## 6 Limitations and Future Work

Our method relies solely on the Mamba Block with a DiT-style layout and conditioning manner. However, a potential limitation of our work is that we cannot exhaustively list all possible spatial continuous zigzag scanning schemes given a specific global patch size. Currently, we set these scanning schemes empirically, which may lead to sub-optimal performance. Additionally, due to GPU resource constraints, we were unable to explore longer training durations, although we anticipate similar conclusions.

For future work, we aim to delve into various applications of the Zigzag Mamba, leveraging its scalability for long-sequence modeling. This exploration may lead to improved utilization of the Mamba framework across different domains and applications.

Ultimately, we anticipate that our scan path will be suitable for other linear attention models such as RWKV [81], xLSTM [11], HGRN [82], GLA [113], and several others listed at FLA [114][3].

## 7 Impact Statement

This work aims to enhance the scalability and unlock the potential of the Mamba algorithm within the framework of diffusion models, enabling the generation of large images with high-fidelity. By incorporating our cross-attention mechanism into the Mamba block, our method can also facilitate text-to-image generation. However, like other endeavors aimed at enhancing the capabilities and control of large-scale image synthesis models, our approach carries the risk of enabling the generation of harmful or deceptive content. Therefore, ethical considerations and safeguards must be implemented to mitigate these risks.

## 8 Appendix

**Table 7: The ablation about Hilbert and Zigzag scan path under various Order Receptive Field ( ORF ) on unconditional MultiModal-CelebA256.**

| Scan-ORF | $FID^{5k}$ |
|---|---|
| hilbert-2 | 61.67 |
| hilbert-8 | 27.38 |
| zigzag-2 | 15.45 |
| zigzag-8 | 13.32 |

[3] https://github.com/sustcsonglin/flash-linear-attention

**Table 8: Various methods for text-to-image generation on the MultiModal-CelebA 256 dataset.**

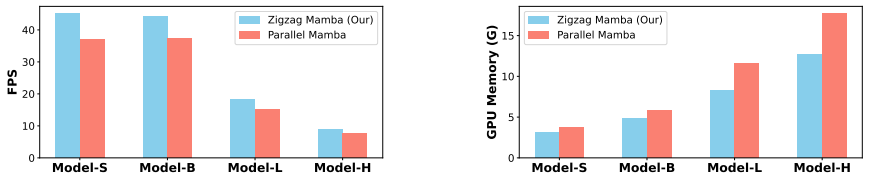
| Method | $FID^{5k}$ | $FDD^{5k}$ | $KID^{5k}$ |
|---|---|---|---|
| In-Context | 61.1 | 39.1 | 0.061 |
| Cross-Attention | 45.5 | 26.4 | 0.011 |

**Table 9: Details of ZigMa Model Variants.** We follow previous works [9, 25, 80] model configurations for the Small (S), Base (B) and Large (L) variants; we also introduce an XLarge (XL) config as our largest model. CA denotes the cross-attention for text-to-image conditioning.

| Model | Layers $N$ | Hidden size $d$ | #params |
|---|---|---|---|
| ZigMa-S | 12 | 384 | 31.3M |
| ZigMa-B | 12 | 768 | 133.8M |
| ZigMa-L | 24 | 1024 | 472.5M |
| ZigMa-XL | 28 | 1152 | 1058.7M |
| CA-ZigMa-S | 12 | 384 | 59.2M |
| CA-ZigMa-B | 12 | 768 | 214.1M |
| CA-ZigMa-L | 24 | 1024 | 724.4M |
| CA-ZigMa-XL | 28 | 1152 | 1549.8M |

### 8.1 Visualization

**FacesHQ** $1024 \times 1024$ **uncurated visualization** in Fig. 16.
**MS-COCO uncurated visualization.** We visualize the samples in Fig. 15.




