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

Supplementary Material

Appendix

In the appendix, we provide additional information as listed below:

A. Method Details

A.1. Loss Function for T2I Generation

We jointly train the Text Variational Encoder with the flow matching model using the following training objective:

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

where λ is the weight of KL-divergence loss. For the flow matching loss L_{FM}, we follow previous work [17] and compute the MSE loss between the predicted velocity $v_\theta(z_t, t)$ at time-step t and the ground-truth velocity $\hat{v}$. To train the Text Variational Encoder, we adopt a CLIP contrastive loss. Specifically, given a batch of N text and image pairs, we use our Text Variational Encoder to obtain text latents z_0, and an image encoder to extract image features $\hat{z}$. Then, we compute the cosine similarity between all pairs of z_0 and $\hat{z}$ in the batch, resulting in a similarity matrix S, where each element s_{ij} represents the cosine similarity between the i^{th} z_0 and j^{th} $\hat{z}$. The similarity scores are then scaled by a temperature parameter τ (a learnable parameter), denoted as logits$_{ij} = s_{ij}/\tau$. After that, a symmetric cross-entropy loss over the similarity scores is computed:

$$L_{\text{I2T}} = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{\exp(\text{logits}_{ii})}{\sum_{j=1}^{N} \exp(\text{logits}_{ij})} \tag{2}$$

$$L_{\text{T2I}} = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{\exp(\text{logits}_{ii})}{\sum_{j=1}^{N} \exp(\text{logits}_{ji})} \tag{3}$$

Finally, we compute the average of these two components to obtain the CLIP loss, which is then used to update our Text Variational Encoder:

$$L_{Enc} = \texttt{CLIP}(z_0, \hat{z}) = \frac{1}{2}(L_{\text{I2T}} + L_{\text{T2I}}) \tag{4}$$

For the KL loss L_{KL}, we adopt the original KL divergence loss [14] with $\lambda = 1 \times 10^{-4}$.

A.2. Experimental Details for Various Tasks

Image captioning. We conduct our experiments on the popular Karpathy split [11] of COCO dataset [16], which contains $113,287$ images for training, $5,000$ images for validation, and $5,000$ image for testing. We train our model with 351M parameters on the training split for 100 epochs, using a batch size of 256 and a base learning rate of 2×10^{-4} with 5 warm-up epochs. Following the standard evaluation setup, we compare the performance over five metrics: BLEU@4 [19] (B@4), METEOR [2] (M), ROUGE [15] (R), CIDEr [30] (C), and SPICE [1] (S).

Monocular depth estimation. We consider KITTI [10] and NYUv2 [28] for outdoor and indoor depth estimation. For KITTI, we use the Eigen split [8], consisting of $23,488$ training images and 697 testing images. For NYUv2, we adopt the official split, which contains $24,231$ training images and 654 testing images. We train our model with 527M parameters on the corresponding training splits for 50 epochs. We use a batch size of 64, and decay the learning rate from 1×10^{-4} to 1×10^{-8} with cosine annealing.

Image super-resolution. We consider natural image super-resolution, training our model on ImageNet 1K [6] for the task of $64 \times 64 \rightarrow 256 \times 256$ super-resolution. We use the dev split for evaluation. During training, we preprocess the images by removing those where the shorter side is less than 256 pixels. The remaining images are then centrally cropped and resized to 256×256. The low-resolution images are then generated by downsampling the 256×256 images using bicubic interpolation with anti-aliasing enabled. For a fair comparison with SR3 [26], we train our CrossFlow with 505M parameters (compared to 625M parameters in SR3). Our model is trained for 1M training steps

Method	**Overall**	**Single Object**	**Two Object**	**Counting**	**Colors**	**Position**	**Attribute binding**
DALL·E 2 [21]	0.52	0.94	0.66	0.49	0.77	0.10	0.19
LDMv1.5 [25]	0.43	0.97	0.38	0.35	0.76	0.04	0.06
LDMv2.1 [25]	0.50	0.98	0.51	0.44	0.85	0.07	0.17
LDM-XL [20]	0.55	0.98	0.74	0.39	0.85	0.15	0.23
PixArt-α [4]	0.48	0.98	0.50	0.44	0.80	0.08	0.07
LDMv3 (512^2) [9]	0.68	0.98	0.84	0.66	0.74	0.40	0.43
CrossFlow	0.55	0.98	0.72	0.39	0.82	0.18	0.21

Table 1. **GenEval comparisons.** Our model achieves comparable performance to state-of-the-art models such as LDM-XL and DALL·E 2, suggesting that CrossFlow is a simple and promising direction for state-of-the-art media generation.

