

Story Visualization by Online Text Augmentation with Context Memory

Daechul Ahn^{1,§} Daneul Kim² Gwangmo Song³ Seung Hwan Kim³
Honglak Lee^{3,4} Dongyeop Kang⁵ Jonghyun Choi^{1,†}

¹Yonsei University ²GIST ³LG AI Research ⁴University of Michigan ⁵University of Minnesota

{dcahn, jc}@yonsei.ac.kr flytodk98@gm.gist.ac.kr gwangmo.song@lgresearch.ai
skcruise@gmail.com honglak@eecs.umich.edu dongyeop@umn.edu

Abstract

Story visualization (SV) is a challenging text-to-image generation task for the difficulty of not only rendering visual details from the text descriptions but also encoding a long-term context across multiple sentences. While prior efforts mostly focus on generating a semantically relevant image for each sentence, encoding a context spread across the given paragraph to generate contextually convincing images (e.g., with a correct character or with a proper background of the scene) remains a challenge. To this end, we propose a novel memory architecture for the Bi-directional Transformer framework with an online text augmentation that generates multiple pseudo-descriptions as supplementary supervision during training for better generalization to the language variation at inference. In extensive experiments on the two popular SV benchmarks, i.e., the Pororo-SV and Flintstones-SV, the proposed method significantly outperforms the state of the arts in various metrics including FID, character F1, frame accuracy, BLEU-2/3, and R-precision with similar or less computational complexity.

1. Introduction

Story visualization (SV) [17] is a task of generating a sequence of images from a paragraph, i.e., a sequence of natural language sentences. It is challenging for the requirement of rendering the visual details in images with convincing background of a scene – seasonal elements, environmental objects such as table, location, and the proper character appearing, which here we refer to as *context*, spread across the given text sentences. Specifically, it needs to encode implicit context presented in the given sentences since each one often omit visual details (i.e., they may be spread over the sentences) necessary to generate a semantically correct im-

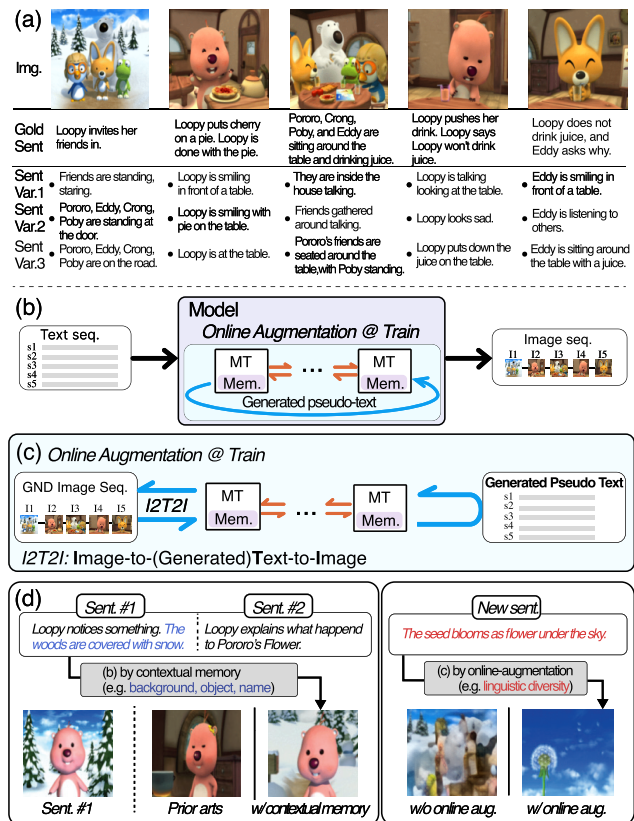


Figure 1. Linguistic variations in story visualization and the overview of the proposed method. (a) An example of a story data with various linguistic variations (Var.#) for each image. (b) Modeling temporal context spread across sentences by our context memory (Sec. 3.1). (c) Addressing linguistic variations by online text augmentation from each image for every epoch (Sec. 3.2). (d) Benefit of the proposed context memory (left) and online text augmentation (right).

age. For example, we can think of Fig. 1-(d), where Sent.#1 (i.e., sentence 1) and Sent.#2 (i.e., sentence 2) are given as sequences. After Sent.#1 is given, generated image by Sent.#2 often exhibits a background that is not semantically

§: work done while interning at LG AI Research. †: corresponding author.
Code: <https://github.com/yonseiivnl/cmota>

correct [17, 22, 23]. However, if we use contextual information given in Sent.#1, “The woods are covered with snow”, it leads to correct image with matching background.

In addition, widely used benchmark datasets for story visualization task [9, 17] provide a single text-image pair for training and inference, mostly due to the annotation budget constraint. This prevents the model from learning language variations, and thus harms the linguistic generalization performance of the model.

To address the aforementioned challenges without requiring large scale data and models, we propose a new memory scheme in bi-directional Transformer for encoding the context which generates pseudo-texts in an online fashion to address linguistic variations at inference. We call our model as **Context Memory and Online Text Augmentation** or **CMOTA** for short. We empirically validate that our model outperforms the state-of-the-art SV methods by large margins on various metrics evaluated with widely used benchmarks in the literature, *i.e.*, Pororo-SV and Flintstones-SV.

Note that while large pre-trained models [3, 5–7, 31, 43] have shown great success in synthesizing an image or a video from a language description [13, 32, 38], huge computational complexity and large training data makes the models prohibited. Moreover, although we propose and evaluate the model for the standard benchmark datasets without large pretraining data trained with a large model, it would be interesting to apply and evaluate the proposal in large models for further improvement.

We summarize our contributions as follows:

- We propose a new memory architecture for Transformer to selectively make use of contexts in a story paragraph.
- For better generalization of linguistic variations in the given paragraph at inference, we generate pseudo-texts and augment them in an *online fashion* for richer linguistic supervision.
- Our model significantly outperforms prior arts (even some hyper-scale models) by large margins in five evaluation metrics with similar or less computational complexity.

2. Related Work

Story visualization. StoryGAN [17] is one of the recent methods that utilized a story-level discriminator to improve global consistency in generated images. CP-CSV [39] disentangles figure and background information to enhance character consistency. In order to improve global semantic matching between paragraph and generated image sequence, DuCo-StoryGAN [23] presents a pre-trained video captioner as an auxiliary loss along with other design improvements on top of StoryGAN. More recently, VLC-StoryGAN [22] utilizes constituency parse-trees and common sense knowledge to improve consistency and an object-level feedback loop to

improve image quality. Another recent work VP-CSV [2] is a two-stage approach, *i.e.*, 1) character generation and 2) background completion, using Transformer model to address this task. We discuss these in more detail in the supplementary material for the space sake. But the prior arts largely neglect to encode the story narrative, which is our primary contribution here.

