

## A Acknowledgement

This work is funded by the CALCULUS project (European Research Council Advanced Grant H2020-ERC-2017-ADG 788506) and the Flanders AI Research Program.

## B Boarder Impacts

Visual generative techniques, particularly text-to-image (T2I) models, hold significant potential for producing coherent visual content across diverse scenarios, making them highly applicable to downstream tasks such as storytelling and personalized content creation. One of the most challenging aspects in this domain is the consistent synthesis of characters across varying contexts—a problem that existing methods continue to struggle with, as discussed in this paper. Our proposed approach addresses this challenge by effectively balancing subject consistency and prompt fidelity, allowing users to generate coherent story sequences featuring the same character while closely adhering to their provided descriptions. In addition, our exploration of the unique structure of zigzag sampling introduces a new perspective on its utility in diffusion-based generation, offering valuable insights that may inspire future research into more controllable and semantically aware generative models.

The application of text-to-image models in visual storytelling, while creatively empowering, introduces significant ethical, privacy, and security risks. A major concern is the non-consensual creation of fictional narratives featuring real individuals, where realistic and consistent character generation enables the fabrication of defamatory, misleading, or harmful visual stories—such as fake memoirs, satirical comics, or illustrated scenarios depicting private citizens or public figures in compromising situations—without their knowledge or consent. The very capability to maintain character coherence across scenes, which enhances narrative immersion, can be exploited to produce persuasive, long-form synthetic content that blurs the line between fiction and reality, facilitating disinformation campaigns or emotional manipulation. Furthermore, privacy is compromised when models trained on unconsented web-scraped data generate characters closely resembling real people, effectively creating digital doppelgängers embedded in fictional universes. These capabilities also pose security threats, as coherent AI-generated visual stories can be weaponized for influence operations, identity exploitation, or viral misinformation, undermining trust and personal autonomy. Without robust safeguards—such as provenance tracking, consent filters, and transparent content policies—AI-driven visual storytelling risks enabling large-scale narrative abuse with profound societal consequences.

## C Usage of LLMs

We only use large language models (LLMs) for writing assistance, such as correcting grammar, fixing typos, and improving clarity.

## D Implementation Details

We implement our method on two open-source models: SDXL and FLUX. For SDXL, we use the *stabilityai/stable-diffusion-xl-base-1.0* version, and for FLUX, we adopt the *black-forest-labs/FLUX.1-dev* version. All baseline methods—including IP-Adapter [24], Consistory [28], StoryDiffusion [27], and IPrompt1Story [29]—are reproduced using their official open-source implementations with default hyperparameters. Since IP-Adapter is designed for image-conditional generation, we adapt it to our setting by first generating an identity image using SDXL with the given identity prompt. This generated image is then used as the conditioning input for IP-Adapter to produce images guided by different prompts.

For the implementation of our method on the SDXL model, we cache visual tokens only from the mid and upper layers across all steps. Accordingly, feature injection during the zig step is also limited to these layers. We use a classifier-free guidance scale of 5.5 for both the zig and generation steps, and set it to 0 during the zag step. All experiments are conducted on a single NVIDIA A100 GPU.

For the FLUX model, which differs architecturally from SDXL by separating text-image and image-image interaction stages, we adopt a different strategy. FLUX begins with several layers of text-image interaction, followed by layers of purely image-based interaction. To cache visual tokens, we first average the attention scores from the early text-image interaction layers to identify subject-relevant

visual features. These selected tokens are then used for feature injection across all image-image interaction layers. Experiments for FLUX are also run on a single NVIDIA A100 GPU.