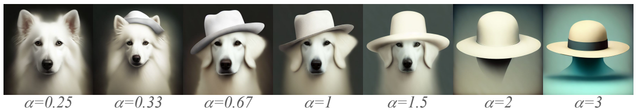


Figure 1. **Arithmetic operation with different scaling terms.** We show images generated by : VE(‘a white dog’)$+\alpha$VE(‘a hat’)

Arithmetic Operation	**Success Rate (%)**
Addition	95.3
Subtraction	92.7
Combination	87.5
Overall	91.4

Table 2. **Success rate of arithmetic operation** We select 1,000 prompts from COCO-val to evaluate the success rate of arithmetic operations. The detection model is used to determine whether the target objects have been successfully added or removed. “Combination” refers to multiple operations involving a combination of both “addition” and “subtraction”.

with a batch size of 512 and a learning rate of 1×10^{-4}, including $5,000$ warm-up steps.

B. Additional Experimental Results

B.1. GenEval Performance

To compare with recent text-to-image models on GenEval, we report the overall score and task-specific scores in Tab. 1. Our model achieves comparable performance to state-of-the-art models such as LDMv2.1 [25], LDM-XL [20], and DALL·E 2 [21]. This demonstrates that directly evolving from text space to image space with our approach is a simple and effective solution for text-to-image generation, indicating a novel and promising direction for state-of-the-art media generation.

B.2. Analysis of Arithmetic Operations

Our model encodes text into a *continuous* latent space with semantic structure. Prior work, such as word2vec [18], has shown that latent arithmetic can *emerge* without explicit training. Arithmetic on these latents effectively retains added and removes subtracted textual information, which our Flow Matching then correspondingly maps to images. We analyze latent arithmetic operations in more detail here. First, we consider addition (+) and test different scaling factors (Fig. 1), showing how they control the amount of information added or removed in the generated image.

In addition, we show qualitatively that the arithmetic works well across diverse concepts and multiple operations in Tab. 2. Specifically, we select 1,000 prompts from COCO-val to test arithmetic operations. A detection model confirms that *91.4% of the objects are accurately added or removed* from the generations, providing quantitative evidence of effective feature disentanglement.

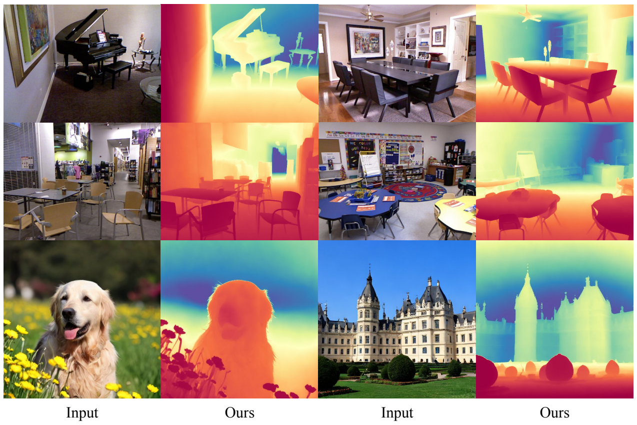


Figure 2. **Qualitative examples for *zero-shot* depth estimation.** The input images in the first two rows are from the NYUv2 dataset, while the input images in the last row were generated by our T2I model. Our model provides robust *zero-shot* depth estimation across domains, whether indoor or outdoor, synthetic or real.

B.3. Zero-shot Depth Estimation

We also evaluate CrossFlow on *zero-shot* depth estimation. Following Marigold [13], we train our model on Hypersim [24] and Virtual KITTI [3], and evaluate our model on 5 real datasets that are not seen during training: KITTI [10], NYUv2 [28], ETH3D [27], ScanNet [5], and DIODE [29]. We follow Marigold [13] to prepare the training and testing data. Our model with 527M parameters is trained for 150K training steps, with a batch size of 512 and a learning rate of 1×10^{-4} with $5,000$ warm-up steps. The results are reported in Tab. 3. Qualitative examples are provided in Fig. 2. Without specific design, CrossFlow achieves comparable or even superior performance compared to state-of-the-art methods, demonstrating the general-purpose nature of our approach on various cross-modal tasks.

