MANGA2ANIMATE: AN AGENTIC FRAMEWORK FOR FAITHFUL MANGA ANIMATION WITH SCRIPT AND STORYBOARD PLANNING

Anonymous authors
Paper under double-blind review

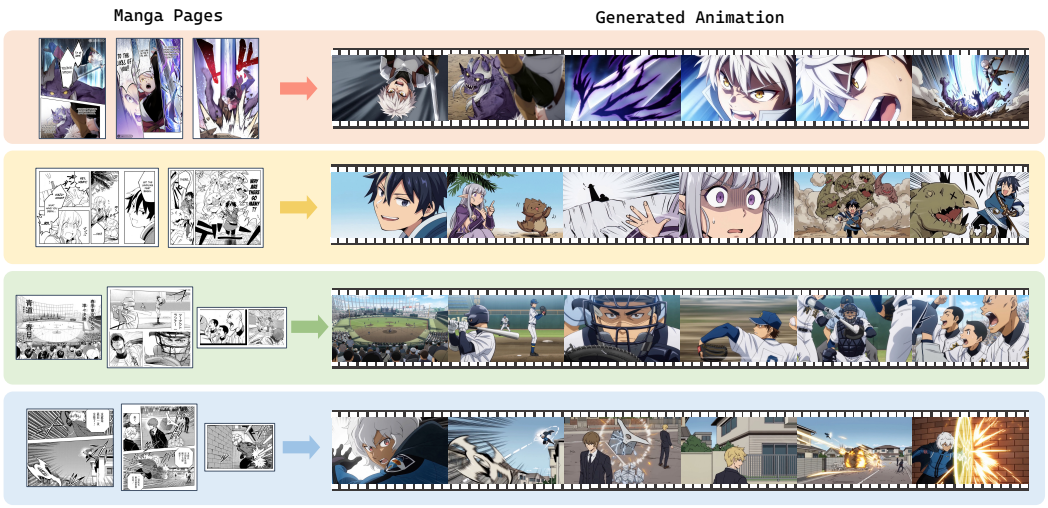


Figure 1: **Manga2Animate turns existing manga pages into coherent animation that stays faithful to the original manga.** Given a few manga pages *(left)*, our multi-agent framework first builds an animation script and a visual storyboard as intermediate representations, iteratively refining both through creator-reviewer collaboration, and then generates the final animation from them *(right)*. The resulting animation preserves the manga's plot, character identity, and art style, while flowing coherently from clip to clip.

ABSTRACT

Adapting existing manga into coherent animation that stays faithful to the source is a valuable yet challenging task. Existing methods either access the manga mainly through text, losing its characters and art style, or feed manga pages directly into video generation models, which misinterpret the manga's structure and produce incorrect storylines. In this paper, we propose **Manga2Animate**, a multi-agent framework that grounds every stage of generation in the source manga. We decompose the task into two intermediate representations, an animation script and a visual storyboard. Each representation is produced by a creator agent and reviewed against the source panels by a paired reviewer agent. We tailor representation-specific selection and dual-level feedback to progressively refine both before synthesizing the final animation. Extensive experiments and a user study demonstrate that Manga2Animate produces animations more faithful to the source manga and more consistent across shots than state-of-the-art approaches.

1 INTRODUCTION

Manga is one of the richest sources of stories for animation. Popular titles are routinely adapted into anime, and fans expect these adaptations to move smoothly and tell the story coherently while remaining faithful to the characters and art style they love. Yet producing such an adaptation remains slow and costly, often taking large teams of artists months of storyboarding, key animation, in-betweening, and coloring.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

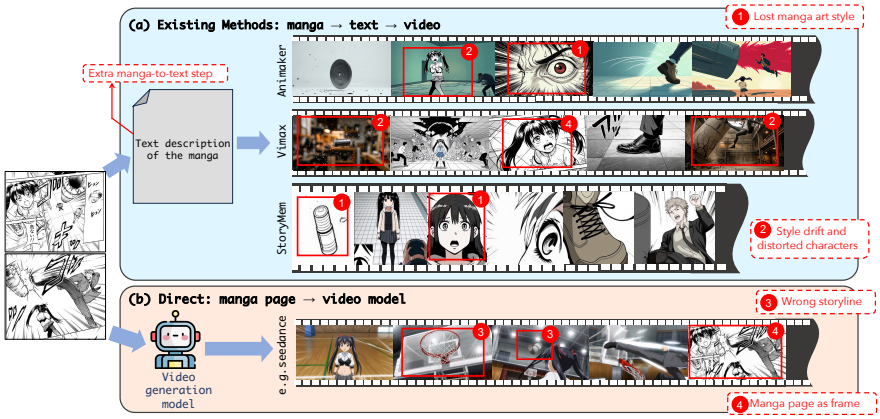


Figure 2: **Limitations of existing methods to manga-to-animation generation.** (a) Text-based storytelling methods (AniMaker, ViMax, StoryMem) access the manga mainly through text, losing its art style (1) or distorting characters (2). (b) Feeding manga pages into a video model (e.g., Seedance 2.0) tells the wrong story (3) and shows manga panels as video frames (4).

At the same time, a manga already specifies much of what an animation needs, including who the characters are and what they look like, how each shot is framed, and how the plot unfolds. With recent advances in video generation, it is natural to ask whether a manga can be brought to life automatically. However, as Fig. 2 shows, storytelling systems that first describe the manga in words discard its visual appearance, so the original art style is lost (1) and characters change appearance or drift between anime and photorealistic styles (2). Feeding manga pages directly into a video model fails for a different reason. The model knows neither the order in which the panels should be read nor what speech balloons mean, so it tells the wrong story (3) or even places manga panels, or entire pages, directly into the video as frames (4).

Our key observation is that today's video generation models already know how to animate. What they cannot do is understand a manga. Given a manga page, they produce smooth and appealing motion, yet they follow the panels in the wrong order and misread the speech balloons, so the result looks convincing but tells a completely wrong story (Fig. 2b). We therefore do not replace these models. Instead, we keep their rich visual knowledge and give them what they are missing, a clear understanding of the manga. Much like an animation studio, Manga2Animate (Fig. 1) first turns the manga into an animation script, which describes what happens in each panel, and a storyboard, which shows how each shot should look. Both are built by AI agents and checked against the original panels, so the video model receives a faithful plan instead of raw manga pages. We find that this plan effectively constrains generation. With the script fixing the plot and the storyboard fixing the characters, composition, and art style, the video model no longer has to interpret the manga on its own and can focus on what it does best, bringing the story to life.

Experiments show that Manga2Animate produces animations that are more faithful to the original characters, art style, and plot than existing storytelling and video generation methods, even when these methods are given the same script as ours. Because our framework guides existing models rather than replacing them, it works with both open-source and commercial image and video generators, and outperforms prior methods with either. In a user study, participants selected our results as the best overall in over 90% of responses, for both storyboards and videos.

In summary, our contributions are threefold:

- We present Manga2Animate, to the best of our knowledge the first automated framework that turns existing manga into animation that is faithful to source manga.
- We show that planning through a script and a storyboard, iteratively refined with dual-level feedback from both and verified against the original manga, is key to preserving the characters, art style, and plot.
- We establish an evaluation protocol for manga-to-animation generation, under which Manga2Animate outperforms existing methods, and we further validate these results with a user study.

2 Related Work

Manga understanding. Built on annotated corpora (Matsui et al., 2017; Aizawa et al., 2020; Vivoli et al., 2024a; Li et al., 2024a), manga understanding has progressed from detecting panels, characters, and text (Shen et al., 2023) to ordering panels and attributing dialogue (Sachdeva & Zisserman, 2024), naming characters across chapters (Sachdeva et al., 2024), and narrating chapters as prose (Sachdeva & Zisserman, 2025). Vision-language models are now benchmarked on manga (Baek et al., 2026; Ikuta et al., 2025; Vivoli et al., 2025), yet still misread cross-panel narrative (Baranwal et al., 2025), and generation from comics remains largely unexplored (Vivoli et al., 2024b). We use Sachdeva & Zisserman (2024) for pre-processing and, given this fragility, review every intermediate output against the source panels.

Story visualization. Story visualization renders a narrative as a consistent image sequence, evolving from GANs (Li et al., 2019) to diffusion models with visual memory (Rahman et al., 2023), identity conditioning (Ye et al., 2023), shared attention (Tewel et al., 2024; Zhou et al., 2024), frame context (Liu et al., 2024a), multi-subject personalization (He et al., 2025b; Wang et al., 2025b; Wu et al., 2025b), iterative refinement (Mao et al., 2026), and language-model layout planning (He et al., 2025a; Wang et al., 2025a; Cheng et al., 2024), with dedicated benchmarks (Zhuang et al., 2026; Gao et al., 2025) and extensions to manga generation (Wu et al., 2025a; Wang et al., 2026a). These methods generate from text, whereas our storyboard is conditioned on and selected against the manga panels.

Multi-shot video generation. Video diffusion has advanced from short clips (Ho et al., 2022) to foundation models (Kong et al., 2024; Wan et al., 2025; Yang et al., 2025; Polyak et al., 2024; Seedance et al., 2026). Cross-shot consistency is pursued through long-context tuning (Guo et al., 2025; Xiao et al., 2026; Meng et al., 2026), keyframe memory (Zhang et al., 2025a; Zhou et al., 2026), storyboard anchoring (Zhang et al., 2026), and reference-based identity injection (Jiang et al., 2025; Liu et al., 2025a; Hu et al., 2025). For animation, anime-oriented models and datasets (Jiang et al., 2024; Qiu et al., 2025) and in-betweening and colorization methods (Xing et al., 2024; Meng et al., 2025; Huang et al., 2024; Li et al., 2025; Liu et al., 2025b; Tang et al., 2025) create motion and color from sparse inputs. We use such models as interchangeable backends and obtain consistency from the reviewed script and storyboard rather than from architecture (Sec. 4.4).

Agentic storytelling video generation. Language and vision-language agents increasingly orchestrate video production from script to storyboard, clips, and assembly (Lin et al., 2023; Li et al., 2024b; Xu et al., 2025a; Wu et al., 2025c; Hu et al., 2024; Xu et al., 2025b; Wang et al., 2026b; Huang et al., 2026), some with critics that select clips by tree search (Shi et al., 2025), gate anime production stages (Zhang et al., 2025b; Yan et al., 2026), or self-improve at test time (Long et al., 2026), following self-feedback in language agents (Madaan et al., 2023; Shinn et al., 2023). Our creator–reviewer design is closest to these systems, but they invent characters, shots, and plot from a prompt and can only check outputs against their own script, whereas our reviewers check against the manga panels. Earlier manga-to-animation systems only moved a virtual camera over panels or added synthesized speech (Cao et al., 2016; Gupta et al., 2021); to our knowledge, ours is the first to generate faithful multi-shot animation from manga pages.

3 Methodology

3.1 Task Formulation

Given a set of manga pages $M = \{m_1, \ldots, m_p\}$ in their original reading order, our goal is to generate a coherent animated video V. Each manga page consists of multiple *manga panels*, where a panel is an individual visual frame depicting a particular moment, action, or scene. From the input pages, we obtain an ordered sequence of N_p target manga panels $P = \{p_1, \ldots, p_{N_p}\}$ ($N_p = 6$ in our experiments) and a transcript T. For prominent characters appearing in the selected manga panels, we additionally use character reference images R. These references are taken from official color artwork when available; otherwise, clear appearances of the characters are cropped from the input manga.

