ instead of sampling from the learned auxiliary model, i.e. it is the performance that VCDM would achieve with a perfect auxiliary model. **Class-cond** is an ablation of VCDM that applies to unconditional tasks where $\mathbf{a}$ is null. It uses discrete $\mathbf{y}$ (as on the right of Fig. 3) so that $p_{\text{data}}(\mathbf{y}|\mathbf{a}) = p_{\text{data}}(\mathbf{y})$ is a simple categorical distribution which we can sample from exactly, but we see that it is outperformed by VCDM.

VCDM consistently outperforms unconditional generation after 1-2 GPU-days and this performance gap continues for as long as we train the networks. Comparing VCDM's performance with and without the oracle we see that they are close. For networks trained from scratch we show in Sec. 5 that VCDM always has an improvement over EDM at least 80% as large as that of VCDM with an oracle, indicating that $p_\phi(\mathbf{y}|\mathbf{a})$ is a good approximation of $p_{\text{data}}(\mathbf{y}|\mathbf{a})$. We can therefore leverage almost the full power of conditional DGMs for unconditional sampling.

6. Related work

Several existing image generative models leverage CLIP embeddings for better text-conditional generation [12, 15]. We differ by suggesting that CLIP embeddings are not only useful for text-conditioning, but also as a general tool to improve the realism of generated images. We demonstrate this for unconditional and class-conditional generation. Our work takes inspiration from Weilbach et al. [21], who show improved performance in various approximate inference settings by modeling problem-specific auxiliary variables (like $\mathbf{y}$) in addition to the variables of interest ($\mathbf{x}$) and observed variables ($\mathbf{a}$). We apply these techniques to the image domain and incorporate pretrained CLIP embedders to obtain auxiliary variables. VCDM also relates to methods which perform diffusion in a learned latent space [16]: our auxiliary model $p_\phi(\mathbf{y}|\mathbf{a})$ is analogous to a "prior" in a latent space and our conditional image model $p_\theta(\mathbf{x}|\mathbf{a}, \mathbf{y})$ to a "decoder" Such methods typically use a near-deterministic decoder and so their latent variables must summarize all information about the image. Our conditional DGM decoder on the other hand will function reasonably however little information is stored in $\mathbf{y}$ and so VCDM provides an additional degree of freedom in terms of what to store. This is an interesting design space for future exploration. Classifier [18] and classifier-free guidance[6] are two alternative methods for conditional sampling from DGMs. Both have a "guidance strength" hyperparameter to trade fidelity to $p_{\text{data}}(\mathbf{x}|\mathbf{y})$ against measures of alignment between $\mathbf{x}$ and $\mathbf{y}$. A possible extension to VCDM could parameterize $p_\theta(\mathbf{x}|\mathbf{y}, \mathbf{a})$ with either of them.

Table 1. Final FID score for the models we train from scratch and a comparison of their improvements over EDM.

Dataset	AFHQ	FFHQ	ImageNet
$\mathbf{y}$	null	null	class label
VCDM	1.83	4.73	18.1
VCDM + oracle	1.57	4.35	19.7
Class-cond.	2.56	5.24	-
EDM	3.53	6.39	26.5
Improv. w/ VCDM	48.2%	26.0%	31.5%
Improv. w/ oracle	55.6%	31.9%	25.6%
$\frac{\text{Improv. w/ VCDM}}{\text{Improv. w/ oracle}}$	86.6%	81.3%	123%

7. Discussion and conclusion

We have presented VCDM, a method for unconditional or lightly-conditional image generation which harnesses the impressive performance of conditional DGMs. A massive unexplored design space remains: there are almost certainly more useful quantities that we could condition on than CLIP embeddings. It may also help to condition on multiple quantities, or "chain" a series of conditional DGMs together. An alternative direction is to simplify VCDM's architecture by, for example, learning a single diffusion model over the joint space of $\mathbf{x}$ and $\mathbf{y}$ instead of generating them sequentially. A drawback of VCDM is that it relies on the availability of a pretrained CLIP embedder. While this is freely available for natural images, it could be a barrier to other applications; an alternative would be to explore the self-supervised representations used by Bao et al. [1], Hu et al. [7].

[3] Each FID is estimated using 20 000 images, each sampled with the SDE solver proposed by Karras et al. [8] using 40 steps, $S_{\text{churn}} = 50$, $S_{\text{noise}} = 1.007$, and other parameters set to their default values.

References

[1] Fan Bao, Chongxuan Li, Jiacheng Sun, and Jun Zhu. Why are conditional generative models better than unconditional ones? *arXiv preprint arXiv:2212.00362*, 2022. 1, 2, 3, 4

[2] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8188–8197, 2020. 2, 4

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 4

[4] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 3, 4

[5] 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. *Advances in neural information processing systems*, 30, 2017. 2

[6] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 1, 4

[7] Vincent Tao Hu, David W Zhang, Yuki M Asano, Gertjan J Burghouts, and Cees GM Snoek. Self-guided diffusion models. *arXiv preprint arXiv:2210.06462*, 2022. 1, 2, 3, 4

[8] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *arXiv preprint arXiv:2206.00364*, 2022. 2, 3, 4

[9] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. arxiv e-prints, page. *arXiv preprint arXiv:1812.04948*, 1, 2018. 2, 4

[10] Yiyang Ma, Huan Yang, Wenjing Wang, Jianlong Fu, and Jiaying Liu. Unified multi-modal latent diffusion for joint subject and text conditional image generation. *arXiv preprint arXiv:2303.09319*, 2023. 1

[11] Chenlin Meng, Ruiqi Gao, Diederik P Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. *arXiv preprint arXiv:2210.03142*, 2022. 1

[12] 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. 4

[13] Jonas Oppenlaender. The creativity of text-based generative art. *arXiv preprint arXiv:2206.02904*, 2022. 1

[14] 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 *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2

[15] 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. 3, 4

[16] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn 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. 1, 4

[17] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. 1

[18] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 3, 4

[19] Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. Csdi: Conditional score-based diffusion models for probabilistic time series imputation. *Advances in Neural Information Processing Systems*, 34:24804–24816, 2021. 2

[20] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*, 2022. 1

[21] Christian Weilbach, William Harvey, and Frank Wood. Graphically structured diffusion models. *arXiv preprint arXiv:2210.11633*, 2022. 4

[22] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*, 2023. 1