B.4. Image Super-resolution

We provide qualitative examples for image super-resolution in Fig. 3. Unlike traditional methods, which typically evolve from Gaussian noise and rely on concatenating up-

Method	# Training samples	KITTI		NYUv2		ETH3D		ScanNet		DIODE	
		AbsRel ↓	**δ_1 ↑**	**AbsRel ↓**	**δ_1 ↑**	**AbsRel ↓**	**δ_1 ↑**	**AbsRel ↓**	**δ_1 ↑**	**AbsRel ↓**	**δ_1 ↑**
DiverseDepth [31]	320K	0.117	0.875	0.190	0.704	0.228	0.694	0.109	0.882	0.376	0.631
MiDaS [22]	2M	0.111	0.885	0.236	0.630	0.184	0.752	0.121	0.846	0.332	0.715
LeReS [32]	300K + 54K	0.090	0.916	0.149	0.784	0.171	0.777	0.091	0.917	0.271	0.766
Omnidata [7]	11.9M + 310K	0.074	0.945	0.149	0.835	0.166	0.778	*0.075*	0.936	0.339	0.742
HDN [33]	300K	*0.069*	*0.948*	0.115	0.867	0.121	0.833	0.080	*0.939*	0.246	**0.780**
DPT [23]	1.2M + 188K	0.098	0.903	**0.100**	*0.901*	0.078	0.946	0.082	0.934	**0.182**	0.758
Marigold [13]	74K	**0.060**	**0.959**	*0.105*	0.904	**0.071**	**0.951**	0.069	**0.945**	0.310	0.772
CrossFlow (Ours)	74K	0.062	0.956	0.103	**0.908**	*0.085*	*0.944*	**0.068**	0.942	*0.270*	*0.768*

Table 3. ***Zero-shot* depth estimation.** Baseline results are reported by Marigold [13]. We follow Marigold and train our CrossFlow on the same datasets, *i.e*., Hypersim [24] and Virtual KITTI [3]. We highlight the **best**, second best, and *third best* entries. With just a unified framework, CrossFlow achieves comparable or even superior performance on complex *zero-shot* depth estimation, demonstrating the general-purpose nature of CrossFlow on various cross-modal tasks.

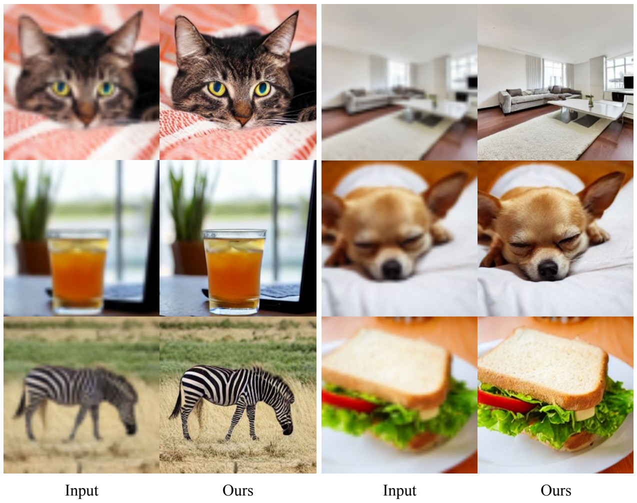


Figure 3. **Qualitative examples for image super-resolution.**

Text encoder	**Recon. accuracy (%)**
Text Encoder (1×1024)	95.12
Text Variational Encoder (1×1024)	94.53

Table 4. **Ablation on text compression.** Both text encoder and Text Variational Encoder preserve most of the input information, despite the large compression ratio ($77 \times 768 \rightarrow 1 \times 1024$, 14.4×).

sampled low-resolution images as conditioning, our approach takes a more direct route: we demonstrate that it is possible to evolve a low-resolution image directly into a high-resolution image, eliminating the need for additional concatenation conditioning.

B.5. Ablation Study

Text compression. In this section, we show that we can compress the input text embedding $x \in \mathbb{R}^{n \times d}$ into $z_0 \in \mathbb{R}^{h \times w \times c}$ (*e.g*., 77×768 CLIP tokens to $4 \times 32 \times 32$ latents for 256px generation, 14.4× compression) with a standard encoder or the proposed Variational Encoder while preserve most of the input information. We report the per-token reconstruction accuracy, computed by cosine similarity, in Tab. 4. The results show that both methods are effective at preserving the input information, achieving high reconstruction accuracy despite a large compression ratio.

CFG indicator. In Fig. 4, we study the effect of our CFG with indicator, and then compare our approach with Autoguidance [12]. The left two columns show the images generated when the indicator $1_c = 0$ (for unconditional generation) and $1_c = 1$ (for conditional generation). It shows that despite generating an image by directly evolving from the text space into the image space without explicit conditioning, our model can still perform unconditional generation with the help of the indicator. This allows our model to support standard CFG. Then, in the middle five columns, we show the images generated with different CFG scaling factors. Similar to the standard flow matching model, the CFG can significantly improve the image quality. Finally, in the last two columns, we compare our CFG with indicator to Autoguidance, using the same scaling factor. Like our approach, Autoguidance also enables low-temperature sampling for models without explicit conditioning. We observe that our CFG with indicator produces higher-fidelity images compared to Autoguidance.