**(a)** Model Complexity *v.s.* FPS. **(b)** Model Complexity *v.s.* GPU Memory.

**Figure 7: The ablation study about Model Complexity, FPS, GPU Memory**.

### 8.2 Spatial Continuity is Critical

We first explore the importance of spatial continuity in Mamba design by grouping patches of size $N \times N$ into various sizes: $2 \times 2$, $4 \times 4$, $8 \times 8$, and $16 \times 16$, resulting in groups of patch sizes $N/2 \times N/2$, $N/4 \times N/4$, $N/8 \times N/8$, and $N/16 \times N/16$, respectively. Then, we apply our designed Zigzag-8 scheme at the group level

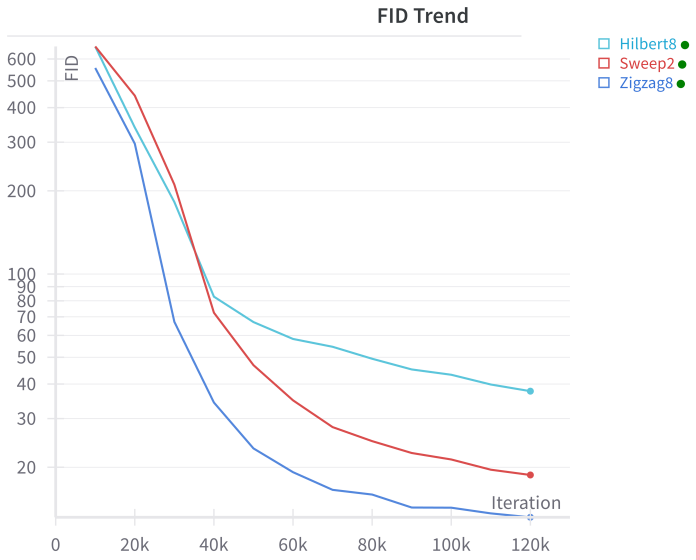
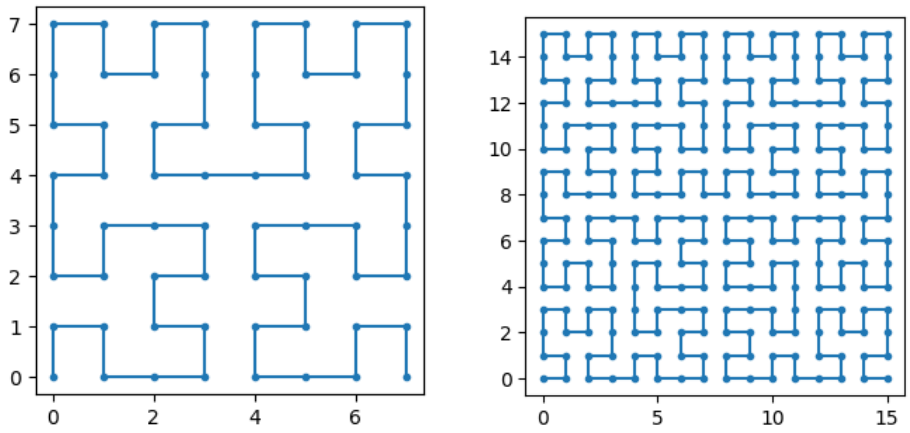



**Figure 8: The FID trends comparing the Hilbert scan, Sweep scan, and our Zigzag scan.** The y-axis is logarithmic scale.




**(a)** size=8 **(b)** size=16

**Figure 9: The Hilbert space-filling curve with various sizes**.

instead of the patch level. Figure 11 illustrates that with increased spatial continuity, notably improved performance is achieved. Furthermore, we compare our approach with random shuffling of $N \times N$ patches, revealing notably inferior performance under random shuffling conditions. All of these results collectively

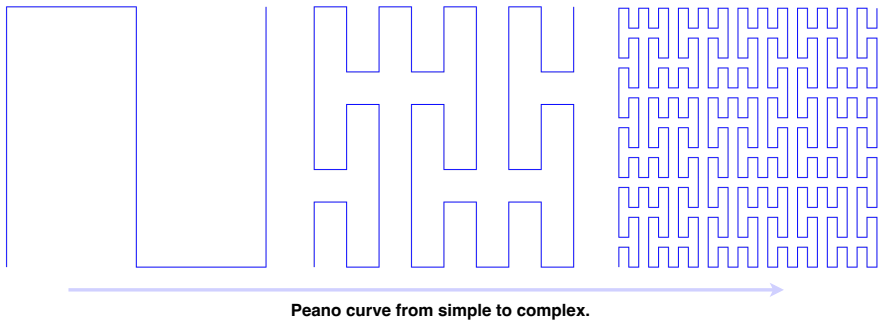



**Figure 10: The demonstration of Peano Curve.** The figure is borrowed from https://en.wikipedia.org/wiki/Peano_curve.

indicate that spatial continuity is a critical requirement when applying Mamba in 2D sequences.