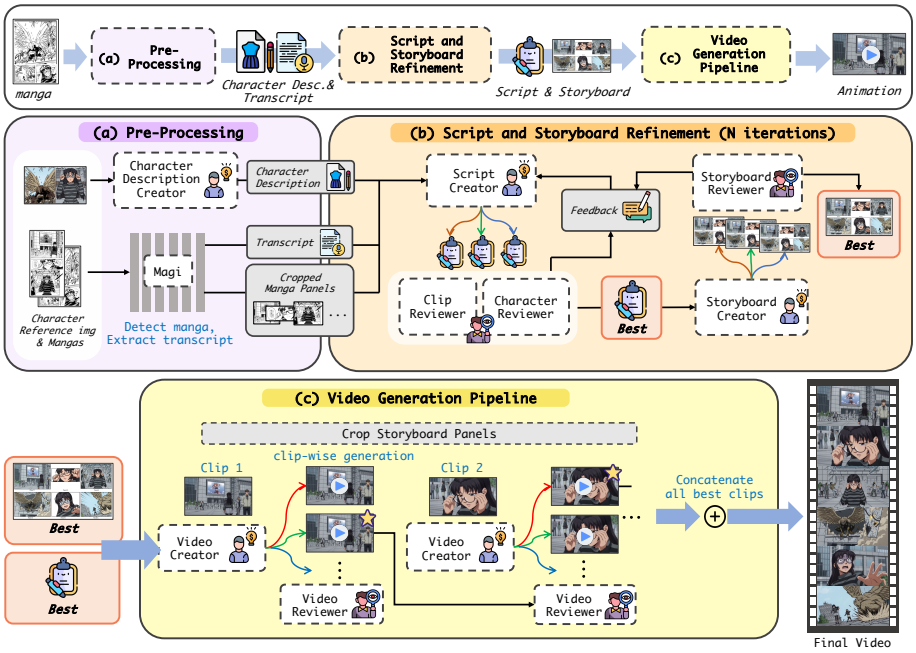


Figure 3: **Overview of the Manga2Animate pipeline.** (a) Pre-processing extracts ordered panels, a transcript, and character descriptions. (b) Creator and reviewer agents refine an animation script and a visual storyboard over N rounds and select the best script and best storyboard, dual-level feedback also guides the next round. (c) Video generation runs in first-frame or reference-conditioned mode.

Our pipeline transforms these inputs through two intermediate representations: an animation script $S = \{s_1, \ldots, s_{N_p}\}$ and a visual storyboard $B = \{b_1, \ldots, b_{N_p}\}$, before producing the final video V, as illustrated in Fig. 3. We also derive per-character descriptions D from the reference images R.

The use of S and B is inspired by real-world animation production workflows, where the narrative is first organized into a script and subsequently translated into a storyboard before animation is produced. Following this workflow, S serves as the textual specification of the animation. Each script clip s_i describes the content corresponding to manga panel p_i, including the scene, camera, character positions and actions, expressions, and dialogue. The storyboard B then converts this textual specification into a visual plan, where each storyboard panel b_i specifies the appearance and scene composition of the corresponding animated clip.

Throughout the pipeline, we maintain an explicit correspondence between the source manga and the generated representations as illustrated in Fig. A1: manga panel p_i, script clip s_i, and storyboard panel b_i all describe the same narrative unit. This alignment allows each generated component to be directly traced back to its source manga panel during review and refinement.

3.2 Agents

Manga2Animate uses two types of agents at each stage of the pipeline (Fig. 3).

Creators. A creator takes the inputs of its stage and produces the intermediate output of that stage.

Reviewers. A reviewer reviews the outputs of its creator, selects the best one, and gives feedback for refinement. Full review criteria and prompts are listed in Appendices G and H.

3.3 Pre-processing

Manga pages often exhibit non-trivial panel layouts, making reading order an important prerequisite for preserving the narrative flow. We therefore detect the panels on each page and deter-

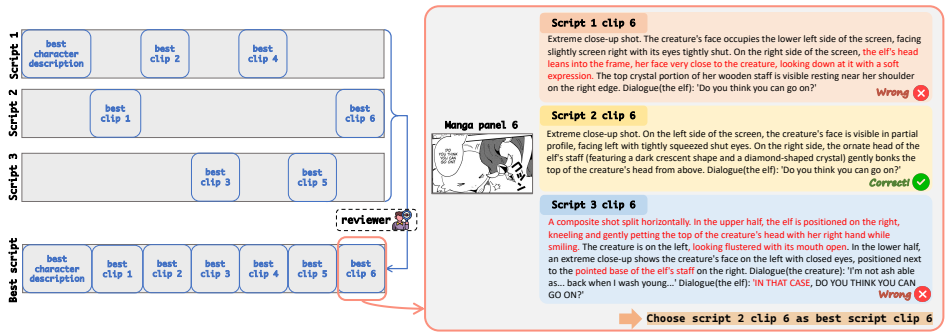


Figure 4: **Compositional (clip-wise) selection by the Script Reviewer.** The reviewer selects the best character and clip description at each panel position across the N_s candidate scripts and reassembles them into one best script (left). For one panel (right), it flags inaccurate details (red) in two candidates and selects the one matching the panel (green).

mine their reading order using a manga-specific detection model and its associated ordering algorithm (Sachdeva & Zisserman, 2024). The resulting ordered panel sequence is then passed to the subsequent stages of the pipeline. In parallel, the same model also detects and transcribes the text on these pages, which we merge into a single transcript spanning all pages to help the Script Creator write each panel's dialogue and plot (Sec. 3.4). Also, we use a Character Description Creator to establish a consistent representation of each character from the reference image R. It extracts relatively stable, identity-defining traits, such as facial features and body build, while excluding variable attributes such as clothing and accessories. These character descriptions are then provided to the downstream Script and Storyboard Creators to help maintain character consistency throughout the generation process.

3.4 Script and Storyboard Refinement

To improve generation reliability while controlling API costs, we combine a small number of candidates per stage with iterative refinement, in which reviewer feedback guides the next round of generation. We run $N = 2$ refinement rounds: the first round explores initial candidates, and the second round revises them using the first-round feedback.

In each refinement round, the Script Creator generates N_s script candidates ($N_s = 3$) conditioned on the manga panels, character descriptions, transcript, and, from the second round onward, the feedback from the previous round. Errors in scripts are largely local, a candidate may describe one panel precisely yet misread another, and different candidates tend to make different mistakes. Selecting a single script as a whole would discard the correct descriptions contained in the other candidates. The Script Reviewer therefore performs compositional selection at the clip level (Fig. 4), preserving complementary strengths across candidates, and records the remaining weaknesses of each selected component as feedback for the next round.

The Storyboard Creator then generates N_b storyboard candidates ($N_b = 3$) for the selected script. Unlike scripts, closed-source storyboards are selected as a complete image because their panels are generated jointly and share style, color, and lighting; recombining panels across candidates could introduce visual inconsistencies. The Storyboard Reviewer compares candidates pairwise in the first round and, in later rounds, compares each new candidate with the previous best, judging each storyboard against the script that produced it. If no new candidate wins, the previous best storyboard is retained with its script and feedback, keeping the final pair consistent. The open-source backend instead generates and selects panels individually before compositing them due to its constraints (more details in Appendix B).

Script-level and storyboard-level feedback are jointly provided to the next-round Script Creator, together with the best script from the previous round and the source manga panels. This dual-level feedback allows the pipeline to refine the textual specification using evidence from its visual realization. Issues flagged by the Storyboard Reviewer are treated as symptoms of ambiguous or inaccurate script descriptions: the Script Creator is instructed to identify the responsible part of the

script, such as an ambiguous spatial description or a missing detail. To prevent unnecessary changes to already correct content, the Script Creator revises only the script clips or character descriptions explicitly identified as problematic by the reviewers, while leaving all other entries unchanged. When feedback from different reviewers conflicts, the corresponding source manga panel is used as the primary reference for deciding the revision. The script and storyboard selected in the final round are then passed to the video-generation stage.

3.5 Video Generation Pipeline

To accommodate different input formats of video generation models, our video generation pipeline supports two modes: a first-frame mode and a reference-conditioned mode. The former is used for models that take a single image as the first frame, while the latter is used for models that can take the complete storyboard as a reference and generate a multi-shot video at once. Our main experimental results use the reference-conditioned mode.

In first-frame mode, each storyboard panel serves as the first frame of its corresponding clip. For each clip, the Video Creator generates N_v candidates ($N_v = 2$) conditioned on that panel and its script description, and the Video Reviewer selects the better one using the script, panel, and previously selected clip when available. This greedy and clip-by-clip process enforces cross-clip continuity at selection time, after which the selected clips are concatenated in story order. In reference-conditioned mode, the Video Creator instead uses the complete best script and full storyboard to generate N_v complete multi-shot candidates in one pass, from which the Video Reviewer selects the final result.

4 Experiments

4.1 Datasets

Our evaluation set contains eight stories of six ordered manga panels, drawn from Manga109 (Matsui et al., 2017; Aizawa et al., 2020) and additional cases covering diverse backgrounds, characters, interactions, and actions. All methods use the same stories, enabling evaluation of both panel-level fidelity and cross-clip consistency.

4.2 Baselines

We use StoryDiffusion (Zhou et al., 2024), Story-Iter (Mao et al., 2026), and AnyStory (He et al., 2025b) as storyboard-generation baselines, and AniMaker (Shi et al., 2025), ViMax (Huang et al., 2026), and StoryMem (Zhang et al., 2025a) as video-generation baselines. We further compare with direct manga-to-video generation by feeding the manga pages directly into Seedance 2.0 or MiniMax-H3. To compare generation ability rather than manga understanding, every script-based baseline receives the same refined script as Manga2Animate. We also provide the same manga character reference images whenever the baseline accepts reference images. Full adaptation details are in Appendix E.

4.3 Evaluation Metrics

MangaEval adapts existing metrics to the manga input and multi-shot output, evaluating storyboard Visual Consistency, Content Alignment, Aesthetic Quality, and Cost, as well as video Manga Alignment, Video Quality, Aesthetic Quality, and Cost. Full definitions and objectives are in Appendix F (Tabs. A6 and A7).

Video Evaluation Metrics. At the video level, we evaluate I2V Subject (I2VS), I2V Background (I2VB), Subject Consistency (SC), Background Consistency (BC), Motion Smoothness (MS), and Dynamic Degree (DD) segment by segment against the source panels. We replace VBench++'s (Huang et al., 2025) DINO (Caron et al., 2021) with DINOv3 (Siméoni et al., 2025) for I2VS and its binary DD with a continuous motion-magnitude score; aesthetics are evaluated on sampled frames with two anime-specific models.

Detailed storyboard evaluation metrics are provided in Appendix F (Table A6).

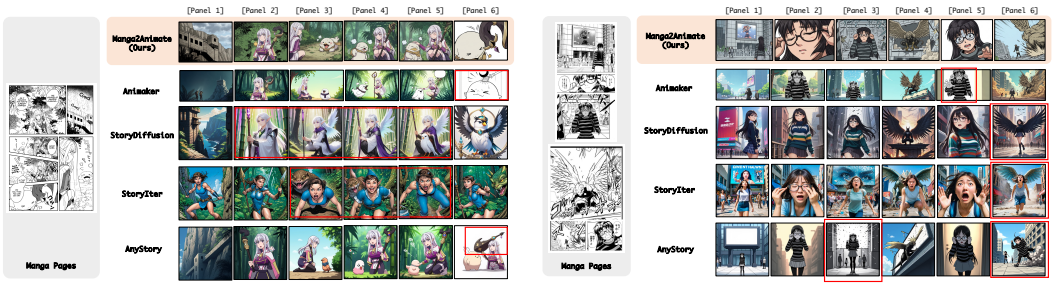
4.4 Quantitative Comparisons

Table 1: **Quantitative comparison of multi-shot video generation.** Our main setting (Gemini + Seedance 2.0) performs better than AniMaker, ViMax, and StoryMem in I2VS, I2VB, BC, Shadow-v2, and Waifu-v3. For a fair comparison with ViMax, both methods use the same script, image model, and video model. Our FLUX.2 + MiniMax-H3 setting also performs well, showing that the improvement is not tied to one model combination. Direct generation has higher DD, but it may animate the page layout or produce the wrong actions (Fig. 2(b)); a higher DD means more movement, not more accurate movement. Red, orange, and yellow mark the top three results; ↑/↓ show whether higher or lower is better. Call counts are not ranked; – means no calls.