C. Additional Qualitative Examples

We provide additional qualitative examples for text-to-image generation here. Specifically, we first provide 512×512 images generated by our CrossFlow in Fig. 5. Next, we provide more examples for linear interpolation in latent space (Fig. 6 and Fig. 7) and arithmetic operation in latent space (Fig. 8).

References

[1] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Spice: Semantic propositional image caption evaluation. In *ECCV*, 2016. 1

[2] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with hu-

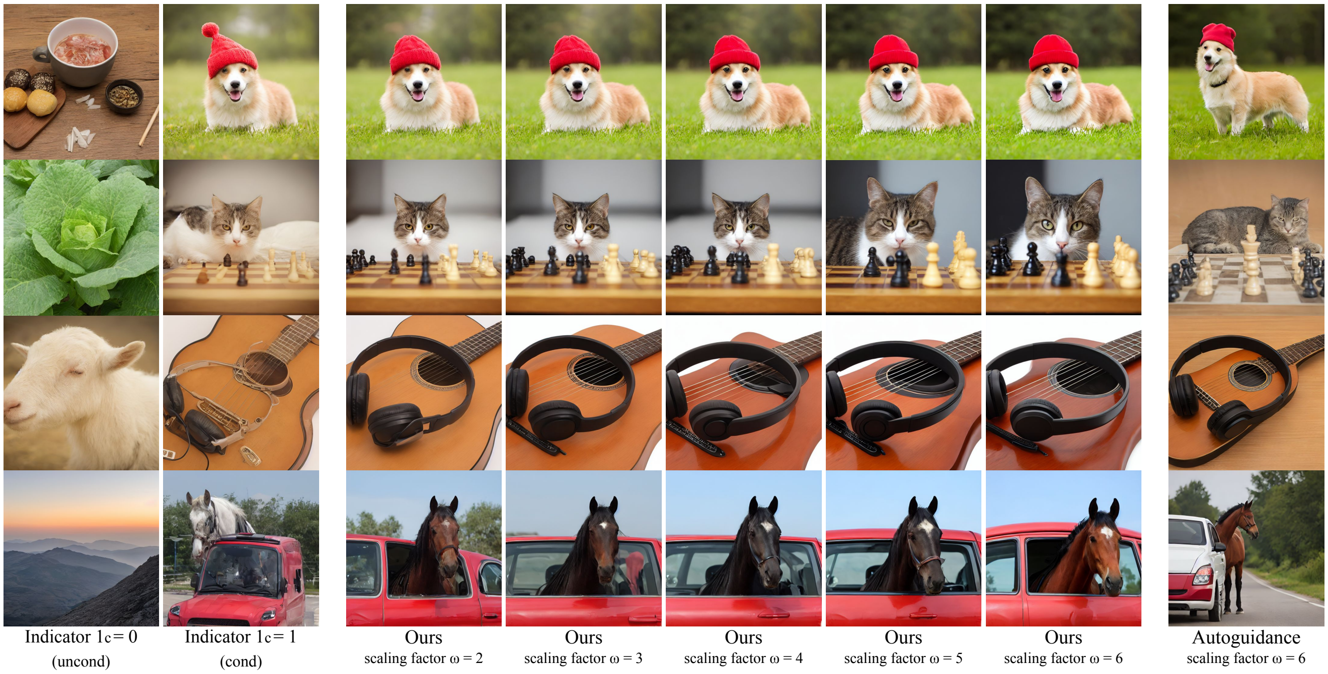


Figure 4. **Ablation on CFG with indicator.** The first two columns show the images generated when the indicator $1_c = 0$ (for unconditional generation) and $1_c = 1$ (for conditional generation), demonstrating that CrossFlow can still perform unconditional generation with the help of the indicator, thereby allowing for the use of standard CFG. We then demonstrate the improvement provided by CFG (middle five columns) and compare it with Autoguidance (last two columns). Prompts used to generate the images: *'a corgi wearing a red hat in the park', 'a cat playing chess', 'a pair of headphones on a guitar', 'a horse in a red car'*

man judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, 2005. 1

[3] Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2. *arXiv preprint arXiv:2001.10773*, 2020. 2, 3