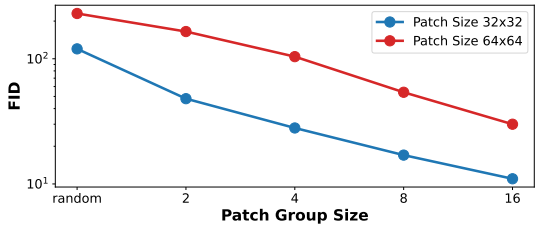



**Figure 11: Spatial Continuity Analysis.** As we incrementally enlarge the patch group size, the continuous segment of the patch also expands. This enhances spatial continuity, which we find improves FID on MultiModal-CelebA 256, 512 dataset.

### 8.3 Visualization

We demonstrate the image visualization of our best results on FacesHQ 1024 and MultiModal-CelebA 512 in Figure 12. For the visualization of videos, please refer to Appendix 8.1. It is evident that the visualization is visually pleasing across various resolutions, indicating the efficacy of our methods.

### 8.4 New Result about the Scanning Scheme

We also conduct basic ablations on various factors, including position embedding and various Hilbert space-filling curves. Unlike the experiments in the main paper, we perform these experiments on unconditional MultiModal-CelebA256 dataset for a uniform comparison. We train the network for 100,000 steps.


**Figure 12: Visualization of various resolutions on FacesHQ** $1024 \times 1024$ **and MultiModal-CelebA** $512 \times 512$**.** Our generated samples present high fidelity across various resolutions.

**Exploration of Hilbert space-filling curve.** Primarily, we ablate the Hilbert scan curve [75], as depicted in Figure 9. There are also eight variants of this scan considering different angles and starting points. We rearrange them in a similar manner to our Zigzag scan. All parameters are kept consistent for a fair comparison. We utilize the Gilbert algorithm [4] to guarantee that the Hilbert curve remains continuous across any square size. We train our network on single A100-SXM4-80GB for 120k iterations. We evaluate the FID on 5,000 images for a fixed step, the FID curve is demonstrated in Figure 8.

While the Hilbert space-filling curve offers increased locality compared to our zigzag scan and maintains continuity, its complex structure appears to hinder the SSM's ability to work on the flattened sequence, resulting in a worse inductive bias than our zigzag curve on natural images. Therefore, we hypothesize that structure may hold greater significance than locality in generative tasks.

**Hilbert Curve is difficult to optimize.** We show the result in Table 7. We can observe that the performance of the Hilbert scan path drops significantly, even if we decrease the Order of Receptive Field (ORF). This confirms the assumption that the Hilbert scan path is difficult to optimize, even when considering only two different schemes of the Hilbert scan.

**Another Interpretation: Zigzag scan is the simplest Peano curve.** Our Zigzag scan can be seen as the simplest case of Peano Curve as shown in Figure 10.

### 8.5 New Result of 2D visual data

**The variants of our ZigMa Models.** We list the variants of our model in Table 9. We use the Base (B) Model as the default. Applying the cross-attention model is optional, as this module can introduce some parameter and speed burdens. However, any advancements in attention optimization can be seamlessly integrated into our model.

[4] https://github.com/jakubcerveny/gilbert

**Ablation of patch size.** We conducted an ablation study on patch sizes ranging from 1, 2, 4, to 8 in Figure 13, aiming to explore their behaviors under the framework of Mamba. The results reveal that the FID deteriorates as the patch size increases, aligning with the common understanding observed in the field of transformers [25, 98]. This suggests that smaller patch sizes are crucial for optimal performance.

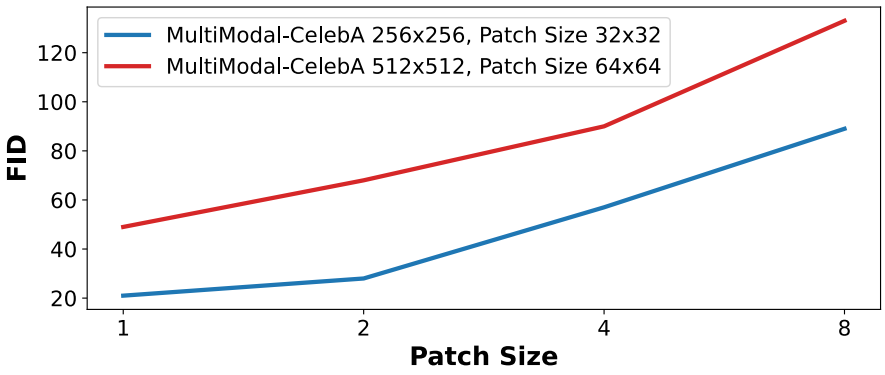
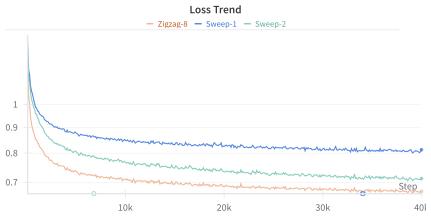