---

**Algorithm 1** Identity Visual Token Cache (Subject token extraction & top- $k$  selection)

---

**Require:** identity prompt  $P_{\text{id}}$ , pretrained diffusion model  $\epsilon_\theta$ , text encoder  $\tau_\zeta$ , time steps  $\{m\}$  (used for identity extraction), layers  $\mathcal{L}$ , top- $k$  ratio  $k$

**Ensure:** cached key tokens  $\mathcal{I}^{\text{key}} = \{I_{\text{key}}^{\ell,m}\}$  and value tokens  $\mathcal{I}^{\text{value}} = \{I_{\text{value}}^{\ell,m}\}$

- 1: Compute text embedding  $T_{\text{id}} \leftarrow \tau_\zeta(P_{\text{id}})$ .
- 2: **for** each layer  $\ell \in \mathcal{L}$  and timestep  $m$  used for identity extraction **do**
- 3:   Run denoising pass (or inference pass) with prompt  $P_{\text{id}}$  to obtain intermediate visual tokens at  $(\ell, m)$ .
- 4:   Compute text-image attention scores  $S^{\ell,m} = \text{AttentionScores}(\text{visual tokens}, T_{\text{id}})$ .
- 5:   Identify top- $k$  indices by score:  $J^{\ell,m} \leftarrow \text{TopKIndices}(S^{\ell,m}, k)$ .
- 6:   Extract corresponding key / value projections:

$$I_{\text{key}}^{\ell,m} \leftarrow \{i_{\text{key},j}^{\ell,m} : j \in J^{\ell,m}\}, \quad I_{\text{value}}^{\ell,m} \leftarrow \{i_{\text{value},j}^{\ell,m} : j \in J^{\ell,m}\}.$$

- 7:   Store  $I_{\text{key}}^{\ell,m}$  into  $\mathcal{I}^{\text{key}}$  and  $I_{\text{value}}^{\ell,m}$  into  $\mathcal{I}^{\text{value}}$ .
  - 8: **end for**
  - 9: **return**  $\mathcal{I}^{\text{key}}, \mathcal{I}^{\text{value}}$
- 

---

**Algorithm 2** Asymmetric Zigzag Sampling with Zig Visual Sharing (ZVS) & Asymmetric Prompt Zigzag Inference (APZI)

---

**Require:** target prompt  $P$ , identity token caches  $\mathcal{I}^{\text{key}}, \mathcal{I}^{\text{value}}$  (from Alg. 1), diffusion model  $\epsilon_\theta$ , text encoder  $\tau_\zeta$ , full zigzag time schedule  $t = T, \dots, 1$ , and its noise coefficients  $\alpha_t$

**Ensure:** latent  $x_0$  (decoded to image by decoder  $D$ )

- 1: Compute full prompt embedding  $T \leftarrow \tau_\zeta(P)$  (can use one-prompt fusion / reweighting as in paper).
- 2: Initialize noisy latent  $x_T \sim \mathcal{N}(0, I)$ .
- 3: **for** each diffusion step index  $t = T, T-1, \dots, 1$  **do**
- 4:   **Zig step (forward denoise + visual injection):**
  1. Compute standard denoising prediction  $\hat{\epsilon} \leftarrow \epsilon_\theta(x_t, t, T)$  using prompt embedding  $T$ .
  2. Perform denoising update (forward) to get intermediate latent  $x_{t-1}$
  3. **Inject identity visual tokens** into self-attention layers: for each layer  $\ell$  used,

$$K^{\ell,+} \leftarrow \text{Concat}(I_{\text{key}}^{\ell,m}, K^\ell), \quad V^{\ell,+} \leftarrow \text{Concat}(I_{\text{value}}^{\ell,m}, V^\ell),$$

where  $K^\ell, V^\ell$  are the current layer key/value projections and  $I_{\cdot}^{\ell,m}$  come from the caches. Keep queries unchanged. (Apply only in Zig.)