Very recently, Maharana *et al.* propose a new task setup of story continuation; using first image as a condition. They fine-tune the large model DALL-E [32] for the SV, which they call StoryDALL-E [24]. Ours differs from it in several aspects as follows. We have explicit memory connections between adjacent image generators using the context memory to globally encode the sentences. In contrast, [24] utilizes global story embedding as additional input for understanding context. Further, while it is huge in size (1.3B parameters), trained with 14 million text-image pair, ours are much larger (97M parameters) and outperforms it in multiple metrics by large margins; even in the image quality metric, FID. Please refer to Sec. 4.2 for empirical comparisons.

Text-to-video generation. Similar to story visualization, text-to-video generation also generates multiple frames from a given text. Since the pioneering work of Sync-DRAW [25] in text-to-video generation, [9, 18, 27] utilize generative adversarial network for high-quality image generation.

Recently, high-quality video generation models are proposed, including GODIVA [46] and NÜWA [47]. Furthermore, recent studies generate high-resolution videos, making sequential frames in high-quality [11, 38]. However, most of state-of-the-art text-to-video model [11, 38] generates a video from a *single* sentence, mostly having consistent backgrounds. Very recently, there is a method proposed to generate a video from a long paragraph [45]. Although they generate the video in a long time horizon, they require both a mega-scale model trained with huge data and a detailed paragraph where each sentence is describing the scene that are close in time. In contrast, story visualization requires generating frames arbitrarily distant in time (*i.e.*, so-called ‘key-frames’) corresponding to different sentences, requiring to generate an image sequence that have contextually convincing background.

Text-to-image generation. Text-to-image generation is a sub-problem of story visualization, with literature focusing on semantic relevance and resolution improvements. Recently, text-based image synthesis has been greatly improved with the help of a vast amount of training data with a hyper-scale model including DALL-E [32] and its successor DALL-E2, CogView [5] and Make-A-Scene [8] using a sketch input.

Although text-to-image generation models generates very high-quality images, it may lack encoding the context, metaphoric sentences spread across multiple sentences. In

addition, naively using state-of-the-art text-to-image generation models is computationally prohibited. For example, diffusion-based models [31, 34] have hyper-scale model size, (e.g., Imagen [34] parameter count of 2-B, DALL-E2 [31] parameter count of 3.5-B) making it non-trivial for applying it in a wide range of inference scenarios that may not have the sufficient computing resource. Here, we consider relatively light architectures as our base model for computational efficiency. More discussion are in the supplement.

Transformer using memory. To mitigate context fragmentation issue [4], *i.e.*, losing long-term dependency over a data stream in context, there are efforts to encode long-term contexts in generating an image sequence. [4, 15] adopt a recurrent path into transformer architecture. Specifically, the modeling of new data segments is conditioned on historical hidden states produced in the previous time step and uses highly summarized memory states. Unlike the conventional memory architectures, we propose a novel memory that has a dense connection from the past with attentive weighting schemes for their better usage (Sec. 3.1).

Online augmentation. While offline data augmentation that prepares data outside of learning process is prevalent in many literature [1, 12, 19], online-augmentation that depends on training is seldom explored especially in story visualization task [40]. [35, 40] propose an online augmentation for image classification and VQA task. [26] uses bi-level optimization in image classification. In medical domain, [48] investigate the online augmentation for personalized image based Histopathology diagnosis [48]. Note that we show the benefit of online augmentation over the offline augmentation in the SV for the first time in this literature.

3. Approach

There are multiple challenges in story visualization, including (1) generating semantically natural images without artifacts from a description, (2) encoding the *context* (e.g., consistent background of a scene - seasonal elements, environmental objects such as table and location, and consistent characters) spread across the sentences in a given paragraph [17] and (3) addressing the linguistic variation at the inference time (*i.e.*, a given text description may not be in different writing style).

For photo-realistic image generation from a language description, we recently witness unprecedented improvements in the quality of generated images by the help of large scale model [3, 5-7, 31, 43]. To leverage the large model’s benefits for the story visualization task, we may use it to generate an image sequence by *gradually* concatenating sentences to visualize the story (*i.e.*, use the first sentence to generate the first image, use a concatenated sentence of the first and the second to generate the second, and so on). Although this

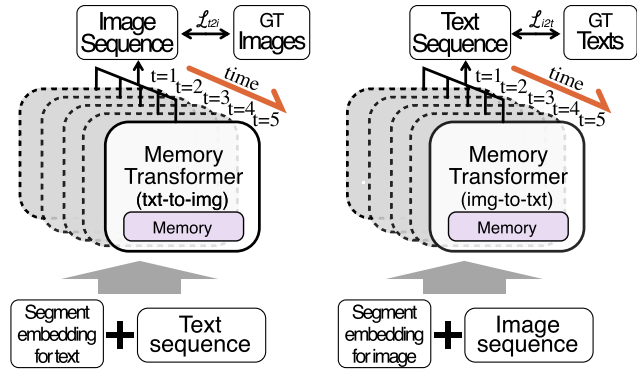


Figure 2. **Bi-directional (between text and image modality) Transformer for text or image ‘sequence’ generation with the proposed memory.** Our bi-directional ‘base’ model can simultaneously generates a sequence of both text and image from $t = 1 \dots 5$ with the proposed memory (Sec. 3.1). We use a segment embedding to indicate input modality and enable output of any modality.

may generate a single photo-realistic image, it is not able to capture the context sparsely spread across the sentences as story progresses (see supplement for more discussion).

A more involved way of using the large model for SV is to fine-tune it for the downstream SV or using a video generation models. However, as the most high-performing models are huge in size and the codes and the pre-trained models are not publicly available, the computational cost and reproducibility reduces practicality.

For a given dataset without using large extra training data, we propose a new memory module to better encode past information with a Transformer by an attentively weighted densely connected architecture (Sec. 3.1). We further propose to generate pseudo-texts during the learning process to augment (online-augmentation) for better linguistic generalization without requiring large external data by learning the *bi-directional* Transformer [14, 30, 33] in both directions of generating images from texts and *vice versa* as illustrated in Fig. 2 (Sec. 3.2).

Base model. As shown in Fig. 2, we use bi-directional (*i.e.*, multi-modal) transformer that iteratively generates images and texts in both ways. Similar to [32], the image tokens are sequentially predicted from the input text sequences by the Transformer. Then, the decoder of the VQ-VAE [42] translates the predicted image tokens into image sequence. The text tokens are also sequentially predicted from the input image token sequence by the same Transformer.