**Figure 13:** FPS *v.s.* Patch Size.

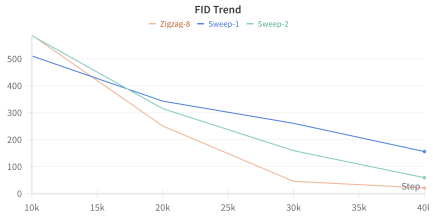
**Ablation study about the Model Complexity and FPS/GPU-Memory.** As shown in Figure 7. Our method can achieve much better parameter efficiency when incorporating the receptive order. The receptive order refers to the cumulative spatial-continuous zigzag scan path in 2D images, which we've incorporated into the Mamba as an inductive bias. We list the parameter consumption when we gradually increase the receptive order in Figure 7. The receptive order refers to the cumulative spatial-continuous zigzag scan path in 2D images, which we've incorporated into the Mamba as an inductive bias.

**Loss and FID curve.** The training loss curve and the FID curve are demonstrated in Figure 14. The loss and FID show the same trend, with our Zigzag Mamba consistently outperforming other baselines like Sweep-1 and Sweep-2.

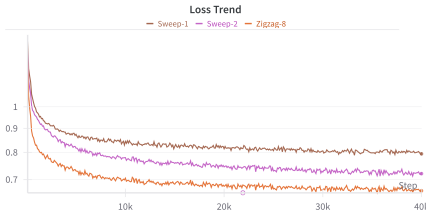
**In-context v.s. Cross Attention** We compare our cross-attention with in-context attention in Table 8. For in-context attention, we concatenate the text tokens with the image tokens and feed them into the Mamba block. Our results demonstrate that in-context attention performs worse than our cross-attention. We hypothesize that this is due to the discontinuity between the text tokens and the image patch tokens. We discovered that PointMamba [61] arrives at the same conclusion and hypothesis as we do.



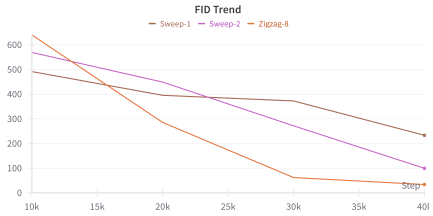

**(a)** Loss trend of the MultiModal-CelebA256.




**(b)** FID trend of the MultiModal-CelebA256.




**(c)** Loss trend of the MultiModal-CelebA512.




**(d)** FID trend of the MultiModal-CelebA512.

**Figure 14: The loss and FID trend** under various resolutions on dataset MultiModal-CelebA. Sweep-1 and Sweep-2 are the Mamba scans without spatial continuity, while Zigzag-8 represents our method. This is the direct log from weight-and-bias (wandb).

### 8.6 New result of 3D Visual Data

**The choice of the 3D Zigzag Mamba.** For Factorized 3D Zigzag Mamba in video processing, we deploy the *sst* scheme for factorizing spatial and temporal modeling. This scheme prioritizes spatial information (ss)complexity over temporal information (t), hypothesizing that redundancy exists in the temporal domain. There are numerous other possible combinations of s and t to explore, which we leave for future work.

### 8.7 More related works

Several works [102, 103] have demonstrated that the State-Space Model possesses universal approximation ability under certain conditions. Mamba, as a new State-Space Model, has superior potential for modeling long sequences efficiently, which has been explored in various fields such as medical imaging [31, 73, 86, 108, 111], image restoration [38, 122], graphs [12, 99], NLP word byte [100], tabular data [2], human motion synthesis [121], point clouds [61, 118], image generation [27], semi-supervised learning [107],interpretability [5], image dehazing [122] and pan sharpening [41]. It has been extended to Mixture of Experts [7], spectral space [1], multi-dimension [59, 70, 77, 123] and dense connection [40]. Among them, the most related to us are VisionMamba [70, 123], S4ND [77] and Mamba-ND [59]. VisionMamba [70, 123] uses a bidirectional SSM

in discriminative tasks which incurs a high computational cost. Our method applies a simple alternative mamba diffusion in generative models. S4ND [77] introduces local convolution into Mamba's reasoning process, moving beyond the use of only 1D data. Mamba-ND [59] takes multi-dimensionality into account in discriminative tasks, making use of various scans within a single block. In contrast, our focus is on distributing scan complexity across every layer of the network, thus maximizing the incorporation of inductive bias from visual data with zero parameter burden.