[4] 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

[5] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 2

[6] 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. 1

[7] Ainaz Eftekhar, Alexander Sax, Jitendra Malik, and Amir Zamir. Omnidata: A scalable pipeline for making multi-task mid-level vision datasets from 3d scans. In *ICCV*, 2021. 3

[8] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *NeurIPS*, 2014. 1

[9] 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

[10] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 2013. 1, 2

[11] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015. 1

[12] 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. 3

[13] 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. 2, 3

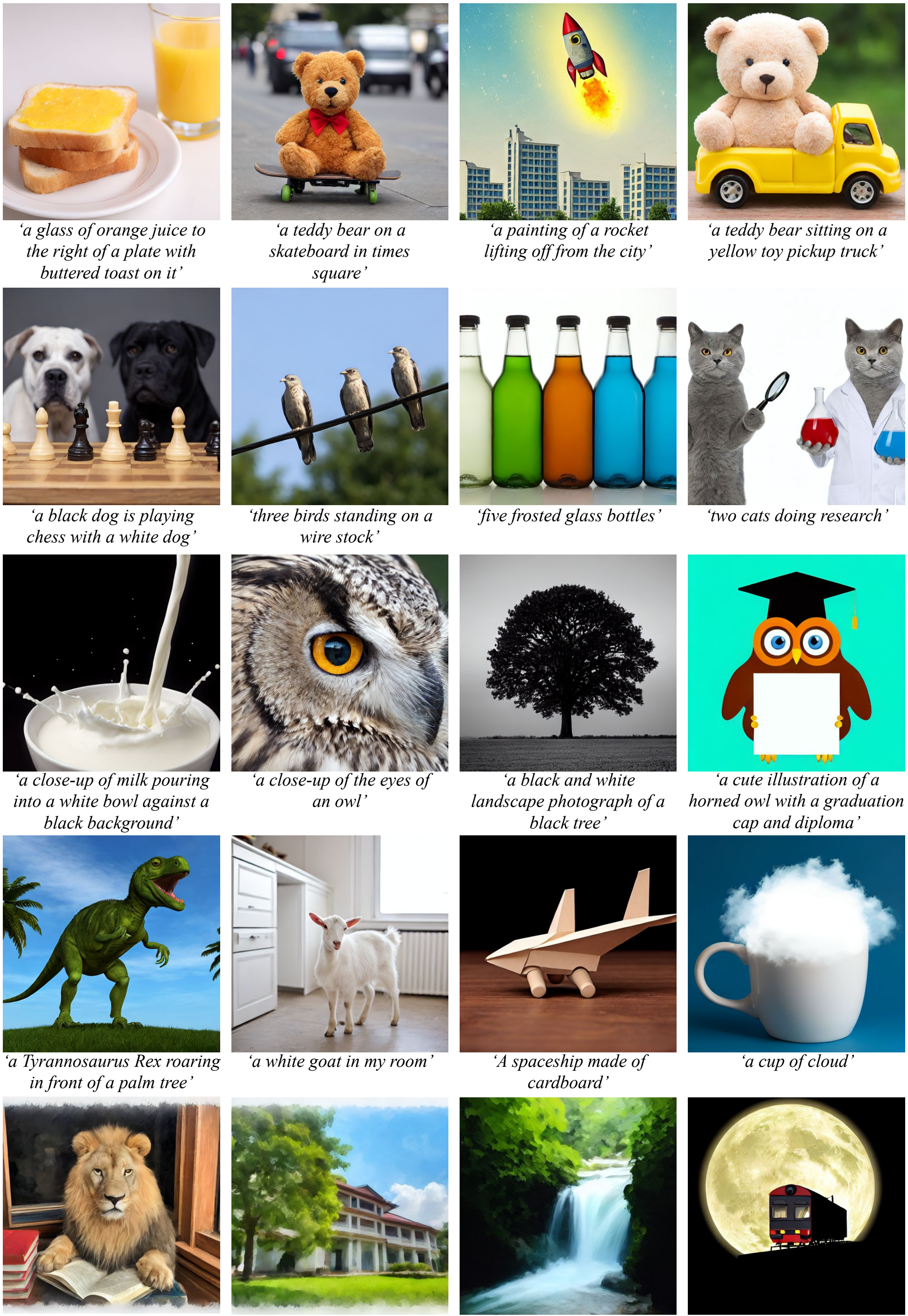
[14] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 1951. 1

[15] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, 2004. 1

[16] 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. 1

[17] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *ICLR*, 2022. 1

[18] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013. 2



'a glass of orange juice to the right of a plate with buttered toast on it'

'a teddy bear on a skateboard in times square'