Method	Model	Alignment		Video Quality				Aesthetic Quality		Cost		
		I2VS↑	I2VB↑	SC↑	BC↑	MS↑	DD↑	Shadow-v2↑	Waifu-v3↑	Calls (VLM)	Calls (Video Gen)	Time↓
AniMaker	–	0.775	0.749	0.922	0.950	0.979	7.680	0.410	2.694	82.500	–	8h9m
ViMax	Veo 3.1 Fast	0.774	0.752	0.881	0.932	0.978	1.639	0.368	4.394	15.125	6.000	12m18s
ViMax	Seedance 2.0	0.753	0.747	0.927	0.948	0.977	2.541	0.385	4.764	15.125	6.000	16m33s
ViMax	MiniMax-H3	0.771	0.767	0.938	0.957	0.977	0.977	0.326	4.505	15.500	–	28m58s
StoryMem	–	0.728	0.731	0.885	0.922	0.956	5.203	0.275	3.602	10.300	–	2h16m8s
Manga → Video (Direct)	MiniMax-H3	0.777	0.735	0.865	0.932	0.953	11.787	0.402	4.611	–	–	5m54s
Manga → Video (Direct)	Seedance 2.0	0.750	0.711	0.852	0.931	0.951	13.941	0.492	5.250	–	1.000	3m29s
Ours (FLUX.2)	MiniMax-H3	0.794	0.768	0.943	0.953	0.968	6.872	0.549	4.054	59.000	–	2h8m39s
Ours (FLUX.2)	Seedance 2.0	0.781	0.741	0.892	0.942	0.958	9.377	0.595	4.429	59.000	2.000	2h5m46s
Ours (Gemini)	MiniMax-H3	0.792	0.791	0.940	0.953	0.969	3.715	0.413	4.352	29.500	–	27m5s
Ours (Gemini)	Seedance 2.0	0.791	0.768	0.931	0.958	0.960	9.457	0.518	5.090	29.500	2.000	24m49s

Overall, Table 1 shows that our pipeline better preserves the manga content and keeps the video clips visually consistent. The improvement remains when we change the image or video model, suggesting that it comes from the script, storyboard, and review process rather than one specific model. DD only measures how much movement a video contains, so a higher DD does not mean that the movement is correct. Storyboard results are reported in Appendix C.

4.5 Qualitative Analysis



(a) (b)

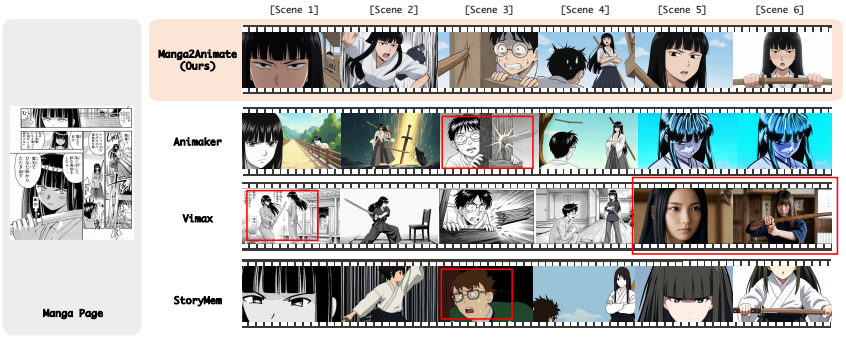
Figure 5: **Qualitative comparison of storyboard generation** on two manga examples (source pages on the left). Manga2Animate keeps the source art style, character identity, and plot across all six panels. AniMaker and AnyStory sometimes leave panels uncolored, StoryDiffusion uses a generic illustration style and repeats content, and Story-Iter becomes photorealistic and changes character appearance. In (b), the baselines also show the falling winged creature incorrectly or leave it out. Red boxes mark example failures.

Storyboard Qualitative Results. Figure 5 shows the benefit of generating and reviewing storyboards with the source panels instead of relying only on the shared refined script. Figure 4 shows how errors are corrected at the script stage.

Video Qualitative Results. As Fig. 6 shows, the reviewed storyboard provides appearance and composition that match the manga, while the script provides actions and dialogue. Together, they reduce incorrect content even though all methods receive the same refined script. Full videos are in the supplementary material.



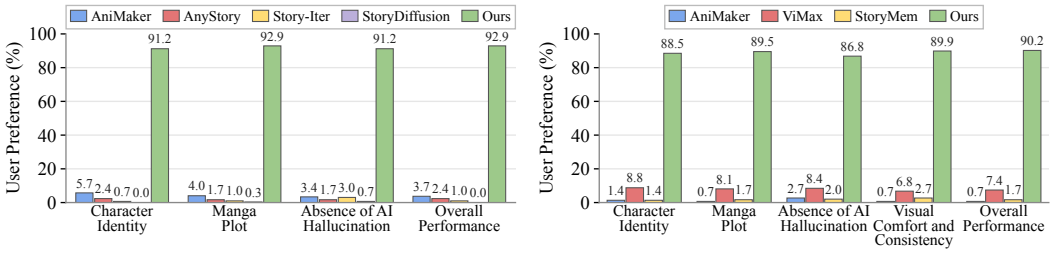
(a)



(b)

Figure 6: **Qualitative comparison of video generation** on two manga examples (source pages on the left). Manga2Animate keeps the manga's content, characters, and art style. AniMaker adds unrelated people or backgrounds, ViMax mixes anime and photorealistic styles and adds text, and StoryMem produces blurry frames and mistakes the basketball scene for tennis. Baselines may also switch to black-and-white frames or change characters within a video; red boxes mark example failures.

4.6 User Study



(a) Storyboard

(b) Video

Figure 7: **User study on storyboard and video generation.** Thirty-seven participants evaluated eight storyboard and eight video cases shown with the input manga; method names were hidden and outputs randomized. For each criterion, participants selected the best result. Manga2Animate achieves the highest preference on every storyboard criterion (Character Identity, Manga Plot, Absence of AI Hallucination, and Overall Performance) and every video criterion (the same four plus Visual Comfort and Consistency).

As shown in Fig. 7, the strong preferences for Manga Plot and Character Identity agree in direction with MangaEval's fidelity metrics (e.g., PA and CIDS), although the human and automatic criteria do not correspond one-to-one.

4.7 Ablation Studies

We ablate the three stage-specific reviewers, compare clip-wise with whole-script selection, remove each level of feedback, ablate the refinement loop and storyboard generation, and evaluate video generation mode, using the main configuration as reference.

Table 2: **Effect of removing each stage-specific reviewer.** When a reviewer is removed, that stage generates only one candidate and receives no review or feedback; all other settings stay the same. Removing any reviewer lowers the overall alignment and consistency scores. Removing the Script Reviewer causes the largest drop in I2VS. Some variants score higher on MS or Shadow-v2, but score lower on alignment, consistency, or DD. Red and orange mark the best and second-best results; higher is better.

Method	Alignment		Video Quality				Aesthetic Quality	
	I2VS↑	I2VB↑	SC↑	BC↑	MS↑	DD↑	Shadow-v2↑	Waifu-v3↑
Without Script Reviewer	0.766	0.752	0.912	0.949	0.963	8.592	0.523	4.789
Without Storyboard Reviewer	0.783	0.756	0.923	0.948	0.971	5.616	0.479	4.540
Without Video Reviewer	0.768	0.749	0.902	0.941	0.957	8.330	0.517	4.996
Full Model	0.791	0.768	0.931	0.958	0.960	9.457	0.518	5.090

Stage-Specific Reviewers. Table 2 shows that all three reviewers contribute to the full model's balanced performance.

Script Selection Strategy. Clip-wise selection improves alignment, consistency, and aesthetics, with only small decreases in I2VB and MS. Full results are in Appendix D (Table A2), and Fig. 4 shows an example.

Dual-Level Feedback. Removing either script- or storyboard-level feedback lowers CSD, CIDS, and PA, showing that the two are complementary. Full results are in Appendix D (Table A3).

Refinement Loop and Storyboard Generation. Skipping the feedback refinement or the storyboard generation step both lower video quality relative to the full model. Full results are in Appendix D (Table A4).

Video Generation Mode. These results show that Manga2Animate can use either mode. We use reference-conditioned generation for our main results; full results are in Appendix D (Table A5).

5 Limitations

Our method has several limitations. First, as an agentic pipeline built on stochastic generative models, its outputs may vary across runs. Second, API-based models may restrict generation of copyrighted or policy-sensitive content. Finally, when the source manga is highly sketchy, cluttered, or compositionally complex, or a scene requires context from elsewhere in the story, the pipeline may misinterpret panel content.

6 Conclusion

We present Manga2Animate, a multi-agent framework for faithfully adapting existing manga into coherent multi-shot animation. We decompose the task into two intermediate representations grounded in the source manga, an animation script and a visual storyboard. Creator and reviewer agents iteratively generate, review, and refine both. Since the two representations fail in different ways, we compose scripts clip by clip, select storyboards as a whole, and feed storyboard-level issues back to the script. Under MangaEval, an evaluation protocol adapted from ViStoryBench and VBench++, Manga2Animate outperforms state-of-the-art storyboard and video generation methods on manga-fidelity and consistency metrics, even when baselines receive the same refined script and character references, and is strongly preferred in a user study. Promising future directions include improving robustness to complex or roughly sketched art styles, developing source-grounded evaluation of story events, and extending narrative understanding across pages and chapters.

AI USE STATEMENT

In this work, we used generative AI tools to polish the writing of the paper, such as improving grammar, wording, and readability. We have not used generative AI tools for research ideation, method design, experiment design, or the analysis and interpretation of results. Note that the proposed Manga2Animate framework itself uses generative models (e.g., VLMs and image and video generation models) as components, as described in Sec. 3 and Appendix A; this use is part of the method rather than of the writing process. We have reviewed all AI-assisted text, and we take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Data and copyright. Our evaluation set combines two sources of manga panels (Sec. 4.1): (i) Manga109 (Matsui et al., 2017; Aizawa et al., 2020), whose access and use are governed by the Manga109 academic-use agreement, and (ii) a small number of additional manga cases used only to test whether Manga2Animate generalizes beyond Manga109's specific art styles and layouts. The panels in this second group, and the videos our method generates from them, remain the property of their original copyright holders; we hold no license for them. We include only a handful of low-resolution panels per case, together with the corresponding generated videos, solely to illustrate and evaluate the method's behavior on manga that we did not curate or license ourselves, which we believe constitutes limited research quotation. We do not use these materials for any commercial purpose, we do not redistribute the source panels as a dataset, and we will promptly remove any specific example from the paper, appendix, or supplementary website upon request from the corresponding rights holder. Beyond our own use, we note a broader dual-use concern: a system that automatically adapts existing manga into animation could, if applied to commercially licensed manga without authorization, be used to produce unauthorized derivative works that substitute for or compete with official adaptations, potentially harming the economic interests of the original creators and publishers. Manga2Animate is released as a research prototype for studying source-grounded, agentic manga-to-animation generation. We do not endorse, and explicitly discourage, applying it to copyrighted manga without the rights holder's permission, and we encourage anyone building on this work to use it only on manga they own or are licensed to adapt.

User study. The 37 participants in our human study (Sec. 4.6) took part voluntarily after being informed of the purpose and procedure of the study, and gave their consent before starting. The task involved only viewing and comparing anonymized outputs, presented with method names hidden and in random order, and selecting a preferred method for each criterion; videos were muted by default. We did not collect any personally identifiable information, and all responses were recorded and analyzed anonymously. Participants could withdraw at any time without penalty. We followed standard practice for minimal-risk behavioral studies of this kind, in which the only task is comparing already-public-facing generated media.

REPRODUCIBILITY STATEMENT

We provide extensive implementation details to support reproduction of Manga2Animate and our evaluation.

Method. All agent prompts (the Character Description, Script, Storyboard, and Video Creators and their corresponding Reviewers) are given verbatim in Appendix H, together with the reviewer criteria, scoring rules, and tie-breaking order used at every stage (Appendix G). Model versions, sampling settings, output resolutions, the number of refinement rounds and candidates (N=2, N_s=3, N_b=3, N_v=2), and the hardware used are listed in Appendix A. Manga2Animate supports two generation backends: a closed-source one that uses the hosted Gemini image model and Seedance 2.0, and an open-source one that runs FLUX.2-dev and MiniMax-H3 or Index-AniSora on local GPUs (Appendix B). In both configurations, MAGI runs locally, while the Character Description and Script Creators and all reviewer agents use hosted Gemini models (Appendix A). We will publicly release our full code, including the agent pipeline, all prompts, the open-source backend integration, and the MangaEval implementation, upon publication.