Particularly, for the *bi-directional multi-modal* generation, *i.e.*, generating simultaneously text and image from the unified architecture, we add two embeddings; a positional embedding for absolute position between tokens and a segment embedding for distinguishing source and target. Tokens of a text ($\{t_1, \dots, t_m\}$) and an image ($\{z_1, \dots, z_n\}$) (m, n : # of tokens for text and image) are fed into the Trans-

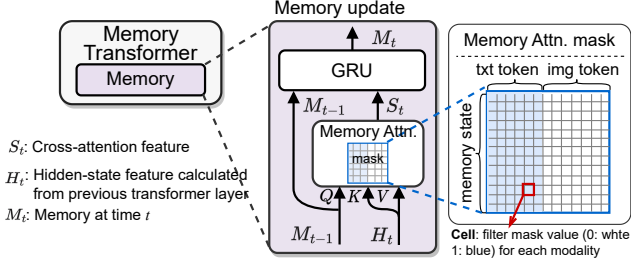


Figure 3. **Proposed memory module for the bi-directional Transformer.** Memory module updates current memory M_t by utilizing memory attention and GRU operation with propagated memory, M_{t-1} , and previously calculated hidden state, H_t^l . To produce image, we propose a novel memory attention (Attn.) mask denoted by blue squares to select only text tokens for less bias to previously generated image tokens. Within the memory attention mask, white-colored squares filter out the content, *i.e.*, image (img) token, whereas the blue squares allow the text (txt) token to propagate through for memory interaction.

former to predict tokens in the other modality in multiple epochs with the following objective function written as:

$$\begin{aligned} \mathcal{L}_{j,t2i} &= \sum_{k=1}^n -\ln p_j(z_k | t_1, \dots, t_m, z_1, \dots, z_{k-1}), \\ \mathcal{L}_{j,i2t} &= \sum_{k=1}^m -\ln p_j(t_k | z_1, \dots, z_n, t_1, \dots, t_{k-1}), \\ \mathcal{L}_j &= \mathcal{L}_{j,t2i} + \lambda_1 \mathcal{L}_{j,i2t}, \end{aligned} \quad (1)$$

where $p_j(\cdot)$ is a likelihood of j -th generated tokens in one modality given the other modality, $\mathcal{L}_{j,t2i}$ is a loss corresponding to j -th text-to-image generation (*i.e.*, negative log likelihood), $\mathcal{L}_{j,i2t}$ is vice versa. λ_1 is a balancing hyper-parameter.

To train the model, we iteratively train the generation model in each modality multiple times per each epoch.

3.1. Context Memory

To encode the context and calculate the propagated memory, as depicted in Fig. 3, we first apply cross attention between the current hidden state, $H_t^l \in \mathcal{R}^{T_c \times d}$ calculated from $(l-1)$ -th transformer layer and memory state at time $(t-1)$, $M_{t-1} \in \mathcal{R}^{T_M \times d}$ (T_c : # tokens, T_M : # memory states, d : hidden state dimension). We then obtain $S_t = \text{Attn}(M_{t-1}, H_t, H_t)$, where $\text{Attn}(Q, K, V) := \text{Softmax}\left(\frac{QK^T}{\sqrt{d_q}}\right)V$ and d_q is a query dimension (Q), with the memory attention mask depicted in the blue box in Fig. 3.

In particular, we apply a memory mask in the attention operation to select text tokens as memory content (depicted as blue-shaded grid cells in the right of Fig. 3) because including image content as memory could be strong constraint for text-to-image generation (see empirical studies of the

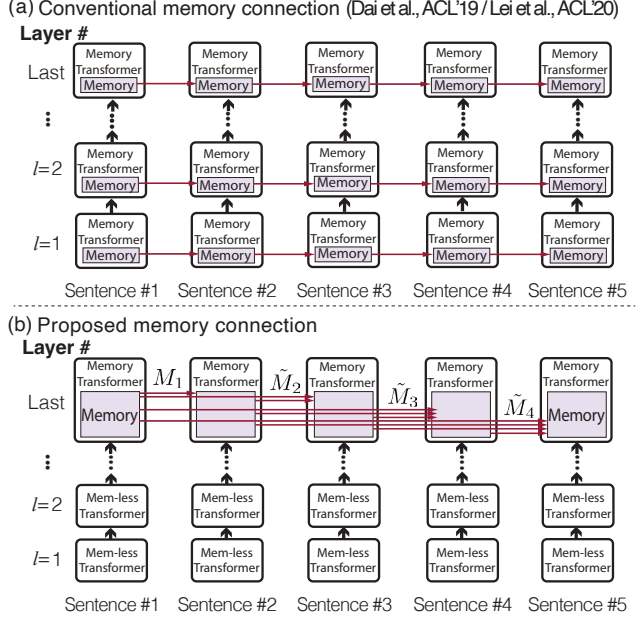


Figure 4. **Comparison of our memory connection scheme to the conventional one [4, 15].** We depict the memory connections in the multi-layer Transformer architecture per each sentence input (unfolded in a row). Unlike (a) the previous memory architecture having a single memory path from the immediate past in all layers, (b) ours has a dense connection from all the pasts to be attentively weighted (denoted here by \tilde{M} 's) in only the last layer. We discuss its empirical benefit in Sec. 4.2.2.

proposed mask in the supplementary material). Then we feed the S_t and M_{t-1} to the GRU in order to compute the information to be stored in the memory, M_t , propagated to $(t+1)$.

Memory connection. Following [4, 15], we stack multiple Transformer's layers, depicted in Fig. 4-(b). Compared to conventional memory updating architectures with serial hidden state connections in all layers (Fig. 4-(a)), our approach only has memory connections in the last layer, which we call it as partial-level memory augmented (PMA). This is because the later layers (closer to the last layer) achieve better representations with higher-level features, abstract and structured representations [28, 29, 41] with improved computational efficiency, which is unlike prior works (see Sec. 4.2.2 and supp for further discussion).

Additionally, not all historic information is equally important for generating an image at a time step. Similar to the masked self-attention [43], we attentively weight the past information for better modeling of sparse context as:

$$\begin{aligned} \bar{M}_{1:(t-1)} &= \text{Attn}(M_{(t-1)}, M_{[1:(t-2)]}, M_{[1:(t-2)]}), \\ \tilde{M}_{(t-1)} &= [M_{t-1}; \bar{M}_{1:(t-1)}], \quad (3 \leq t \leq 5), \\ H_t^l &= \text{Attn}(H_t^l, [H_t^l; \tilde{M}_{(t-1)}], [H_t^l; \tilde{M}_{(t-1)}]), \end{aligned} \quad (2)$$

where $M_{(t-1)}$ is a memory at time $(t-1)$ and $[1:(t-2)]$ refers to the concatenation from time 1 to $(t-2)$ as depicted in Fig. 4-(b). By doing so, we fuse the contextual information into current hidden state H_t . At $t=2$, we use M_1 , instead of \tilde{M}_1 as the M_1 is only available. We call it as attentively weighted memory (AWM).

3.2. Online Text-Augmentation

Vedantam *et al.* argue that multiple descriptions for an image help generalization as they address language variations presented in descriptions [44]. Hence, a number of image captioning datasets [10, 20, 37] provide multiple natural language directives, obtained by multiple human annotators, in both training and evaluation splits. But the SV benchmark datasets [17] provide only a single sentence per an image.