- 5:   **Zag step (inverse denoise — null prompt / no text guidance):**
  1. Use null prompt.
  2. Perform inverse denoising mapping to propagate the injected identity into the noisy latent:

$$\tilde{x}_t \leftarrow \text{InverseDenoise}(x_{t-1}, t-1 \rightarrow t, \epsilon_\theta, \text{null prompt}),$$

3. (No visual injection in Zag.)
- 6:   **Generation step (final denoise with text guidance):**
  1. Use full prompt embedding  $T$  to compute  $\epsilon$ -prediction on  $\tilde{x}_t$ .
  2. Perform forward denoising to obtain  $x_{t-1}^{\text{final}}$  (standard step).
  3. Set  $x_{t-1} \leftarrow x_{t-1}^{\text{final}}$  and continue.

- 7: **end for**
  - 8: Decode  $x_0$  to image:  $I \leftarrow D(x_0)$  and return  $I$ .
-



## E More Details about User Study

We conducted a user study to evaluate our method in comparison with four existing approaches: IP-Adapter [24], Consistory Model [28], Story Diffusion [27], and 1Prompt1Story [29]. All models were used to generate images based on prompts from the *ConsiStory+* benchmark, using the same random seeds as reported in their respective papers to ensure a fair comparison.

From the generated dataset, we randomly selected 30 prompts, each associated with 4 images. For each participant in the user study, the system randomly sampled 20 out of these 30 prompts to form an evaluation set. Before starting the evaluation, users were briefed on three key criteria:

- **Identity Consistency:** Measures whether the same character or subject appears consistently across all images for a given prompt.
- **Prompt Alignment:** Assesses how well each image reflects the content and intent of the original text prompt.
- **Image Quality:** Evaluates the overall visual quality, including clarity, detail, and aesthetic appeal.

To minimize potential bias, the presentation order of the five methods was randomized for each question in the study interface. Figure 10 presented the user interface of our user study system.

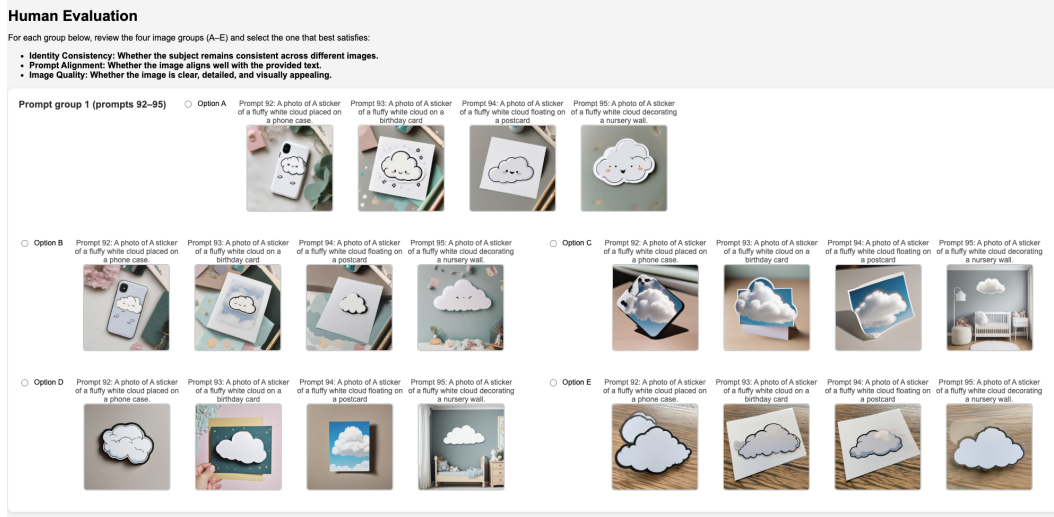


Figure 7: A visualization of the user interface used in our user study system.

## F More Visualizations



Figure 8: More visualizations of results generated using FLUX model.





Figure 9: More visualizations of results generated using SDXL model.



## G Long Story Visualizations

An anime-style illustration of a 16 years old girl with wavy chestnut hair, a slender frame, and soft brown eyes



Figure 10: Long story generated using FLUX model