Baselines and metrics. Appendix E details, for every baseline, exactly how the shared refined script and character references were converted into its native input format, so that the comparisons in Sec. 4.4 and Appendix C can be reproduced under the same protocol. Appendix F gives full formulas, backbones, and judge models for every MangaEval metric, including our modifications to the metrics adapted from ViStoryBench (Zhuang et al., 2026) and VBench++ (Huang et al., 2025).

Code and data availability. Our full code, including the agent pipeline, all prompts, the MangaEval implementation, and the open-source backend's integration code (Appendix B), will be made publicly available upon publication. Until then, Appendices H, G, and F give all prompts, reviewer rules, and metric definitions, and Appendix A and Appendix B give the model checkpoints, hyperparameters, and generation/selection procedure in enough detail to reimplement the open-source generation components and reproduce the remaining stages using the documented hosted Gemini APIs. Our evaluation set draws on Manga109 (Matsui et al., 2017; Aizawa et al., 2020); consistent with its academic-use terms, we do not redistribute Manga109 images ourselves, and we will release the exact list of stories and panels used together with our code, so that the Manga109 subset of our evaluation set can be reconstructed by anyone who separately obtains access under Manga109's own terms. The additional, non-Manga109 manga cases shown in some qualitative examples (Sec. 4.5 and the supplementary website) are included only as limited illustrative excerpts (see Ethics statement) and are not released as a redistributable dataset; readers wishing to reproduce results on manga beyond Manga109 should supply their own manga pages as input, since our pipeline is designed to take arbitrary manga panels and a character reference image as input (Sec. 3.1).

Caveats. Several pipeline stages rely on closed, hosted, third-party models (the Gemini family, Seedance 2.0) whose behavior may change over time and whose sampling process we do not fully control beyond fixing the documented model identifiers and default parameters; exact numerical reproduction of Tables 1 and A1 is therefore not guaranteed even with a fixed random seed, although we expect the qualitative conclusions and relative comparisons between methods to hold. Wall-clock timings additionally depend on API latency and local hardware, as noted in Appendix A. The open-source backend removes the dependence on hosted image and video generation models, although its creator and reviewer agents still use Gemini.

REFERENCES

Kiyoharu Aizawa, Azuma Fujimoto, Atsushi Otsubo, Toru Ogawa, Yusuke Matsui, Koki Tsubota, and Hikaru Ikuta. Building a manga dataset" manga109" with annotations for multimedia applications. *arXiv preprint arXiv:2005.04425*, 2020.

Jeonghun Baek, Kazuki Egashira, Shota Onohara, Atsuyuki Miyai, Yuki Imajuku, Hikaru Ikuta, and Kiyoharu Aizawa. Mangavqa and mangalmm: A benchmark and specialized model for multimodal manga understanding. In *Findings of the Association for Computational Linguistics: EACL 2026*, pp. 5349–5370, 2026.

Aaditya Baranwal, Madhav Kataria, Naitik Agrawal, Yogesh S Rawat, and Shruti Vyas. Re: Verse-can your vlm read a manga? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3820–3830, 2025.

Ying Cao, Xufang Pang, Antoni B Chan, and Rynson WH Lau. Dynamic manga: Animating still manga via camera movement. *IEEE Transactions on Multimedia*, 19(1):160–172, 2016.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *2021 IEEE/CVF international conference on computer vision (ICCV)*, pp. 9630–9640. IEEE, 2021.

Junhao Cheng, Baiqiao Yin, Kaixin Cai, Minbin Huang, Hanhui Li, Yuxin He, Xi Lu, Yue Li, Yifei Li, Yuhao Cheng, et al. Theatergen: Character management with llm for consistent multi-turn image generation. *arXiv preprint arXiv:2404.18919*, 2024.

Stephanie Fu, Netanel Tamir, Shobhita Sundaram, Lucy Chai, Richard Zhang, Tali Dekel, and Phillip Isola. Dreamsim: Learning new dimensions of human visual similarity using synthetic data. *arXiv preprint arXiv:2306.09344*, 2023.

Silin Gao, Sheryl Mathew, Li Mi, Sepideh Mamooler, Mengjie Zhao, Hiromi Wakaki, Yuki Mitsufuji, Syrielle Montariol, and Antoine Bosselut. Vinabench: Benchmark for faithful and consistent visual narratives. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2870–2879. IEEE, 2025.

Yuwei Guo, Ceyuan Yang, Ziyan Yang, Zhibei Ma, Zhijie Lin, Zhenheng Yang, Dahua Lin, and Lu Jiang. Long context tuning for video generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17281–17291, 2025.

Vaibhavi Gupta, Vinay Detani, Vivek Khokar, and Chiranjoy Chattopadhyay. C2vnet: A deep learning framework towards comic strip to audio-visual scene synthesis. In *International Conference on Document Analysis and Recognition*, pp. 160–175. Springer, 2021.

Huiguo He, Huan Yang, Zixi Tuo, Yuan Zhou, Qiuyue Wang, Yuhang Zhang, Zeyu Liu, Wenhao Huang, Hongyang Chao, and Jian Yin. Dreamstory: Open-domain story visualization by llm-guided multi-subject consistent diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025a.

Junjie He, Yuxiang Tuo, Binghui Chen, Chongyang Zhong, Yifeng Geng, and Liefeng Bo. Anystory: Towards unified single and multiple subject personalization in text-to-image generation. *arXiv preprint arXiv:2501.09503*, 2025b.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in neural information processing systems*, 35:8633–8646, 2022.

Panwen Hu, Jin Jiang, Jianqi Chen, Mingfei Han, Shengcai Liao, Xiaojun Chang, and Xiaodan Liang. Storyagent: Customized storytelling video generation via multi-agent collaboration. *arXiv preprint arXiv:2411.04925*, 2024.

Teng Hu, Zhentao Yu, Zhengguang Zhou, Sen Liang, Yuan Zhou, Qin Lin, and Qinglin Lu. Hunyuancustom: A multimodal-driven architecture for customized video generation. *arXiv preprint arXiv:2505.04512*, 2025.

Lingxuan Huang, Sizhe He, Hengji Zhou, Liqiang Nie, Lianghao Xia, and Chao Huang. Vimax: Agentic video generation. *arXiv preprint arXiv:2606.07649*, 2026.

Zhitong Huang, Mohan Zhang, and Jing Liao. Lvcd: reference-based lineart video colorization with diffusion models. *ACM Transactions on Graphics (TOG)*, 43(6):1–11, 2024.

Ziqi Huang, Fan Zhang, Xiaojie Xu, Yinan He, Jiashuo Yu, Ziyue Dong, Qianli Ma, Nattapol Chanpaisit, Chenyang Si, Yuming Jiang, et al. Vbench++: Comprehensive and versatile benchmark suite for video generative models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Hikaru Ikuta, Leslie Wöhler, and Kiyoharu Aizawa. Mangaub: A manga understanding benchmark for large multimodal models. *IEEE MultiMedia*, 32(2):33–43, 2025.

Yudong Jiang, Baohan Xu, Siqian Yang, Mingyu Yin, Jing Liu, Chao Xu, Siqi Wang, Yidi Wu, Bingwen Zhu, Xinwen Zhang, et al. Anisora: Exploring the frontiers of animation video generation in the sora era. *arXiv preprint arXiv:2412.10255*, 2024.

Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. Vace: All-in-one video creation and editing. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17191–17202. IEEE, 2025.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

Lingen Li, Guangzhi Wang, Zhaoyang Zhang, Yaowei Li, Xiaoyu Li, Qi Dou, Jinwei Gu, Tianfan Xue, and Ying Shan. Tooncomposer: Streamlining cartoon production with generative post-keyframing. *arXiv preprint arXiv:2508.10881*, 2025.

Yingxuan Li, Kiyoharu Aizawa, and Yusuke Matsui. Manga109dialog: A large-scale dialogue dataset for comics speaker detection. In *2024 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6. IEEE, 2024a.

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 *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6322–6331. IEEE, 2019.

Yunxin Li, Haoyuan Shi, Baotian Hu, Longyue Wang, Jiashun Zhu, Jinyi Xu, Zhen Zhao, and Min Zhang. Anim-director: A large multimodal model powered agent for controllable animation video generation. In *SIGGRAPH Asia 2024 Conference Papers*, pp. 1–11, 2024b.

Zhen Li, Zuo-Liang Zhu, Ling-Hao Han, Qibin Hou, Chun-Le Guo, and Ming-Ming Cheng. Amt: All-pairs multi-field transforms for efficient frame interpolation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9801–9810. IEEE, 2023.

Han Lin, Abhay Zala, Jaemin Cho, and Mohit Bansal. Videodirectorgpt: Consistent multi-scene video generation via llm-guided planning. *arXiv preprint arXiv:2309.15091*, 2023.

Chang Liu, Haoning Wu, Yujie Zhong, Xiaoyun Zhang, Yanfeng Wang, and Weidi Xie. Intelligent grimm-open-ended visual storytelling via latent diffusion models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6190–6200. IEEE, 2024a.

Lijie Liu, Tianxiang Ma, Bingchuan Li, Zhuowei Chen, Jiawei Liu, Gen Li, Siyu Zhou, Qian He, and Xinglong Wu. Phantom: Subject-consistent video generation via cross-modal alignment. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 14951–14961. IEEE, 2025a.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision*, pp. 38–55. Springer, 2024b.

Zhiheng Liu, Ka Leong Cheng, Xi Chen, Jie Xiao, Hao Ouyang, Kai Zhu, Yu Liu, Yujun Shen, Qifeng Chen, and Ping Luo. Manganinja: Line art colorization with precise reference following. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5666–5677. IEEE, 2025b.

Do Xuan Long, Xingchen Wan, Hootan Nakhost, Chen-Yu Lee, Tomas Pfister, and Sercan Ö Arik. Vista: A test-time self-improving video generation agent. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6021–6032, 2026.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. Self-refine: Iterative refinement with self-feedback. *Advances in neural information processing systems*, 36:46534–46594, 2023.

Jiawei Mao, Xiaoke Huang, Yunfei Xie, Yuanqi Chang, Mude Hui, Bingjie Xu, Zeyu Zheng, Zirui Wang, Cihang Xie, and Yuyin Zhou. Story-iter: A training-free iterative paradigm for long story visualization. In *International Conference on Learning Representations*, volume 2026, pp. 113143–113173, 2026.

Yusuke Matsui, Kota Ito, Yuji Aramaki, Azuma Fujimoto, Toru Ogawa, Toshihiko Yamasaki, and Kiyoharu Aizawa. Sketch-based manga retrieval using manga109 dataset. *Multimedia tools and applications*, 76(20):21811–21838, 2017.

Yihao Meng, Hao Ouyang, Hanlin Wang, Qiuyu Wang, Wen Wang, Ka Leong Cheng, Zhiheng Liu, Yujun Shen, and Huamin Qu. Anidoc: Animation creation made easier. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18187–18197. IEEE, 2025.

Yihao Meng, Hao Ouyang, Yue Yu, Qiuyu Wang, Wen Wang, Ka Leong Cheng, Hanlin Wang, Shuailei Ma, Yixuan Li, Cheng Chen, et al. Holocine: Holistic generation of cinematic multi-shot long video narratives. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 461–471, 2026.

Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024.

Lu Qiu, Yizhuo Li, Yuying Ge, Yixiao Ge, Ying Shan, and Xihui Liu. Animeshooter: A multi-shot animation dataset for reference-guided video generation. *arXiv preprint arXiv:2506.03126*, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Tanzila Rahman, Hsin-Ying Lee, Jian Ren, Sergey Tulyakov, Shweta Mahajan, and Leonid Sigal. Make-a-story: Visual memory conditioned consistent story generation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2493–2502. IEEE, 2023.

Ragav Sachdeva and Andrew Zisserman. The manga whisperer: Automatically generating transcriptions for comics. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12967–12976. IEEE, 2024.

Ragav Sachdeva and Andrew Zisserman. From panels to prose: Generating literary narratives from comics. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 21864–21873. IEEE, 2025.