Certain studies, such as Li's work in 2024 [60], often explore the order of patches in token-based networks. However, while these studies concentrate on auto-regressive transformers, our focus is on the Mamba-based structure.

Several previous works [51, 119] have focused on the shuffling operation to exchange information along the spatial or channel dimension. For instance, the Shuffle Transformer [51] applies shuffling to spatial tokens to encourage cross-reasoning outside the attention windows. Our method follows the same approach. We shuffle the tokens to maintain a continuous spatial-filling scan path, promoting optimization across various layers. Given that the shuffling order differs across the layers, it could potentially avert the overfit problem [70].

### 8.8 More Details

**Double-Indexing Issue for $\Omega_i$.** As shown in Fig. 2. We need to *arrange* and *rearrange* operation that needs to conduct indexing along the token number dimension to achieve spatial-continuous mamba reasoning, as the indexing can be time-consuming [5] when considering the large token numbers, We can formulate the arrange and rearrange operation as follows:

$$\Omega_i^{'} = \bar{\Omega}_{i-1} \cdot \Omega_i, \tag{14}$$

$$\mathbf{z}_{i+1} = \text{scan}(\mathbf{z}_{\Omega_i^{'}}), \tag{15}$$

$$\tag{16}$$

where $\bar{\Omega}_{-1} = I$, this assumes that the Mamba-based networks are permutation equivariant to the order of the tokens. They require 50% fewer indexing operations, a point which we reiterate here for clearer comparison:

$$\mathbf{z}_{\Omega_i} = \text{arrange}(\mathbf{z}_i, \Omega_i), \tag{17}$$

$$\bar{\mathbf{z}}_{\Omega_i} = \text{scan}(\mathbf{z}_{\Omega}), \tag{18}$$

$$\mathbf{z}_{i+1} = \text{arrange}(\bar{\mathbf{z}}_{\Omega_i}, \bar{\Omega}_i), \tag{19}$$

**Evaluation Metrics.** For image-level fidelity, we use established metrics such as Fréchet Inception Distance (FID) and Kernel Inception Distance (KID), following previous works. However, since studies [21, 92] have shown that FID does not

[5] We found that using the torch.compile() can largely ease the time issue, see https://taohu.me/zigma for more detail comparison.

fully reflect human-based opinions, we also adopt the Fréchet DINOv2 Distance (FDD) using the official repository [6]. Our method primarily involves sampling 5,000 real and 5,000 fake images to compute the related metrics.

We primarily consider two metrics for video fidelity evaluation: framewise FID and Fréchet Video Distance (FVD) [97]. We sample 200 videos and compute the respective metrics based on these samples.

We utilize the EMA models for evaluation, as they can deliver superior performance, as indicated in [117].

**Extra Training Details.** For text-conditioned generation, we conduct the experiments on the MultiModal-CelebA $256^2$,$512^2$ [110] and MS COCO $256 \times 256$ [63] datasets. Both datasets are composed of text-image pairs for training. Typically, there are 5 to 10 captions per image in COCO and MultiModal-CelebA. We convert discrete texts to a sequence of embeddings using a CLIP text encoder [83] following Stable Diffusion [84]. Then these embeddings are fed into the network as a sequence of tokens.

**The training parameters of various datasets** are listed in Tab. 10. We don't apply any position encoding because Mamba, unlike Transformer, is not permutation invariant. Therefore, its position is automatically encoded by its order in Mamba. Surprisingly, we also found that adding extra learnable position encoding can lead to better performance compared to the baseline. We hypothesize that these extra inductive biases can further benefit performance, even though the order of the tokens already incorporates some bias. For the COCO dataset, a weight decay of 0.01 can contribute to marginal FID gains (approximately 0.8).

**The conditioning of timestep and prompt.** The conditioning process is illustrated in Algorithm 1. For the Mamba block, we incorporate the condition information. Specifically, we concatenate the condition token with the image patch token to enhance the conditioning mechanism.