To address the linguistic variations of a text input at inference process, we first consider to generate a pseudo text by a well-trained image-to-text generation model, which we refer it as *offline-augmentation* [16]. But the offline augmentation generates only a single sentence, which may not provide sufficient diversity. Instead, we propose to generate multiple pseudo-texts and augment them in an *online* fashion when training our model to increase the diversity. We call it *online text augmentation*, depicted in Fig. 5.

Thanks to our bi-directional multi-modal architecture, we can naturally integrate the process of generating pseudo-texts to the process of learning image-to-text model and the text-to-image generation model as depicted in Fig. 5. In the early epochs, less meaningful sentences are generated, but as training progresses, more meaningful sentences are generated (see orange box in Fig. 5). As a side-product, by supervising the model learning with intermediate goals at each time step, we expect to expedite the convergence of learning. When we use the online text augmentation, we can rewrite the objective as:

$$\mathcal{L}_{j,pt2i} = \sum_{k=1}^n -\ln p_j(z_k | \hat{t}_1, \dots, \hat{t}_m, z_1, \dots, z_{k-1}), \quad (3)$$

$$\mathcal{L}_j = \mathcal{L}_{j,t2i} + \lambda_1 \mathcal{L}_{j,i2t} + \lambda_2 \mathcal{L}_{j,pt2i},$$

where $\mathcal{L}_{j,pt2i}$ is the additional loss with the augmented pseudo-texts and $\mathcal{L}_{j,t2i}$, $\mathcal{L}_{j,i2t}$ are defined in Eq. 1 and λ_1 and λ_2 are balancing hyper-parameters. \hat{t} means the pseudo-text token, predicted during online augmentation without gradient flow. Fig. 6 shows pseudo-texts generated by our method. Detailed training procedure and more examples are in the supplementary material.

4. Experiments

4.1. Experimental Setup

Datasets. We use two popular benchmark datasets for evaluating the task of story visualization. Following [23], we use

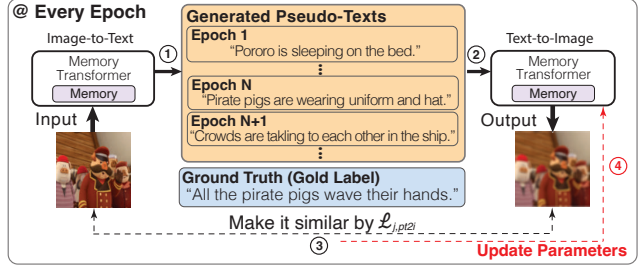


Figure 5. **Online text augmentation.** We iteratively augment the pseudo-text for better linguistic generalization. (1) generating a pseudo-text at each epoch for each ground-truth image, without gradient flow. (2) generating an image for every epoch. (3,4) learn the text-to-image model with L_{CE} . In early epochs, it generates less meaningful pseudo-text (thus discard them), but produces pseudo-texts matching with the input image as the training progresses.

Pororo-SV dataset for story visualization task without data overlap. Following prior work [22], we use Flintstones-SV which was originally exploited in the text-to-video synthesis task [9].

Metrics. For the evaluation metrics, we use Fréchet Inception Distance (FID) (used in [22, 39]) for visual quality of the generated images, character classification score (Char. F1, Frm. Acc.) [17] for character consistency, and global semantic matching (B-2/3, R-Prec.) [22, 23] between generated image sequence and story paragraph.. We elaborate on the details of the datasets and implementation in the supplementary material for the sake of space.

Baselines. We compare our method to a number of prior arts including state-of-the-art story visualization methods such as StoryGAN [17], CPCSV [39], DuCo-StoryGAN [23], VLC-StoryGAN [22] and VP-CSV [2] for both quantitative and qualitative analysis in the story visualization setting. Moreover, we compare [24] in the story continuation setting.

Implementation details. To train our model, we first tokenize the text and image inputs. Particularly, the image inputs are encoded by the encoder of the VQ-VAE [42] into image tokens and decoded by the decoder of the VQ-VAE for generating image sequence. Once the inputs are tokenized, we append two special embedding tokens to the text and image tokens. These tokens signify the beginning of each modality, denoted as ‘SOS’ (start of sentence) and ‘SOI’ (start of image). Subsequently, we add two embeddings: (1) a positional embedding to indicate absolute position of each token and (2) a segment embedding to distinguish source and target modality, *e.g.*, for text-to-image generation task, the source corresponds to the text while the target corresponds to the image. Through the utilization of segment embeddings to distinguish between source and target, we enable the model to generate target data from source data





Image				
Generated Pseudo-Texts	① Pororo is rubbing the lamp while closing his eyes. ② Pororo closes his eyes and talks. ③ Pororo looks down at lamp then looks left and right.	Eddy asks Crong what Crong is doing. Crong points something to Eddy. Eddy calls Crong. Crong looks back.	Fred is talking to someone with his hand up in the front yard. Fred is in the doorway yelling. Fred is talking to someone with his hand up in the front yard.	Fred and Barney are in a pink room. Fred and Barney are standing in a room. Fred and Barney are in a room .
Ground Truth (Gold Label)	Pororo is holding and looking down at a lamp.	Eddy calls Crong. Crong looks back.	Fred is talking to someone with his hand out angrily as he stands in front of a doorway.	Barney and Fred are in a room.

Figure 6. **Generated pseudo-texts by the proposed method during training.** We depict the generated pseudo-texts in training procedure. We utilize them as supplementary supervisions to address linguistic diversity of a sentence at inference.

within a unified architecture, as illustrated in Fig. 2. Here, we set the token length for each modality as $T_{text} = 80$ and $T_{image} = 256$ (i.e., 16×16), the hidden dimension size to 512, the number of transformer layer to 6, and the number of attention heads to 16, thereafter the number of trainable parameters approximately 93.7-M as shown in Tab. 1. Particularly, we generate an image per each sentence without any sampling process. We train the CMOTA using AdamW [21] optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.95$, $\epsilon = 1e-8$, weight decay multiplier with $1e-2$ and learning rate with $4.5e-6$ multiplied by batch size. Moreover, we set the hyper-parameters for loss balancing in Eq. 3 as $\lambda_1 = 1.0$ and $\lambda_2 = 0.5$ and the number of memory state, T_M as 1. Finally, to generate a high-resolution image, we modified the VQ-VAE with the additional trainable parameters (i.e., 7.1-M shown in Tab. 1). This modification allows for the generation of high-resolution images, increasing the image resolution from 64×64 to 128×128 .

4.2. Quantitative Analysis

4.2.1 Comparison with the State of the Arts

In Table 1, we summarize the performance of the prior arts and the proposed CMOTA in Pororo-SV [17] and Flintstones-SV [9] dataset in the various metrics. By default, all methods generate 64×64 images along with the higher resolution 128×128 images. CMOTA outperforms existing methods by a large margin in both benchmarks in most of metrics.