Ragav Sachdeva, Gyungin Shin, and Andrew Zisserman. Tails tell tales: Chapter-wide manga transcriptions with character names. In *Asian Conference on Computer Vision*, pp. 63–80. Springer, 2024.

Team Seedance, De Chen, Liyang Chen, Xin Chen, Ying Chen, Zhuo Chen, Zhuowei Chen, Feng Cheng, Tianheng Cheng, Yufeng Cheng, et al. Seedance 2.0: Advancing video generation for world complexity. *arXiv preprint arXiv:2604.14148*, 2026.

Conghao Tom Shen, Violet Yao, and Yixin Liu. Maru: A manga retrieval and understanding system connecting vision and language. *arXiv preprint arXiv:2311.02083*, 2023.

Haoyuan Shi, Yunxin Li, Xinyu Chen, Longyue Wang, Baotian Hu, and Min Zhang. Animaker: Multi-agent animated storytelling with mcts-driven clip generation. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, pp. 1–11, 2025.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in neural information processing systems*, 36:8634–8652, 2023.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Gowthami Somepalli, Anubhav Gupta, Kamal Gupta, Shramay Palta, Micah Goldblum, Jonas Geiping, Abhinav Shrivastava, and Tom Goldstein. Measuring style similarity in diffusion models. *arXiv preprint arXiv:2404.01292*, 2024.

Yunlong Tang, Junjia Guo, Pinxin Liu, Zhiyuan Wang, Hang Hua, Jia-Xing Zhong, Yunzhong Xiao, Chao Huang, Luchuan Song, Susan Liang, et al. Generative ai for cel-animation: A survey. In *2025 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pp. 3837–3850. IEEE, 2025.

Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pp. 402–419. Springer, 2020.

Yoad Tewel, Omri Kaduri, Rinon Gal, Yoni Kasten, Lior Wolf, Gal Chechik, and Yuval Atzmon. Training-free consistent text-to-image generation. *ACM Transactions on Graphics (TOG)*, 43(4): 1–18, 2024.

Emanuele Vivoli, Marco Bertini, and Dimosthenis Karatzas. Comix: A comprehensive benchmark for multi-task comic understanding. *Advances in Neural Information Processing Systems*, 37: 140828–140846, 2024a.

Emanuele Vivoli, Mohamed Ali Souibgui, Andrey Barsky, Artemis LLabres, Marco Bertini, and Dimosthenis Karatzas. One missing piece in vision and language: A survey on comics understanding. *arXiv preprint arXiv:2409.09502*, 2024b.

Emanuele Vivoli, Artemis Llabrés, Mohamed Ali Souibgui, Marco Bertini, Ernest Valveny Llobet, and Dimosthenis Karatzas. Comicspap: understanding comic strips by picking the correct panel. In *International Conference on Document Analysis and Recognition*, pp. 337–350. Springer, 2025.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Muyao Wang, Zeke Xie, Yanhao Chen, Lixin Xiu, and Hideki Nakayama. Mangaflow: An end-to-end agentic framework for controllable story to manga generation. *arXiv preprint arXiv:2605.28173*, 2026a.

Qian Wang, Ziqi Huang, Ruoxi Jia, Paul Debevec, and Ning Yu. Mavis: A multi-agent framework for long-sequence video storytelling. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2273–2295, 2026b.

Wen Wang, Canyu Zhao, Hao Chen, Zhekai Chen, Kecheng Zheng, and Chunhua Shen. Autostory: Generating diverse storytelling images with minimal human efforts. *International Journal of Computer Vision*, 133(6):3083–3104, 2025a.

Xierui Wang, Siming Fu, Qihan Huang, Wanggui He, and Hao Jiang. Ms-diffusion: Multi-subject zero-shot image personalization with layout guidance. In *International Conference on Learning Representations*, volume 2025, pp. 95118–95146, 2025b.

Jianzong Wu, Chao Tang, Jingbo Wang, Yanhong Zeng, Xiangtai Li, and Yunhai Tong. Diffsensei: Bridging multi-modal llms and diffusion models for customized manga generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28684–28693. IEEE, 2025a.

Shaojin Wu, Mengqi Huang, Wenxu Wu, Yufeng Cheng, Fei Ding, and Qian He. Less-to-more generalization: Unlocking more controllability by in-context generation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 18682–18692. IEEE, 2025b.

Weijia Wu, Zeyu Zhu, and Mike Zheng Shou. Automated movie generation via multi-agent cot planning. *arXiv preprint arXiv:2503.07314*, 2025c.

Junfei Xiao, Ceyuan Yang, Lvmin Zhang, Shengqu Cai, Yang Zhao, Yuwei Guo, Gordon Wetzstein, Maneesh Agrawala, Alan Yuille, and Lu Jiang. Captain cinema: Towards short movie generation. In *International Conference on Learning Representations*, volume 2026, pp. 83349–83375, 2026.

Jinbo Xing, Hanyuan Liu, Menghan Xia, Yong Zhang, Xintao Wang, Ying Shan, and Tien-Tsin Wong. Tooncrafter: Generative cartoon interpolation. *ACM Transactions on Graphics (TOG)*, 43 (6):1–11, 2024.

Xuenan Xu, Jiahao Mei, Chenliang Li, Yuning Wu, Ming Yan, Shaopeng Lai, Ji Zhang, and Mengyue Wu. Mm-storyagent: Immersive narrated storybook video generation with a multi-agent paradigm across text, image and audio. *arXiv preprint arXiv:2503.05242*, 2025a.

Zhenran Xu, Longyue Wang, Jifang Wang, Zhouyi Li, Senbao Shi, Xue Yang, Yiyu Wang, Baotian Hu, Jun Yu, and Min Zhang. Filmagent: A multi-agent framework for end-to-end film automation in virtual 3d spaces. *arXiv preprint arXiv:2501.12909*, 2025b.

Hailong Yan, Shice Liu, Tao Wang, Xiangtao Zhang, Yijie Zhong, Jinwei Chen, Le Zhang, and Bo Li. Animeagent: Is the multi-agent via image-to-video models a good disney storytelling artist? *arXiv preprint arXiv:2602.20664*, 2026.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations*, volume 2025, pp. 83048–83077, 2025.

Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11941–11952. IEEE, 2023.

Kaiwen Zhang, Liming Jiang, Angtian Wang, Jacob Zhiyuan Fang, Tiancheng Zhi, Qing Yan, Hao Kang, Xin Lu, and Xingang Pan. Storymem: Multi-shot long video storytelling with memory. *arXiv preprint arXiv:2512.19539*, 2025a.

Lisai Zhang, Baohan Xu, Siqian Yang, Mingyu Yin, Jing Liu, Chao Xu, Siqi Wang, Yidi Wu, Yuxin Hong, Zihao Zhang, et al. Anime: Adaptive multi-agent planning for long animation generation. In *Proceedings of the SIGGRAPH Asia 2025 Posters*, pp. 1–3. 2025b.

Peixuan Zhang, Zijian Jia, Kaiqi Liu, Shuchen Weng, Si Li, and Boxin Shi. Stage: Storyboard-anchored generation for cinematic multi-shot narrative. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 659–669, 2026.

Zibo Zhao, Zeqiang Lai, Qingxiang Lin, Yunfei Zhao, Haolin Liu, Shuhui Yang, Yifei Feng, Mingxin Yang, Sheng Zhang, Xianghui Yang, et al. Hunyuan3d 2.0: Scaling diffusion models for high resolution textured 3d assets generation. *arXiv preprint arXiv:2501.12202*, 2025.

Jinsong Zhou, Yihua Du, Xinli Xu, Luozhou Wang, Zijie Zhuang, Yehang Zhang, Shuaibo Li, Xiaojun Hu, Bolan Su, and Ying-cong Chen. Videomemory: Toward consistent video generation via memory integration. *arXiv preprint arXiv:2601.03655*, 2026.

Yupeng Zhou, Daquan Zhou, Ming-Ming Cheng, Jiashi Feng, and Qibin Hou. Storydiffusion: Consistent self-attention for long-range image and video generation. *Advances in Neural Information Processing Systems*, 37:110315–110340, 2024.

Cailin Zhuang, Ailin Huang, Yaoqi Hu, Jingwei Wu, Wei Cheng, Jiaqi Liao, Hongyuan Wang, Xinyao Liao, Weiwei Cai, Hengyuan Xu, et al. Vistorybench: Comprehensive benchmark suite for story visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9455–9467, 2026.

APPENDIX

This appendix provides additional details that complement the main paper. Sec. A describes the implementation details, including model versions, hyperparameters, video specifications, hardware, and cost. Sec. B explains the panel-wise generation strategy of the open-source storyboard backend, Sec. C reports the storyboard quantitative evaluation, Sec. D gives the full script-selection-strategy ablation, the dual-level feedback ablation, an ablation on the refinement loop and storyboard generation, and a comparison of reference-conditioned and first-frame video generation, Sec. E describes how each baseline is adapted to manga inputs, Sec. F gives the full definitions of the MangaEval metrics and our modifications, Sec. G summarizes the criteria, scoring, and selection rules of every reviewer, and Sec. H lists the full prompts of all agents. The correspondence among panels, script clips, storyboard panels, and video clips (Sec. 3.1) is illustrated in Fig. A1. We further provide an HTML page in the supplementary material showcasing our generated videos.

A IMPLEMENTATION DETAILS

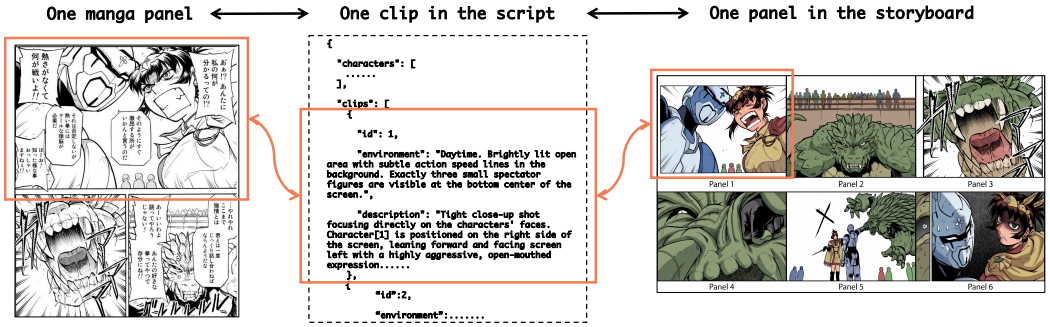


Figure A1: **Correspondence among intermediate representations.** Each manga panel *(left)* corresponds to one clip in the animation script *(middle)*, which describes the environment, character positions, actions, and expressions of that panel in a structured JSON format. Each script clip is then rendered as one storyboard panel *(right)*. In first-frame mode, that panel conditions the corresponding video clip directly; in reference-conditioned mode, the complete storyboard jointly conditions the multi-shot video.

Models. The Character Description Creator, Script Creator, Character Reviewer, Clip Reviewer, Storyboard Reviewer, and open-source Panel Reviewer use gemini-3.7-flash with the thinking level set to HIGH; the Video Reviewer uses gemini-3.1-pro-preview. The closed-source Storyboard Creator uses gemini-3.1-flash-image and outputs a 16:9 storyboard at 1376×768 in a 3×2 grid. We keep the default sampling parameters, including temperature, for all Gemini calls. Panel detection, reading-order estimation, and OCR use MAGI (Sachdeva & Zisserman, 2024). The open-source Storyboard Creator uses FLUX.2-dev with 28 inference steps and a guidance scale of 4.0. It generates three candidates per panel at 768×640, and its panel reviewer, also gemini-3.7-flash, scores each candidate independently and keeps the highest-scoring one per panel. All experiments use $N = 2$ refinement rounds, $N_s = 3$, $N_b = 3$, and $N_v = 2$.