'a painting of a rocket lifting off from the city'

'a teddy bear sitting on a yellow toy pickup truck'

'a black dog is playing chess with a white dog'

'three birds standing on a wire stock'

'five frosted glass bottles'

'two cats doing research'

'a close-up of milk pouring into a white bowl against a black background'

'a close-up of the eyes of an owl'

'a black and white landscape photograph of a black tree'

'a cute illustration of a horned owl with a graduation cap and diploma'

'a Tyrannosaurus Rex roaring in front of a palm tree'

'a white goat in my room'

'A spaceship made of cardboard'

'a cup of cloud'

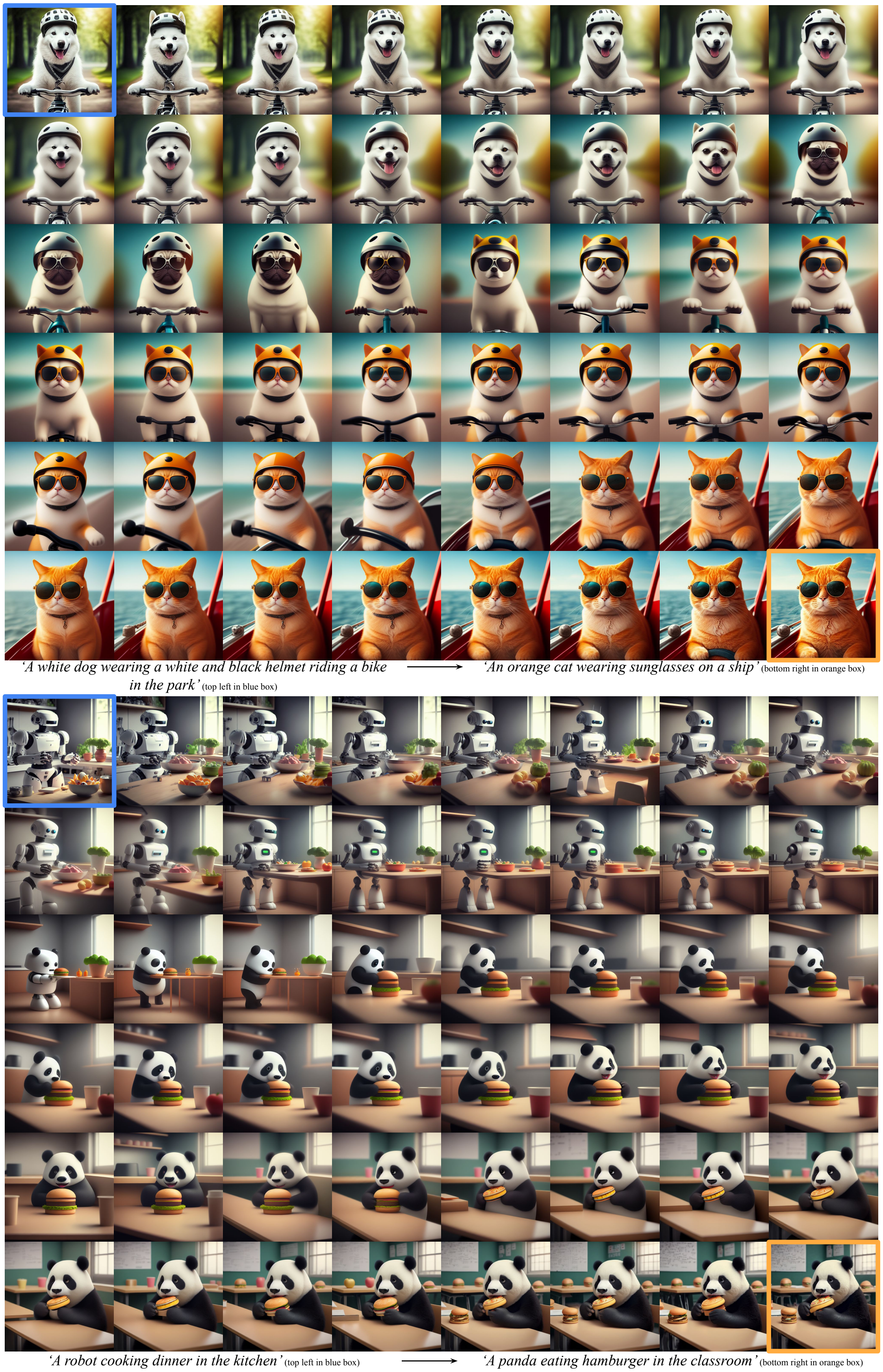
'a impressionistic painting of a lion reading books'