The very recent method VP-CSV [2] performs *on par* to our CMOTA while ours still outperforms in FID, BLEU and R-precision, implying that CMOTA generates high quality images maintaining global semantic matching between story paragraph and images better than the VP-CSV. The lower performance than the VP-CSV in Char. F1 score and Frm. Acc. is attributed to the specialized character-centric module that only focuses the model to generate accurate characters

for each image at the expense of the other performance metrics. In addition, our CMOTA-HR model even outperforms the VP-CSV [2] by large margin without the specialized character-centric module.

Low resolution to high resolution. Not surprisingly, CMOTA-HR outperforms CMOTA with default (low-) resolution (i.e., 64×64) in almost all metrics. Interestingly, however, VLC-StoryGAN-HR [22] shows similar performance to 64×64 model in Frame accuracy, BLEU, R-precision, and drops performance in all other metrics. For the increased FID scores by the VLC-StoryGAN when goes to high resolution, we hypothesize that the constituency parse trees of input sentences in the VLC-StoryGAN increases the difficulty in generating visual details as the images contains more visual details in high resolution.

In addition, we observe that the performance drop of FID in both our CMOTA and VLC, when goes to high resolution. It is because as high-resolution image requires the model to depict fine-grained details, generation task would be more difficult compared to the low-resolution image generation. In particular, the drop of FID in Flintstones-SV is larger than that in Pororo-SV. It is because there are more complicated visual details in the Flintstones-SV than the Pororo-SV dataset. Note that the FID drops by our method from low-resolution to high resolution is less than the VLC in Pororo-SV. Furthermore, in terms of computational cost, our CMOTA gains performance boost by using high-resolution effectively with just adding relatively small computational cost on top of original 64×64 model.

Character consistency. Although our method does not explicitly enforce the character semantic preservation, the proposed memory delivers the semantic across the sentence implicitly. We here investigate the character context preservation performance by comparing our method to [22, 23] on per-character classification F1-score on test split of Pororo-

Dataset	Resolution	Methods	# Params.	FID↓	Char. F1↑	Frm. Acc.↑	BLEU-2/3↑	R-Prec.↑
Pororo-SV	64 × 64	StoryGAN [17]	-	158.06	18.59	9.34	3.24 / 1.22	1.51 ± 0.15
		CP-CSV [39]	-	140.24	21.78	10.03	3.25 / 1.22	1.76 ± 0.04
		DuCo-StoryGAN [23]	101M	96.51	38.01	13.97	3.68 / 1.34	3.56 ± 0.04
		VLC-StoryGAN [22]	100M	84.96	43.02	17.36	3.80 / 1.44	3.28 ± 0.00
		VP-CSV [2]	-	65.51	56.84	25.87	4.45 / 1.80	6.95 ± 0.00
		CMOTA (Ours)	96.6M	52.13	53.25	24.72	4.58 / 1.90	7.34 ± 0.03
Pororo-SV	128 × 128	VLC-StoryGAN-HR [†] [22]	102.6M	97.08	40.36	17.17	3.89 / 1.58	3.47 ± 0.03
		CMOTA-HR (Ours)	103.7M	52.77	58.86	28.89	5.45 / 2.34	16.36 ± 0.05
Flintstones-SV	64 × 64	StoryGAN [17]	-	127.19	46.20	32.96	13.87 / 7.83	1.72 ± 0.18
		DuCo-StoryGAN [23]	101M	78.02	54.92	36.34	15.48 / 9.17	2.64 ± 0.17
		VLC-StoryGAN [‡] [22]	100M	72.87	58.81	39.18	-	-
		CMOTA (Ours)	96.6M	36.71	79.74	66.01	19.85 / 12.98	10.50 ± 0.35
	128 × 128	CMOTA-HR (Ours)	103.7M	54.81	86.44	74.06	22.18 / 15.17	25.71 ± 0.70

Table 1. **Quantitative comparison with the state of the arts.** On the test split of Pororo-SV and Flintstones-SV. # Params. refers to the number of trainable parameters. Char. F1 refers to character F1 score. Frm. Acc. refers to frame accuracy. R-Prec. refers to R-Precision. ↓ indicates ‘lower the better’ and ↑ indicates ‘higher the better’. Experiments are done in both 64 × 64 and 128 × 128 resolutions. ‘HR’ refers to its high resolution (128 × 128) version. † indicates our results with author’s implementation (<https://github.com/adymaharana/VLCStoryGAN>). ‡ indicates the absolute values that we computed by the given relative values in [22].

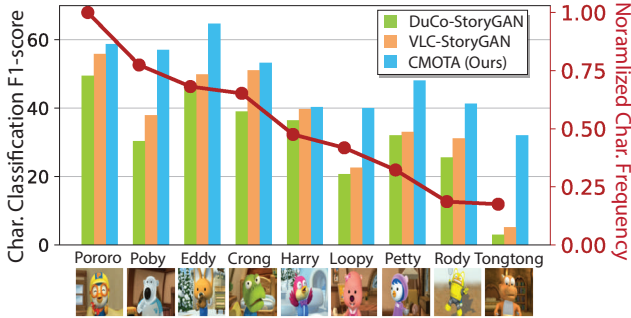


Figure 7. **Per-character Classification F1-score on the Test Split of Pororo-SV Dataset.** Normalized character (Char.) frequency is shown in red line. It represents the relative number of characters compared to the count of Pororo in training dataset. While DuCo/VLC-StoryGAN shows degraded performance as the character frequency decreases, CMOTA outperforms them across all the characters.

SV dataset and normalized character frequency in train split dataset as illustrated in Fig. 7. Especially, it is interesting to compare with [2] as it is explicitly proposed for that purpose.

As shown in the figure, our method outperforms prior arts [22, 23] by large margins. Although we did not have any special module for character information preserving, unlike [2], our model maintains such information over multiple sentences, even outperforming the image quality denoted as FID by -13.38 .

4.2.2 Ablation Studies

Benefit of the proposed architectural components. To investigate the benefit of each proposed architectural component, we build up the full model with the components from the vanilla, single-directional transformer without memory

as summarized in Tab. 2.

Using the proposed partial-level connected memory architecture only with the vanilla transformer (Tr.), the generation performance improves in all metrics compared to the Tr. Also, if we compare it with all-level connected memory architecture [15], PMA shows better performance in all metrics as in Tab. 3 (see supplementary material for more experimental analysis for various memory connection schemes). Employing attentional weighted memory (AWM) improves performance overall with slight degradation in FID (+0.12). The bi-directional learning scheme (Bi.) improves performance in all metrics except the character classification related metrics, *i.e.*, Char.F1 and Frm. Acc. It is because there are captions that does not contain characters’ name, just mentioning characters as ‘friends’, making the model difficult to generate specific characters.