Video generation. In the reference-conditioned mode, the video model receives the complete best script as a text prompt and the full storyboard grid as a single reference image. The closed-source pipeline uses Seedance 2.0 (dreamina-seedance-2-0-260128) in this mode and produces 15 s videos at 1280×720 and 24 fps, with generated audio. The open-source video-generation configuration supports two video backends. MiniMax-H3 (Ref2VA) runs in the same reference-conditioned mode and produces 15 s videos at 1344×768 and 24 fps with audio. Index-AniSora V3.2 (Jiang et al., 2024) runs in the first-frame mode: each cropped storyboard panel is placed as the first frame of its clip, and each clip has 81 frames at 16 fps and 1280×720. Cropped panels are passed to the video model without resizing or padding, and any aspect-ratio adaptation is handled by the video model's own preprocessing. In all cases, two candidate videos are generated and the Video Reviewer

selects the better one; when the reviewer judges the two candidates comparable, the first candidate is kept. All main quantitative comparisons in Table 1 use the reference-conditioned mode; Table A5 separately compares the reference-conditioned and first-frame modes.

Hardware and cost. Gemini and Seedance calls go through hosted APIs. MAGI runs locally on an NVIDIA RTX A6000 (48 GB). FLUX.2-dev runs on an NVIDIA A100 80GB PCIe, MiniMax-H3 on two NVIDIA RTX PRO 6000 Blackwell Server Edition (96 GB) GPUs, and Index-AniSora V3.2 on a single NVIDIA RTX PRO 6000 Blackwell Server Edition (96 GB). For each story, the closed-source pipeline makes 34.5 calls on average in the script and storyboard stages (28.5 Gemini text calls and 6 image-generation calls) and 3 in the video stage (2 Seedance generations and 1 Gemini review), costing about US$5.65 per story in total, of which US$4.55 is Seedance. The costs in Table A1 cover the script and storyboard stages, whereas the VLM calls and time in Table 1 are counted end-to-end, covering script generation, storyboard generation, video generation, and all reviews, and Calls (Video Gen) counts video-generation calls only. For locally run image and video backends, no external generation API calls are required, although the creator and reviewer agents still use Gemini APIs. Their local computational cost is reflected in the reported wall-clock time. Wall-clock times depend on hardware, parallelism, and API latency and should be regarded as indicative.

B Open-Source Storyboard Backend

In our experiments, the open-source image model we tested (FLUX.2-dev) did not reliably generate a complete, high-quality multi-panel storyboard in a single pass. We therefore adopt a panel-wise generation strategy, producing multiple candidates for each panel and selecting the best one before composing the final storyboard. For each panel in the selected script, we construct a prompt that incorporates the character descriptions, the corresponding script content, and constraints derived from the original manga panel to preserve its composition, camera viewpoint, and character pose. When character reference crops are available, they are explicitly associated with the corresponding characters during generation.

FLUX.2-dev is conditioned on the source manga panel and the available character references to generate multiple candidates for each panel. A Gemini-based reviewer then scores each candidate independently on manga/script follow (the primary criterion), character identity consistency, art style and colorization quality, and background fidelity, each on a 0–4 scale, and the highest-scoring candidate is retained. The selected panels are then assembled into a 3×2 storyboard grid, following the same output format as the closed-source backend to enable evaluation under a unified protocol. Since every panel is conditioned on its source manga panel and the shared character references, these common conditions help maintain cross-panel consistency.

C Storyboard Quantitative Evaluation

Evaluation metrics. We evaluate storyboards with CSD, CIDS, PA, and OCCM from ViStory-Bench (Zhuang et al., 2026), adapting three of them to manga: CIDS uses SigLIP (Zhai et al., 2023) instead of CLIP (Radford et al., 2021) for finer character-crop detail, OCCM's expected character count comes from the source manga panel rather than the script's cast list, and PA is judged by GPT-4.1 instead of Gemini (our pipeline's reviewer) to reduce self-preference bias. Since all script-conditioned methods share the same refined script, PA measures content adherence under an identical specification, while CSD, CIDS, and OCCM (and the Manga Plot criterion of our user study) capture fidelity to the source manga itself; full definitions are in Appendix F. For Aesthetic Quality, we replace ViStoryBench's general-purpose predictor with two anime-specific models, Aesthetic Shadow v2[1] and Waifu-v3[2], trained on anime/illustration imagery and used for training-data curation in production anime models.

[1]Released by shadowlilac as shadowlilac/aesthetic-shadow-v2, which is no longer available; we use the community mirror at https://huggingface.co/NeoChen1024/aesthetic-shadow-v2-backup.

[2]https://huggingface.co/Eugeoter/waifu-scorer-v3

Table A1: **Quantitative comparison of storyboard generation.** Red, orange, and yellow cells denote the best, second-best, and third-best results; ↑/↓ indicate that higher/lower is better. Call counts are not ranked; – denotes no calls of that type.

Method	V.C.		C.A.		A.Q.		Cost		
	CSD↑	CIDS↑	PA↑	OCCM↑	Shadow-v2↑	Waifu-v3↑	Calls (VLM)	Calls (Img Gen)	Time↓
StoryDiffusion	0.509	0.713	2.05	57.5	0.697	5.789	–	–	49s
Story-Iter	0.486	0.652	1.85	66.3	0.615	6.308	60.000	11.000	13m20s
AnyStory	0.605	0.792	2.66	69.3	0.606	3.781	10.000	1.000	2m7s
AniMaker	0.508	0.797	3.34	67.5	0.448	4.522	82.500	18.000	8h9m
Ours (FLUX.2)	0.622	0.797	3.14	73.5	0.552	3.423	58.000	–	1h59m
Ours (Gemini 3.1 Image)	0.605	0.808	3.50	72.2	0.350	3.798	28.500	6.000	20m2s

Results. Table A1 compares storyboard generation across methods. The two Manga2Animate configurations jointly achieve the best results on all four manga-fidelity metrics: the open-source FLUX.2 backend reaches the highest CSD and OCCM, while the closed-source Gemini backend achieves the highest CIDS and PA, each improving over the best baseline. Since all baselines use the same refined script as our method, the PA gap reflects content adherence under an identical specification, and this advantage holds for both an open and a closed image backend rather than relying on a specific image model.

In contrast, StoryDiffusion and Story-Iter score higher on the anime-specific aesthetic metrics but considerably lower on manga fidelity (CIDS, PA, OCCM), since both freely generate imagery from text without being constrained by the original characters and composition. This indicates that single-image aesthetics and adaptation fidelity are distinct aspects: an aesthetic predictor assesses the appeal of an individual illustration without considering whether it is consistent with the source. The strengths of Manga2Animate lie in character preservation, content alignment, and cross-panel consistency, which matter most for this task, and participants in our user study also clearly preferred our results overall (Sec. 4.6).

The two backends offer different cost trade-offs: FLUX.2 avoids external image-generation APIs but takes 1 h 59 min on local GPUs, while Gemini takes 20 min 2 s with 6 image-generation calls. Both are slower than StoryDiffusion and AnyStory but considerably cheaper than AniMaker, another multi-agent pipeline (8 h 9 min); users can choose a backend according to their compute resources and API budget.

D Additional Ablation Results

Script selection strategy. Sec. 4.7 compares clip-wise and script-wise script selection and reports the headline result; Table A2 gives the full metric breakdown. Clip-wise selection outperforms script-wise selection on I2VS, Subject Consistency, Background Consistency, and both aesthetic metrics, raising Waifu-v3 from 4.423 to 5.090 and Shadow-v2 from 0.473 to 0.518, while DD increases from 5.951 to 9.457. I2VB and Motion Smoothness are slightly lower than with script-wise selection (0.768 vs. 0.772 and 0.960 vs. 0.963), with both gaps within 0.004. We conjecture that clip-wise selection retains the more complete and specific clip descriptions among the candidates, enabling the video model to produce richer motion and visuals, whereas combining descriptions from different candidates may introduce slight variations in scene settings, causing the small drop in I2VB.

Dual-level feedback. To examine the contribution of each feedback source in the refinement loop (Sec. 3.4), we withhold either the script-level or the storyboard-level feedback from the next-round Script Creator, while the reviewers still select the best script and storyboard in every round. All variants use the closed-source configuration, i.e., gemini-3.7-flash for all creator and reviewer agents and gemini-3.1-flash-image as the Storyboard Creator, and are evaluated at the storyboard level, where the feedback takes effect directly. Withholding a feedback source changes only the content of the Script Creator's input, not the number of generation or review calls, so every variant has the same computational cost as the full pipeline (Table A1); we therefore omit the cost columns. As shown in Table A3, removing either feedback source lowers CSD (0.605 to 0.597 and 0.588), CIDS

Table A2: **Ablation on script selection strategy.** We compare Manga2Animate with clip-wise and script-wise selection. The full model selects the best clip description for each manga panel in a clip-wise manner and composes them into the final script, whereas script-wise selection directly picks one complete candidate script. All other models and settings are unchanged. Red cells denote the better result in each column.

Method	Alignment		Video Quality				Aesthetic Quality	
	I2VS↑	I2VB↑	SC↑	BC↑	MS↑	DD↑	Shadow-v2↑	Waifu-v3↑
Script-wise Selection	0.785	0.772	0.928	0.956	0.963	5.951	0.473	4.423
Clip-wise Selection (Full Model)	0.791	0.768	0.931	0.958	0.960	9.457	0.518	5.090

Table A3: **Ablation on dual-level feedback.** The reviewers still select the best script and storyboard, but the feedback of one reviewer is withheld from the next-round Script Creator. All variants use the closed-source (Gemini) configuration. The Full Model row is the run reported in Table A1. Cost is identical to the full pipeline and is therefore omitted. Red and orange mark the best and second-best results; higher is better.

Method	V.C.		C.A.		A.Q.	
	CSD↑	CIDS↑	PA↑	OCCM↑	Shadow-v2↑	Waifu-v3↑
Without Script Feedback	0.597	0.786	3.33	70.8	0.334	3.813
Without Storyboard Feedback	0.588	0.770	3.39	72.6	0.249	3.257
Full Model	0.605	0.808	3.50	72.2	0.350	3.798

(0.808 to 0.786 and 0.770), and PA (3.50 to 3.33 and 3.39) for the variants without script and without storyboard feedback, respectively. For OCCM, removing script feedback lowers the score from 72.2 to 70.8, whereas removing storyboard feedback yields a comparable 72.6. Removing storyboard feedback also lowers both aesthetic metrics more than removing script feedback (Shadow-v2: 0.249 vs. 0.334; Waifu-v3: 3.257 vs. 3.813), though the without-script-feedback variant slightly exceeds the full model on Waifu-v3 (3.813 vs. 3.798). These results indicate that script-level and storyboard-level feedback are complementary, and combining them mainly improves character identity and content alignment.

Refinement loop and storyboard generation. Skipping the second-round feedback (w/o Refinement) or replacing the generated storyboard with the original manga panels (w/o Storyboard Gen.) both lower I2VS, I2VB, SC, and BC relative to the full model, and removing storyboard generation causes the larger drop, confirming that both stages contribute to the final video quality.

Table A4: **Ablation on the refinement loop and storyboard generation.** *w/o Refinement* uses the first-pass script and storyboard directly, skipping the second-round feedback; *w/o Storyboard Gen.* keeps the same fully refined script but replaces the generated storyboard with a composite of the original manga panels. Removing either stage lowers I2VS, I2VB, SC, and BC relative to the full model, and removing storyboard generation causes the larger drop. Some variants score higher on MS or Shadow-v2, but lower on alignment, consistency, or DD. Red and orange mark the best and second-best results; higher is better.

Method	Alignment		Video Quality				Aesthetic Quality	
	I2VS↑	I2VB↑	SC↑	BC↑	MS↑	DD↑	Shadow-v2↑	Waifu-v3↑
w/o Refinement	0.778	0.754	0.914	0.952	0.965	7.872	0.521	4.818
w/o Storyboard Gen.	0.766	0.745	0.896	0.947	0.961	8.289	0.515	5.050
Full Model	0.791	0.768	0.931	0.958	0.960	9.457	0.518	5.090

Video generation mode. Manga2Animate supports both reference-conditioned and first-frame video generation. In the former, the video model receives the complete storyboard as a reference

image and generates a full video. In the latter, each storyboard panel serves as the first frame of its corresponding clip, and the selected clips are assembled in order.

Table A5: **Comparison of Video Generation Modes.** Both modes use Gemini for storyboard generation and Seedance 2.0 for video generation. Bold indicates the higher value for each metric. Reference-conditioned generation gives better subject alignment, background consistency, and aesthetic scores, while first-frame generation gives better background alignment and slightly better subject consistency and motion smoothness.