'a watercolor painting of a tree and a building'

'an abstract painting of a waterfall'

'a cartoon of a train going to the moon'

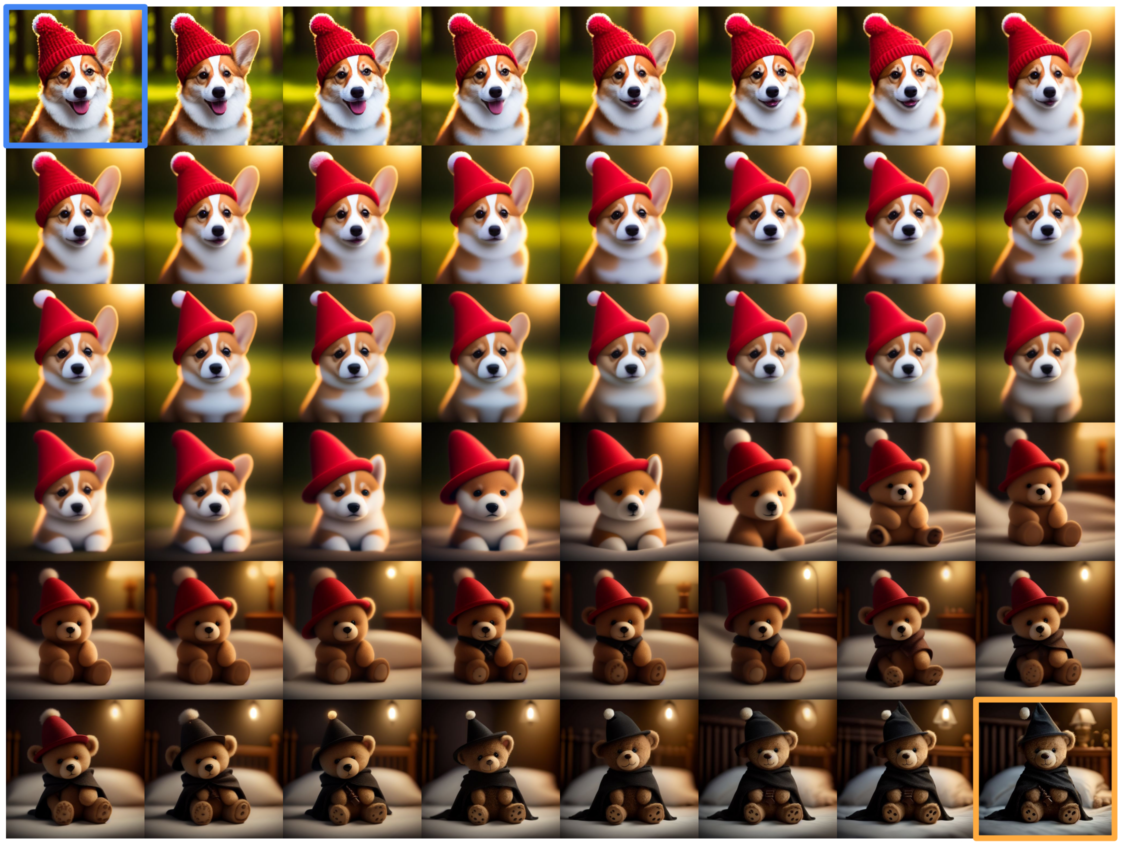
Figure 5. **Qualitative examples for text-to-image with CrossFlow.**



'A white dog wearing a white and black helmet riding a bike in the park' (top left in blue box) ⟶ *'An orange cat wearing sunglasses on a ship'* (bottom right in orange box)