Benefit of the text augmentation. By augmenting the generated text in offline manner (*i.e.*, using an image captioning model pretrained on the dataset of interest), we observe overall performance increase in all metrics. We hypothesize its reason as that the pseudo-text may convey contextual information such as characters’ existence or description of background, thereby making the generation model to be robust to language variation as shown in a first row of Tab. 4. By the online augmentation of pseudo texts (*i.e.*, gradually updating text generator, thus generating a diverse and gradually varying set of pseudo-texts), the performance further improves, as shown in Tab. 2 and Tab. 4.

4.2.3 Comparison with Large-Scale Models

Despite the unfairness of comparison to the large-scale models due to the model size, we compare CMOTA to larger

Methods	# Params.	FID↓	Char. F1↑	Frm. Acc.↑	BLEU-2/3↑	R-Prec.↑
VLC-StoryGAN [22]	100M	84.96	43.02	17.36	3.28 / 1.44	3.28
Single Directional Transformer (Tr.)	93.7M	63.88	45.48	18.44	4.18 / 1.69	5.67
+ Partial-level Memory Connection (PMA)	95.8M	59.05	49.72	21.79	4.41 / 1.77	6.28
+ Attentional Weighted Memory (AWM)	96.6M	56.98	50.15	22.01	4.50 / 1.91	6.79
+ Bi-directional Training (Bi.)	96.6M	54.69	51.27	22.15	4.52 / 1.84	7.12
+ Online Augmentation (Full model)	96.6M	52.13	53.25	24.72	4.58 / 1.90	7.34

Table 2. **Benefit of the proposed components.** On the test split of Pororo-SV Dataset. As a baseline, we use a single directional transformer-based text-to-image generation model. We gradually add each of proposed components to investigate its effect for improving performance, *i.e.*, partial-level memory connection (Sec. 3.1), attentively weighted memory (Sec. 3.1), bi-directional training (Sec 3.2), and online-augmentation (Sec. 3.2).

Memory Architecture	# Params.	FID↓	Char. F1↑	Frm. Acc.↑	BLEU-2/3↑	R-Prec.↑
All-Level Connection [15]	118M	61.23	47.21	19.21	4.21 / 1.72	6.08
Partial-level Connection (Ours)	95.8M	59.05	49.72	21.79	4.41 / 1.77	6.28

Table 3. **Benefit of the proposed memory connection scheme.** Proposed scheme outperforms conventional memory connection that uses all-level connections [15] (Fig. 4-(a)) with less number of parameters (test split of Pororo-SV).

Augmentation	# Params.	FID↓	Char. F1↑	Frm. Acc.↑	BLEU-2/3↑	R-Prec.↑
Offline	96.6M	54.51	51.32	22.31	4.50 / 1.90	7.09
Online (Ours)	96.6M	52.13	53.25	24.72	4.58 / 1.90	7.34

Table 4. **Benefit of the online augmentation.** On the Pororo-SV test split. The proposed online augmentation outperforms the offline augmentation using a pretrained captioner [15].

models [24] in Tab. 5. it is surprising to see that CMOTA outperforms on an evaluation metric of character classification (*i.e.*, Char.F1 and Frm. Acc.), regardless of using prompt-tuning or fine-tuning of StoryDALL-E.

But as expected, the image quality denoted as FID is worse than the StoryDALL-E thanks to the huge size of the model and training dataset. Nevertheless, FID score is slightly better (Pororo-SV) and on-par (Flintstones-SV) compared to ‘prompt-tune’ that updates 30% of parameters [24]. We discuss more about this in the supplementary material.

4.3. Human Preference Study

We conduct a larger scale (than the prior arts) human survey using the Amazon Mechanical Turk platform to qualitatively compare the generated image sequence by our method over VLC-StoryGAN [22] on three criteria, *i.e.*, visual quality, temporal consistency, and semantic relevancy between generated images and descriptions, following [17, 22, 23, 39]. Unlike the previous arts that employ 2 to 9 human judges [2, 17, 22, 23, 39], we employ 100 judges for statistically more reliable results.

As shown in the top table in Tab. 6, it is clear that human subjects prefer the generated images by our model over those from VLC-StoryGAN [22] on winning ratio(%). We discuss more details about this study in the supplementary material.

	Methods	# Params.	FID↓	Char. F1↑	Frm. Acc.↑
Pororo-SV	StoryDALL-E (prompt-tune)	1.3B	61.23	29.68	11.65
	StoryDALL-E (fine-tune)	1.3B	25.90	36.97	17.26
	MEGA-StoryDALL-E (fine-tune)	2.8B	23.48	39.91	18.01
	CMOTA (Ours)	96.6M	55.26	51.48	22.73
Flintstones-SV	StoryDALL-E (prompt-tune)	1.3B	53.71	42.38	32.54
	StoryDALL-E (fine-tune)	1.3B	26.49	73.43	55.19
	MEGA-StoryDALL-E (fine-tune)	2.8B	23.58	74.26	54.68
	CMOTA (Ours)	96.6M	58.59	79.75	62.98

Table 5. **Quantitative comparisons to large scale models.** On the test split of Pororo-SV and Flintstones-SV dataset. All models are implemented on top of the large pretrained transformer (*i.e.*, StoryDALL-E [24] pretrained on 14 million and MEGA-StoryDALL-E [24] pretrained on 15 million from Conceptual Caption dataset [36]). In [24], the ‘prompt-tune’ update 30% of model parameters compared to full ‘fine-tune’. Without pretraining, we use a much small model compared to the prior arts, *i.e.*, 96.6M (CMOTA) vs. 1.3B (StoryDALL-E).

Resolution	Attribute	VLC-SG [22]	Tie	Ours
64 × 64 (Low-Res.)	Visual Quality	27.8%	8.6%	63.6%
	Temporal Consistency	24.3%	8.3%	59.0%
	Semantic Relevance	31.2%	10.8%	57.9%
128 × 128 (High-Res.)	Visual Quality	21.9%	1.5%	76.6%
	Temporal Consistency	21.8%	2.5%	75.7%
	Semantic Relevance	23.6%	1.7%	74.6%
		CMOTA w/o Mem.	Tie	CMOTA
Temporal Consistency		34.5%	4.3%	61.2%

Table 6. **Human preference studies.** (Top) With 100 judges in the Amazon Mechanical Turk, on Pororo-SV test split dataset. Win (%) refers to the % times one model is preferred over the others. ‘Tie’ refers to the same. Ours are clearly preferred over the VLC-SG (refers to VLC-StoryGAN) [22]. (Bottom) For the model using our memory (Mem.) module, contrasting results with and without the memory module that considers the contexts spread across the sentences, our method significantly improves human preference in temporal consistency.