Mode	Alignment		Video Quality				Aesthetic Quality	
	I2VS↑	I2VB↑	SC↑	BC↑	MS↑	DD↑	Shadow-v2↑	Waifu-v3↑
Reference-conditioned	**0.791**	0.768	0.931	**0.958**	0.960	**9.457**	**0.518**	**5.090**
First-frame	0.772	**0.778**	**0.935**	0.941	**0.962**	4.045	0.302	4.042

As shown in Table A5, reference-conditioned mode obtains higher I2VS, BC, Shadow-v2, and Waifu-v3 scores, while first-frame mode obtains higher I2VB and slightly higher SC and MS scores. The results demonstrate that the pipeline can also operate with first-frame video generation when full-storyboard reference conditioning is unavailable. We use reference-conditioned mode for the main results.

E Baseline Adaptation

StoryDiffusion (storyboard). StoryDiffusion's official multi-character mechanism takes only plain-text, per-shot prompts that tag which characters appear, with no interface for image-conditioned identity. To keep the comparison fair, we take the same final best script produced by Manga2Animate's own creator-reviewer refinement (Sec. 3.4) and reformat character descriptions and per-shot scene/action text into the per-shot prompts its official multi-character mechanism requires. Because StoryDiffusion has no reference-image input, the manga-cropped character images used elsewhere in our evaluation do not apply here; identity relies purely on text description, a limitation of the method when processing image information. Each story yields 6 panels, composed into the same 3×2 grid as the other backends for evaluation.

Story-Iter (storyboard). Story-Iter generates a sequence of story images from ordered text prompts and maintains cross-image consistency through Global Reference Cross-Attention (GRCA) over previously generated results. To unify the story content, we convert the final refined script of Manga2Animate (Sec. 3.4) into six prompts ordered by manga panel, preserving the characters, interactions, scenes, actions, and camera descriptions. Generation follows the official settings with the comic style. Since Story-Iter does not accept external character reference images, we do not add an identity-control module. The six generated images are composed into a 3×2 grid in order and scored with MangaEval.

AnyStory (storyboard). AnyStory supports personalized story image generation with character reference images. To unify the story content, we convert the final refined script of Manga2Animate (Sec. 3.4) into the structured script required by AnyStory, preserving the characters, scenes, actions, interactions, and camera descriptions of the six panels, and we use the same manga-cropped character images as character references. All other stages follow the official AnyStory implementation, without adding the Storyboard Creator or Reviewer of Manga2Animate. The six generated images are composed into a 3×2 grid in panel order and scored with MangaEval.

AniMaker (video). AniMaker natively takes a plain-text story and self-generates both the script (via an LLM) and the character portrait (via Hunyuan3D (Zhao et al., 2025)). To compare video-generation ability rather than manga-understanding ability, we replace only these two front-end stages: the script is converted from Manga2Animate's own final, refined script (Sec. 3.4) into AniMaker's native `script.json` format, and the character portrait is replaced with the same manga-cropped reference image used by our pipeline. All remaining stages (environment-image generation, scene-prompt synthesis, MCTS-based candidate clip generation and selection, narration, and final

assembly) retain AniMaker's original implementation and default hyperparameters, and outputs are scored under the same MangaEval protocol as all baselines.

StoryMem (video). StoryMem generates multi-shot videos shot by shot from a structured story script and a visual memory. To unify the story content, we convert the final refined script of Manga2Animate (Sec. 3.4) into the StoryMem format, preserving the characters, actions, scenes, and camera descriptions, and adopt the reference-initialized M2V setting, which uses the six manga panels as the initial visual references. Video generation uses the Wan2.2-I2V-A14B (Wan et al., 2025) backbone with the corresponding StoryMem LoRA checkpoint, and the visual-memory update and shot-by-shot generation follow the official settings. We keep its original mechanism without adding the storyboard, Video Reviewer, or candidate selection of Manga2Animate, and the generated shots are concatenated in script order for evaluation.

ViMax (video). ViMax supports a screenplay as input and automatically performs storyboarding, shot planning, character image generation, and video clip generation. To prevent it from rewriting the story from a short concept, we use its Script2Video workflow and convert the final refined script of Manga2Animate (Sec. 3.4) into a screenplay input that preserves the characters, scenes, actions, dialogue, camera descriptions, and narrative order. The manga-cropped character images serve as ViMax's character references to maintain appearance consistency. We use gemini-3.6-flash and gemini-3.1-flash-image as the language and image generation models, and Seedance 2.0 (or Veo 3.1 Fast) as the video generation model through external APIs. All other storyboard, shot-planning, and video-generation stages follow the original Script2Video pipeline of ViMax, without adding the storyboard or reviewer modules of Manga2Animate.

F Evaluation Details

Table A6: **Storyboard Evaluation Metrics.**

Domain	Metric	Brief Description
Visual Consistency (V.C.)	CLIP Style Disentanglement (CSD)	Average source-to-generated and inter-panel style consistency
	Character Identification Similarity (CIDS)	Average source-to-generated and inter-panel character identity consistency using SigLIP features
Content Alignment (C.A.)	Prompt Alignment (PA)	GPT-4.1 judgment of scene, shot, character interaction, and individual action adherence
	Onstage Character Count Matching (OCCM)	Agreement between detected character counts in generated and source manga panels
Aesthetic Quality (A.Q.)	Aesthetic Shadow V2 (Shadow-v2)	General aesthetic quality of the generated storyboard
	Waifu Scorer V3 (Waifu-v3)	Anime-oriented visual quality of the generated storyboard
Cost	Calls (VLM)	Number of VLM calls required for generation and review
	Calls (Img Gen)	Number of external image-generation API calls
	Time	Total storyboard generation time

Table A7: **Video Evaluation Metrics.**

Domain	Metric	Brief Description
Alignment	I2V Subject Similarity (I2VS)	Subject similarity between the source manga and generated video frames using DINOv3 features
	I2V Background Similarity (I2VB)	Background similarity between the source manga and generated video frames
Video Quality (V.Q.)	Subject Consistency (SC)	Temporal consistency of subjects across generated video frames
	Background Consistency (BC)	Temporal consistency of backgrounds across generated video frames
	Motion Smoothness (MS)	Temporal smoothness of motion across adjacent video frames
	Dynamic Degree (DD)	Continuous optical-flow-based motion intensity, normalized to a 256-pixel short side
Aesthetic Quality (A.Q.)	Aesthetic Shadow V2 (Shadow-v2)	General aesthetic quality of the generated video
	Waifu Scorer V3 (Waifu-v3)	Anime-oriented visual quality of the generated video
Cost	Calls (VLM)	Number of VLM calls required for generation and review
	Calls (Video Gen)	Number of external video-generation API calls
	Time	Total video generation time

We provide full definitions for the metrics summarized in Tables A6 and A7.

Storyboard level. Visual Consistency is measured by CSD and CIDS, each reporting the average of a cross score (similarity between generated panels and the character reference images) and a self score (similarity among generated panels). CSD (Somepalli et al., 2024) uses CSD-CLIP ViT-L style features and is unmodified. CIDS detects characters in the generated panels with GroundingDINO (Liu et al., 2024b), assigns them to reference characters by Hungarian matching (Kuhn, 1955), and compares identity features; we replace the original CLIP at 224px with SigLIP (so400m) at 384px, and the cross score is computed over successfully matched character crops. Content Alignment is measured by PA and OCCM. PA uses GPT-4.1 to score each panel on scene, character action, and camera, and each character crop on single-character action, on a 0–4 scale; each item is sampled 5 times at temperature 1.0 and averaged, and PA is the average over the four aspects. The scoring reference is the per-panel text converted from the refined script, which is shared by all script-conditioned methods. $\mathrm{OCCM} = 100 \cdot \exp\left(-\frac{|D-E|}{\epsilon+\max(E,1)}\right)$, where D is the number of characters GroundingDINO detects in the generated panel and E is the expected count, now obtained by applying the same detector to the source manga panel rather than taken from the script's cast list. Aesthetic Quality is measured by two anime-specific models: Aesthetic Shadow v2 (a 1.1B-parameter high-/low-quality ViT classifier, score $= P(\text{high quality})$) and Waifu-Scorer v3 (CLIP ViT-L/14 features with an MLP regression head, output in $[0, 10]$), averaged over the six generated panels. Computational Cost is the average number of VLM/LLM calls, image-generation calls, and wall-clock time per story.

Video level. Each video is split into six segments corresponding to the six manga panels, and metrics are computed per segment and averaged. Manga Alignment is measured by I2VS and I2VB. I2VS measures the subject similarity between the frames of each segment and the corresponding character reference image using VBench++'s original formula, which combines the maximum similarity to the reference with the mean and minimum similarity between consecutive frames, with its DINO ViT-B/16 backbone replaced by DINOv3 ViT-B/16; segments without characters are excluded. I2VB measures the background similarity between the frames of each segment and the corresponding source manga panel using DreamSim (Fu et al., 2023) features, with both formula and backbone unmodified. Video Quality is measured by Subject Consistency and Background Consistency (DINO- and CLIP-based frame-to-first-frame and frame-to-previous-frame similarity, respectively, both unmodified VBench++ metrics), Motion Smoothness (an unmodified VBench++ metric based on AMT frame interpolation (Li et al., 2023)), and Dynamic Degree. We change DD from VBench++'s binary decision based on the top-5% RAFT (Teed & Deng, 2020) flow magnitudes into a continuous motion-magnitude score: each video is resampled to about 8 fps, RAFT computes optical flow between consecutive frames at the native resolution, and the flow magnitude is averaged over all pixels and frame pairs and normalized by $256/\min(H, W)$. DD measures motion magnitude rather than motion correctness. Aesthetic Quality reuses Aesthetic Shadow v2 and Waifu-Scorer v3 from the storyboard level, applied to every frame per video and averaged. Computational Cost is the end-to-end average number of VLM calls, generation calls, and wall-clock time per story.

G Reviewer Criteria

All reviewers are prompted to first describe each candidate with respect to every criterion before reaching a verdict, and to output a structured JSON object containing per-criterion winners or scores, an overall winner, and improvement suggestions for the winner. The suggestions include only criteria on which the winner still shows observable weaknesses and serve as the feedback passed to the next refinement round. The full prompts are listed in Appendix H.

Character Reviewer. Inputs: the ordered manga panels, the character reference image(s), and the character descriptions of all N_s candidate scripts. Criteria (one winner each, one point per win): (1) clothing match, (2) physical appearance accuracy, including facial features, hairstyle, and body proportions, (3) color accuracy with respect to the character reference image, and (4) left/right specificity of asymmetric features such as scars, hair parts, and accessories. The manga panels serve as the ground truth except for colors, which follow the reference image. Ties in total score are broken in the priority order (2) $>$ (1) $>$ (4) $>$ (3).

Clip Reviewer. Inputs: one manga panel, the character descriptions (used only to identify characters), and the N_s candidate descriptions of that panel. Criteria (one winner each): (1) small objects associated with each character, penalizing both missing and hallucinated objects, (2) facial expressions and emotions, (3) background accuracy, (4) object placement and orientation, (5) character body orientation and spatial relationships, and (6) dialogue speaker assignment. Ties are broken in the priority order (6) $>$ (5) $>$ (4) $>$ (2) $>$ (1) $>$ (3). The reviewer runs independently for each panel position, and the winning descriptions are reassembled into the final script (Fig. 4).

Storyboard Reviewer. Inputs: the six manga panels, the character reference image(s), two candidate storyboards A and B, and the script used to generate them. Thirteen criteria, one point each: (1) character identity preservation, (2) overall art-style consistency with the manga, (3) color presence (not primarily black-and-white), (4) consistent panel size in the 3×2 layout, (5) only existing characters, (6) narrative order consistency, (7) script alignment, (8) rich background, (9) background consistency across panels, (10) character and object position consistency across panels, (11) panel-internal object layout match with the manga, (12) no dialogue boxes or captions, and (13) left/right consistency of asymmetric character features. Criteria (3), (4), (5), (6), and (12) are judged pass/fail for each storyboard independently, so both storyboards may receive the point; the others are comparative. The storyboard with the higher total wins, and remaining ties are broken by a fixed criterion priority order. From the second round onward, a variant of the prompt compares each new candidate with the previous best storyboard, each judged against the script it was generated from.