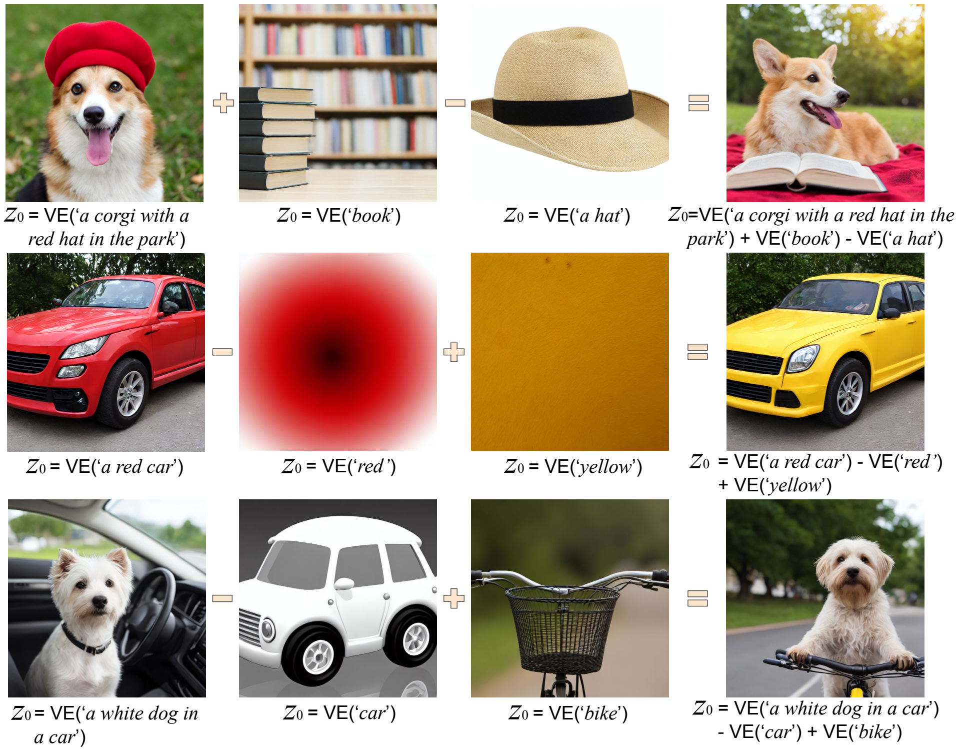
'A robot cooking dinner in the kitchen' (top left in blue box) ⟶ *'A panda eating hamburger in the classroom'* (bottom right in orange box)

Figure 6. **Linear interpolation in latent space.** We show images generated by linear interpolation between two text latents (*i.e.*, interpolation between z_0). Images generated by the first and second text latents are provided in the top-left and bottom-right corners.



'A corgi wearing a red hat in the park' (top left in blue box) ⟶ *'A teddy bear dressed in black wizard hat and robes sitting on the bed'* (bottom right in orange box)

Figure 7. **Linear interpolation in latent space.** We show images generated by linear interpolation between two text latents (*i.e.*, interpolation between z_0). Images generated by the first and second text latents are provided in the top-left and bottom-right corners.



Z_0 = VE('*a corgi with a red hat in the park*')
+
Z_0 = VE('*book*')
−
Z_0 = VE('*a hat*')
=
Z_0=VE('*a corgi with a red hat in the park*') + VE('*book*') - VE('*a hat*')
Z_0 = VE('*a red car*')
−
Z_0 = VE('*red*')
+
Z_0 = VE('*yellow*')
=
Z_0 = VE('*a red car*') - VE('*red*') + VE('*yellow*')
Z_0 = VE('*a white dog in a car*')
−
Z_0 = VE('*car*')
+
Z_0 = VE('*bike*')
=
Z_0 = VE('*a white dog in a car*') - VE('*car*') + VE('*bike*')

Figure 8. **Arithmetic in text latent space.** We map the text into the text latent space, perform arithmetic operations to obtain new latent representation, and use the resulting representation to generate the image. Latent z_0 used to generate each image is provided at the bottom.

[19] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, 2002. 1

[20] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 2

[21] 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

[22] 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

[23] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *ICCV*, 2021. 3

[24] Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *ICCV*, 2021. 2, 3

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

[26] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *TPAMI*, 2022. 1

[27] Thomas Schops, Johannes L Schonberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *CVPR*, 2017. 2

[28] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 1, 2

[29] Igor Vasiljevic, Nick Kolkin, Shanyi Zhang, Ruotian Luo, Haochen Wang, Falcon Z Dai, Andrea F Daniele, Mohammadreza Mostajabi, Steven Basart, Matthew R Walter, et al. Diode: A dense indoor and outdoor depth dataset. *arXiv preprint arXiv:1908.00463*, 2019. 2

[30] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *CVPR*, 2015. 1

[31] Wei Yin, Xinlong Wang, Chunhua Shen, Yifan Liu, Zhi Tian, Songcen Xu, Changming Sun, and Dou Renyin. Diversedepth: Affine-invariant depth prediction using diverse data. *arXiv preprint arXiv:2002.00569*, 2020. 3

[32] Wei Yin, Jianming Zhang, Oliver Wang, Simon Niklaus, Long Mai, Simon Chen, and Chunhua Shen. Learning to recover 3d scene shape from a single image. In *CVPR*, 2021. 3

[33] Chi Zhang, Wei Yin, Billzb Wang, Gang Yu, Bin Fu, and Chunhua Shen. Hierarchical normalization for robust monocular depth estimation. In *NeurIPS*, 2022. 3