Human preference for memory usage. We further investigate the benefit of using the proposed memory module (Sec. 3.1) by the human study with 100 annotators in Amazon Mechanical Turk. For the comparative analysis between

CMOTA and CMOTA-w/o-Memory, we use 30 randomly sampled image sequences on Pororo-SV test set to conduct A/B test following [39]. Particularly, the annotators need to consider the temporal consistency with semantic relevance between generated image sequence and captions. As depicted in the bottom table in Tab. 6, we observe that 61.2% of annotators prefer our CMOTA with memory module over the one without it. We can conclude that the memory module has a great deal in creating temporally consistent image sequence with respect to semantic relevancy.

4.4. Qualitative Analysis

We now qualitatively analyze the quality of generated image sequences in Fig. 8 on the Pororo-SV dataset’s test split. The top row shows the image sequence of ground truth, the two rows (2-3) contain prior works [17, 23] and the final row is the image sequence generated by CMOTA and its high resolution version (CMOTA-HR). CMOTA generates a semantically more plausible image sequence with better visual quality, compared to prior works [22, 23]. Particularly, the CMOTA-HR demonstrates the ability to generate visual details more effectively even as the size of the image increases, while maintaining similar semantics.

Furthermore, we qualitatively investigate the advantage of using memory architecture and summarize the results in Fig. 9. As shown in the figure, the proposed memory architecture generates a semantically more plausible image sequence with proper context, *e.g.*, background; the CMOTA without memory fails to capture proper background context (shown in dotted red box) since a single sentence could be interpreted in many ways. In contrast, CMOTA (with the context memory) generates an image sequence with plausible background with characters preserved without an explicit character-centric model.

5. Conclusion

We propose to better encode semantic context, *e.g.*, plausible background and characters, for story visualization using a new memory architecture in a multi-modal bi-directional Transformer. We further propose an online text augmentation training scheme to generate pseudo-text descriptions as an intermediate supervision for addressing linguistic diversity in the texts at inference. The proposed method generates a temporally coherent and semantically relevant image sequence for each sentence in the given text paragraph.

The proposed method outperforms prior works by a large margin on various metrics on the two popular SV benchmark datasets, and also outperforms some of hyper-scale models in multiple semantic understanding metrics. Although computationally prohibited and the pre-trained models are not publicly available, it would be intriguing to apply the proposed memory module and the online augmentation scheme to a large model.

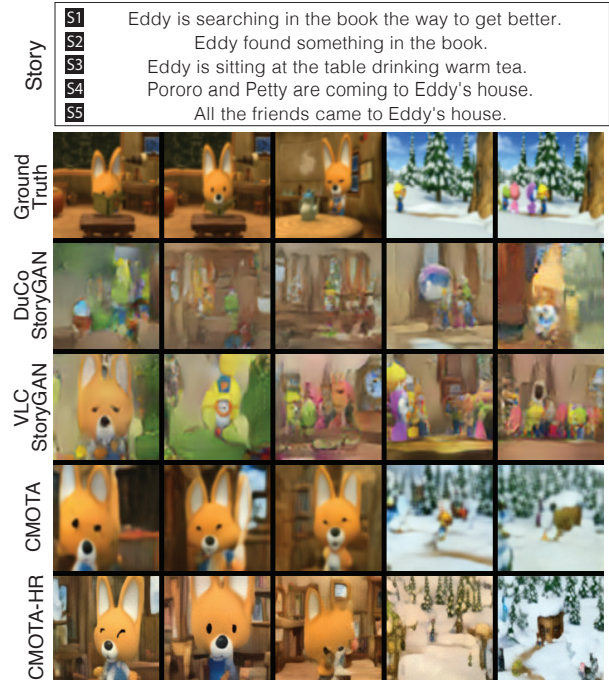


Figure 8. **Qualitative results compared to prior arts.** On the Pororo-SV’s test split. We compare our CMOTA to the prior arts including DuCo-StoryGAN and VLC-StoryGAN. Ours generates a semantically more plausible and temporally coherent image sequence compared to the prior arts. All images except CMOTA-HR (128×128) are generated with resolution of 64×64 for comparison.

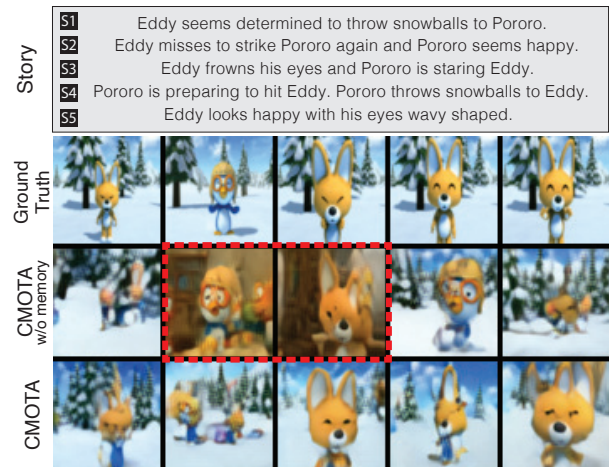


Figure 9. **Benefit of memory module in encoding background context.** Our CMOTA generates both semantically and visually plausible image sequences. Without memory module, inconsistent images (*i.e.*, abrupt background change) are generated as indicated by the red dotted box.

Acknowledgement. This work is partly supported by the NRF grant (No.2022R1A2C4002300) 25%, IITP grants (No.2020-0-01361, AI GS Program (Yonsei University) 5%, No.2021-0-02068, AI Innovation Hub 5%, 2022-0-00077 15%, 2022-0-00113 15%, 2022-0-00959 15%, 2022-0-00871 10%, 2022-0-00951 10%) funded by the Korea government (MSIT).