[6] https://github.com/layer6ai-labs/dgm-eval

**Table 10: Hyperparameters and number of parameters for our network in various datasets.** All models are trained on a single A100 with 40GB of VRAM using a bfloat16 of accelerator package.

| | FacesHQ 1024 | MS-COCO 256 | MultiModal-CelebA 512 | UCF-101 |
|---|---|---|---|---|
| Autoencoder $f$ | 8 | 8 | 8 | 8 |
| $z$-shape | $4 \times 128 \times 128$ | $4 \times 32 \times 32$ | $4 \times 64 \times 64$ | $4 \times 32 \times 32$ |
| Model size | 133.8M | 133.8M | 133.8M | 133.8M |
| Patch size | 2 | 1 | 1 | 2 |
| Channels | 768 | 768 | 768 | 768 |
| Depth | 12 | 12 | 12 | 12 |
| Optimizer | AdamW | AdamW | AdamW | AdamW |
| Batch size/GPU | 8 | 8 | 4 | 8 |
| GPU num | 32 | 32 | 16 | 16 |
| Learning rate | 1e-4 | 1e-4 | 1e-4 | 1e-4 |
| weight decay | 0 | 0 | 0 | 0 |
| EMA rate | 0.9999 | 0.9999 | 0.9999 | 0.9999 |
| Warmup steps | 0 | 0 | 0 | 0 |
| A100-hours | 768 | 768 | 384 | 384 |

**Algorithm 1** Mamba Block

```
def mamba_block(x, t, c = None):
    # x: input data, shape [B, (W x H) , C] or [B, (T x W x H) , C]
    # t: timestep, (B, C)
    # c: condition, (B, D, C)
    x = reshape(x) # (B, K, C)

    def _mamba(x):
        x = rearrange(x) # rearrange by a zigzag manner
        x = mamba(x)
        x = rearrange_back(x)# rearrange back by a zigzag manner

    m, n = AdaLN( t )
    x = _mamba( x * m + n ) + x

    if c is not None:
        p, q = AdaLN( c )
        x = cross_attention( x * p + q ) + x
    return x
```

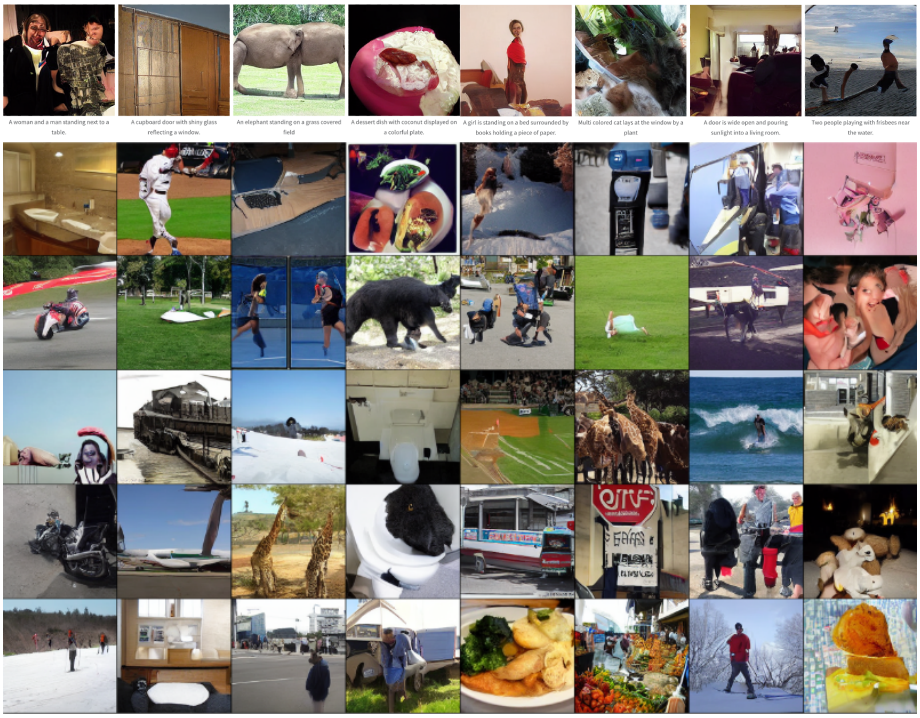



**Figure 15: The Uncurated Visualization of MS-COCO dataset.** The first row is illustrated with pairs of images and their captions, while the remaining rows only images.

**Figure 16: Uncurated Visualization of FacesHQ dataset.**

**Figure 17: Uncurated Visualization of Landscape HQ dataset [87], with 5k FID of 10.07.**