References

- [1] Alexander Buslaev, Vladimir I. Iglovikov, Eugene Khvedchenya, Alex Parinov, Mikhail Druzhinin, and Alexandr A. Kalinin. Albumentations: Fast and flexible image augmentations. *Information*, 11(2), 2020. 3
- [2] Hong Chen, Rujun Han, Te-Lin Wu, Hideki Nakayama, and Nanyun Peng. Character-centric story visualization via visual planning and token alignment. *ArXiv*, 2022. 2, 5, 6, 7, 8
- [3] Mark Chen, Alec Radford, Jeff Wu, Heewoo Jun, Prafulla Dhariwal, David Luan, and Ilya Sutskever. Generative pre-training from pixels. In *ICML*, 2020. 2, 3
- [4] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. Transformer-XL: Attentive language models beyond a fixed-length context. In *ACL*, 2019. 3, 4
- [5] Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, and Jie Tang. Cogview: Mastering text-to-image generation via transformers. In *Advances in Neural Information Processing Systems*, 2021. 2, 3
- [6] Ming Ding, Wendi Zheng, Wenyi Hong, and Jie Tang. Cogview2: Faster and better text-to-image generation via hierarchical transformers. *ArXiv*, 2022. 2, 3
- [7] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors. *arXiv*, 2022. 2, 3
- [8] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors. In *ECCV*, 2022. 2
- [9] Tanmay Gupta, Dustin Schwenk, Ali Farhadi, Derek Hoiem, and Aniruddha Kembhavi. Imagine this! scripts to compositions to videos. In *ECCV*, 2018. 2, 5, 6
- [10] Danna Gurari, Qing Li, Chi Lin, Yinan Zhao, Anhong Guo, Abigale Stangl, and Jeffrey P. Bigham. Vizwiz-priv: A dataset for recognizing the presence and purpose of private visual information in images taken by blind people. In *CVPR*, June 2019. 5
- [11] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *ArXiv*, 2022. 2
- [12] Philip T. G. Jackson, Amir Atapour-Abarghouei, Stephen Bonner, T. Breckon, and Boguslaw Obara. Style augmentation: Data augmentation via style randomization. In *CVPR Workshops*, 2019. 3
- [13] Doyeon Kim, Donggyu Joo, and Junmo Kim. Tivgan: Text to image to video generation with step-by-step evolutionary generator. *IEEE Access*, 2020. 2
- [14] Taehoon Kim, Gwangmo Song, Sihaeng Lee, Sangyun Kim, Yewon Seo, Soonyoung Lee, Seung Hwan Kim, Honglak Lee, and Kyunghoon Bae. L-verse: Bidirectional generation between image and text. In *CVPR*, 2022. 3
- [15] Jie Lei, Liwei Wang, Yelong Shen, Dong Yu, Tamara L Berg, and Mohit Bansal. Mart: Memory-augmented recurrent transformer for coherent video paragraph captioning. In *ACL*, 2020. 3, 4, 7, 8
- [16] 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. 5
- [17] Yitong Li, Zhe Gan, Yelong Shen, Jingjing Liu, Yu Cheng, Yuexin Wu, Lawrence Carin, David Carlson, and Jianfeng Gao. Storygan: A sequential conditional gan for story visualization. In *CVPR*, 2019. 1, 2, 3, 5, 6, 7, 8, 9
- [18] Yitong Li, Martin Renqiang Min, Dinghan Shen, David Carlson, and Lawrence Carin. Video generation from text. In *AAAI*, 2018. 2
- [19] Swee Kiat Lim, Yi Loo, Ngoc-Trung Tran, Ngai-Man Cheung, Gemma Roig, and Yuval Elovici. Doping: Generative data augmentation for unsupervised anomaly detection with gan. *2018 IEEE International Conference on Data Mining (ICDM)*, pages 1122–1127, 2018. 3
- [20] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 5
- [21] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 6
- [22] Adyasha Maharana and Mohit Bansal. Integrating visuospatial, linguistic and commonsense structure into story visualization. In *EMNLP*, 2021. 2, 5, 6, 7, 8, 9
- [23] Adyasha Maharana, Darryl Hannan, and Mohit Bansal. Improving generation and evaluation of visual stories via semantic consistency. In *NAACL-HLT*, 2021. 2, 5, 6, 7, 8, 9
- [24] Adyasha Maharana, Darryl Hannan, and Mohit Bansal. Storydall-e: Adapting pretrained text-to-image transformers for story continuation. *ArXiv*, 2022. 2, 5, 8
- [25] Gaurav Mittal, Tanya Marwah, and Vineeth N. Balasubramanian. Sync-draw: Automatic video generation using deep recurrent attentive architectures. In *ACM Multimedia*, 2017. 2
- [26] Saypraseuth Mounsaveng, Issam Laradji, Ismail Ben Ayed, David Vazquez, and Marco Pedersoli. Learning data augmentation with online bilevel optimization for image classification. *arXiv*, 2020. 3
- [27] Yingwei Pan, Zhaofan Qiu, Ting Yao, Houqiang Li, and Tao Mei. To create what you tell: Generating videos from captions. In *ACM Multimedia*, 2017. 2
- [28] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *CVPR*, 2019. 4
- [29] Nikolaos Passalis and Anastasios Tefas. Learning deep representations with probabilistic knowledge transfer. In *ECCV*, 2018. 4
- [30] Tingting Qiao, Jing Zhang, Duanqing Xu, and Dacheng Tao. Learn, imagine and create: Text-to-image generation from prior knowledge. In *NeurIPS*, 2019. 3
- [31] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv*, 2022. 2, 3
- [32] 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. 2, 3

- [33] Scott E. Reed, Zeynep Akata, Xinchun Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. In *ICML*, 2016. 3
- [34] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L. Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, Seyedeh Sara Mahdavi, Raphael Gontijo Lopes, Tim Salimans, Jonathan Ho, David Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. 2022. 3
- [35] Meet Shah, Xinlei Chen, Marcus Rohrbach, and Devi Parikh. Cycle-consistency for robust visual question answering. *CVPR*, 2019. 3
- [36] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*, 2018. 8
- [37] Oleksii Sidorov, Ronghang Hu, Marcus Rohrbach, and Amanpreet Singh. Textcaps: a dataset for image captioning with reading comprehension. In *ECCV*, 2020. 5
- [38] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, Devi Parikh, Sonal Gupta, and Yaniv Taigman. Make-a-video: Text-to-video generation without text-video data. *ArXiv*, 2022. 2
- [39] Yun-Zhu Song, Zhi Rui Tam, Hung-Jen Chen, Huiao-Han Lu, and Hong-Han Shuai. Character-preserving coherent story visualization. In *ECCV*, 2020. 2, 5, 7, 8, 9
- [40] Zhiqiang Tang, Yunhe Gao, Leonid Karlinsky, Prasanna Sattigeri, Rogerio Feris, and Dimitris Metaxas. Online augmentation: Online data augmentation with less domain knowledge. *arXiv*, 2020. 3
- [41] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. In *ICLR*, 2020. 4
- [42] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *NIPS*, 2017. 3, 5
- [43] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017. 2, 3, 4
- [44] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *CVPR*, June 2015. 5
- [45] Ruben Villegas, Mohammad Babaeizadeh, Pieter-Jan Kindermans, Hernan Moraldo, Han Zhang, Mohammad Taghi Saffar, Santiago Castro, Julius Kunze, and Dumitru Erhan. Phenaki: Variable length video generation from open domain textual description. *arXiv*, 2022. 2
- [46] Chenfei Wu, Lun Huang, Qianxi Zhang, Binyang Li, Lei Ji, Fan Yang, Guillermo Sapiro, and Nan Duan. Godiva: Generating open-domain videos from natural descriptions, 2021. 2
- [47] Chenfei Wu, Jian Liang, Lei Ji, F. Yang, Yuejian Fang, Daxin Jiang, and Nan Duan. Nüwa: Visual synthesis pre-training for neural visual world creation. In *ECCV*, 2022. 2
- [48] Zhiyue Wu, Yijie Wang, Haibo Mi, Hongzuo Xu, Wei Zhang, and Lanlan Feng. Oada: An online data augmentation method for raw histopathology images. *ICONIP*, 2021. 3