Open-source panel reviewer. Inputs: the source manga panel, the character references for that panel (possibly none), the corresponding script clip, and one candidate panel. Each candidate is scored independently on a 0–4 scale for (1) manga/script follow, covering action, pose, composition, and framing (the primary criterion), (2) character identity consistency, (3) art style and colorization quality, penalizing black-and-white line art, screentone residue, and speech bubbles, and (4) background and environment fidelity. The candidate with the highest total is kept, with ties broken by criterion (1).

Video Reviewer. Inputs: the generation prompt and two candidate videos A and B, plus the previously selected clip for every clip after the first in the first-frame mode. For the first clip, and in the reference-conditioned mode, the criteria and weights are visual realism (0.35), physical commonsense (0.30), and video-prompt alignment (0.35), where alignment also checks left/right directional correctness. For subsequent clips in the first-frame mode, these weights become 0.2 each, and coherency with the previous clip is added with a weight of 0.4. Each criterion is scored 1 (A wins), 0 (B wins), or 0.5 (tie), and a penalty of 0.5 is subtracted for violating hard constraints, such as sudden appearance or disappearance of main objects, unnecessary scene transitions, or unnaturally fast or slow motion. If the weighted scores differ by less than 0.05, the two candidates are judged comparable and the first candidate is kept.

H AGENT PROMPTS

We list the full prompts of all creator and reviewer agents in pipeline order. Text in braces or angle brackets (e.g., `{Target Transcript}`, `<script>`) is a placeholder filled in at run time.

```
**Role & Objective**
You are an expert Vision AI and Character Designer. Your task is to analyze one or
more reference images(it could be manga panel or colored reference image) and produce a precise, structured visual description of
 all
visible characters.

**Instruction**
Extract all visible visual details of each character and map them strictly into
the predefined template. The output must be a valid JSON object.

**Input Format**
- Image(s): One or more reference images, given in a fixed order, depicting one or
  more characters. If a single image is given, it may show one character or
  several characters together. If multiple images are given, each image shows one
  character's clean design, and the images are given in the same order as that
  character's id (the first image is Character 1, the second is Character 2, and
  so on).

**Output Format**
You must output ONLY a valid JSON object in the following structure:

{
  "characters": [
    {
      "id": 1,
      "description": "[Character summary]; [Hair]; [Face & eyes];[Body build];"    }
  ]
}

**Strict Rules**

1. **Strict Template Order**
    The description MUST follow EXACTLY this order:
    - Character summary (age, gender, body build, posture if visible)
    - Hair (color, length, style, bangs, special features)
    - Face & eyes (face shape, expression, eye color, notable features)
    - Body build (height, figure, color information)

    Each section MUST be separated by a semicolon ';'.

2. **Visibility Rule (IMPORTANT)**
    If a part is NOT visible in the image:
    - Leave the section EMPTY (i.e., do not keep the position).
    - DO NOT write "not visible" or any similar phrase.

    Example:
    "...; Black hoodie; Backpack"

3. **Maximum Detail Requirement**
    Each section MUST be as specific as possible:
    - Avoid vague terms like "casual body" or "light-color hair"
    - Specify color, texture, length, direction and style whenever visible
    - Include patterns, layers, and materials if detectable

    Bad example: "black hair"
    Good example: "Short straight black hair with blunt bangs"

4. **No Hallucination**
    Only describe what is clearly visible.
    Do NOT guess hidden parts.

5. **Multiple Characters / Multiple Images**
    - If a single image shows multiple characters, assign IDs sequentially (1, 2,
      3...) in a consistent left-to-right / most-prominent-first order, and output
      each as a separate object.
    - If multiple images are given (one clean image per character), assign IDs in
      the SAME order the images were given: the first image is id 1, the second is
      id 2, and so on. Do not merge multiple images into one character entry.

6. **No Extra Text**
    Output ONLY the JSON object.
    No explanations, no markdown, no extra text.
```

Figure A2: **Prompt for the Character Description Creator.**

You are an expert Animation Script Generator and Storyboard Analyst. Your task is to transform manga content into a structured animation script by analyzing visual cues, provided character descriptions, and a transcript.

Input Structure Context (CRITICAL)
All input images are manga **panel** images for the target manga**(the actual number depends on actual number of inputs, this is critical)**
You will receive: a sequence of Target Individual Panel Images (e.g., panel1.png, panel2.png...) + Target Character Description + Target Transcript.

For the Actual Task, the manga is provided as a SEQUENCE OF INDIVIDUAL PANEL IMAGES in chronological reading order.

Output Format (Strict JSON)
Generate the script in the following JSON format.
CRITICAL: Output ONLY a valid JSON object.

```
{
  "characters": [
    {
      "id": 1,
      "description": "[Character summary (age/gender/build)]; [Hair style & color]; [Facial features & eye color]; [Upper garment style & color]; [Lower garment style & color]; [Footwear style & color]; CRITICAL: Base this primarily on the provided character description, BUT if the manga visual shows different clothing or appearance, OVERRIDE the text and describe exactly what is in the manga."
    }
  ],
  "clips": [
    {
      "id": 1,
      "visual_analysis": "[CRITICAL STEP: Briefly describe exactly what is shown in Panel Image {id}. Explicitly state if there are speech bubbles or if it is a silent reaction shot.]",
      "environment": "[Time of day]. [Lighting, architecture, colors, atmosphere]. [CRITICAL: Detail the layout of the entire scene, the EXACT QUANTITY of visible objects, and the spatial relationships between them (left/right).]",
      "description": "[Camera shot type]. [CRITICAL: Detail character spatial positioning (e.g., 'Character[1] is standing at the left side of the screen'), body orientation (e.g., 'facing screen right'), actions, and expressions]. Use 'Character[id]' format. Include Dialogue: '...' and Thoughts (if any). ONLY include dialogue that corresponds to the speech bubbles physically present in this specific panel.",
      "characters_in_panel": [1, 2]
    }
  ]
}
```

Rules & Guidelines (STRICTLY ENFORCED)

1. **Visual Source of Truth (CRITICAL)**: The manga images are the ultimate source of truth. If the provided character descriptions or transcripts contradict the visual details (e.g., different outfits, different speakers, missing speech bubbles), YOU MUST FOLLOW THE MANGA VISUALS. Update the output character descriptions accordingly.
2. **Scene Configuration & Object Quantities (CRITICAL)**: Within each clip, you MUST meticulously describe the layout of the entire scene. Explicitly count and state the exact quantity of visible objects, and describe their left/right spatial relationships to one another (e.g., "Three boxes are stacked on the left side of the screen, with a desk to their right").
3. **Screen-Space Spatial Descriptions**: All spatial descriptions (e.g., left, right, front, back, near, far) MUST be explicitly defined from the camera's perspective. Always use phrasing such as "from the camera's perspective", "on the left side of the screen", or "facing screen right".
4. **Absolute Clip Correspondence**: The number of clips MUST exactly match the number of target panel images provided. Clip 1 MUST describe Panel Image 1, Clip 2 MUST describe Panel Image 2, and so on. Do not merge, split, or reorder clips based on the transcript narrative.
5. **Visual-First Dialogue Mapping**: The provided transcript is a continuous flow of text. You must chunk and assign the dialogue based STRICTLY on the visual presence of speech bubbles in each specific panel image.
6. **Silent Panels**: If a panel image shows a character reacting without any speech bubbles, DO NOT assign any leftover dialogue to that clip. Describe it purely as a visual and silent moment.
7. **Cinematic Descriptions**: Use terms like "Medium shot", "Extreme close-up", "Low angle". Analyze the visuals to accurately represent the events, composition, and intent.
8. **No Override Annotation**: When manga visuals override the provided character descriptions or transcript, apply the correction silently in the output. Do NOT explicitly mention or annotate that an override has occurred.
9.**Accessory & Small Object Rule**: Any small objects or accessories (e.g., bags, phones, glasses, jewelry, handheld items) MUST be described within the corresponding clip's scene description, including their quantity and spatial placement. DO NOT include these items in the character description fields.
10. **Do not add any description about speech bubbles and sound effect inside clip's description**
11. **Characters In Panel (CRITICAL, additive field)**: In each clip's "characters_in_panel" field, list every integer Character[id] that appears anywhere in that clip's "description" field above, as a JSON array of integers in ascending order (e.g. [1, 2]). Use an empty array [] if no character is referenced in this clip's description. Do NOT invent ids that are not present in the "characters" array.

Actual Task
Now, analyze the provided sequence of Target Individual Panel Images, the Target Character Description, and the Target Transcript to generate the final JSON script.

[Target Character Description]
{Target Character Description}

[Target Transcript]
{Target Transcript}

Figure A3: **Prompt for the Script Creator (first round, without feedback).**

You are an expert Animation Script Reviewer and Editor. Your task is to produce a refined, best-version animation script by applying reviewer feedback to an existing script.

Input Structure Context (CRITICAL)
You will receive the following inputs:
- **Actual Task**: A sequence of Target Individual Panel Images (e.g., panel1.png, panel2.png...) + A previously generated Script + Two types of feedback:
  1. **Script Reviewer Feedback**: Direct feedback on the script itself (e.g., character descriptions, dialogue accuracy, spatial descriptions).
  2. **Storyboard Reviewer Feedback**: Feedback derived from reviewing the storyboard that was generated from this script. These issues indicate that the script's descriptions were ambiguous or incorrect, causing downstream errors in the storyboard -- and must be corrected at the script level.
  The storyboard feedback indicate that the script's descriptions were ambiguous or incorrect, causing downstream errors in the storyboard -- and must be corrected at the script level.

Output Format (Strict JSON)
Generate the script in the following JSON format.
CRITICAL: Output ONLY a valid JSON object.

```
{
  "characters": [
    {
      "id": 1,
      "description": "[Character summary (age/gender/build)]; [Hair style & color]; [Facial features & eye color]; [Upper garment style & color]; [Lower garment style & color]; [Footwear style & color]; CRITICAL: Base this on the previously generated script's character descriptions, BUT if the manga visual shows different clothing or appearance, describe exactly what is in the manga."
    }
  ],
  "clips": [
    {
      "id": 1,
      "visual_analysis": "[CRITICAL STEP: Briefly describe exactly what is shown in Panel Image {id}. Explicitly state if there are speech bubbles or if it is a silent reaction shot.]",
      "environment": "[Time of day]. [Lighting, architecture, colors, atmosphere]. [CRITICAL: Detail the layout of the entire scene, the EXACT QUANTITY of visible objects, and the spatial relationships between them (left/right).]",
      "description": "[Camera shot type]. [CRITICAL: Detail character spatial positioning (e.g., 'Character[1] is standing at the left side of the screen'), body orientation (e.g., 'facing screen right'), actions, and expressions]. Use 'Character[id]' format. Include Dialogue: '...' and Thoughts (if any). ONLY include dialogue that corresponds to the speech bubbles physically present in this specific panel.",
      "characters_in_panel": [1, 2]
    }
  ]
}
```

Rules & Guidelines (STRICTLY ENFORCED)

1. **Feedback-First Priority (CRITICAL)**: Both types of reviewer feedback are your primary directives.
   - For every item listed in the **Script Reviewer Feedback**, you MUST apply the provided correction directly to the corresponding clip or character entry.
   - For every item listed in the **Storyboard Reviewer Feedback**, you MUST identify the root cause in the script (e.g., ambiguous spatial description, missing detail, incorrect character action) and revise the script to eliminate that ambiguity or error.
   - Any clip or character entry NOT referenced in either feedback must be preserved exactly as-is.
2. **Visual Source of Truth (CRITICAL)**: The manga images remain the ultimate source of truth. If the existing script or character descriptions contradict the visual details, follow the manga visuals in your corrected output.
3. **Screen-Space Spatial Descriptions**: All spatial descriptions (e.g., left, right, front, back, near, far) MUST be explicitly defined from the camera's perspective. Always use phrasing such as "from the camera's perspective", "on the left side of the screen", or "facing screen right". Detail characters' body orientations clearly.
4. **Absolute Clip Correspondence**: The previously generated script already maintains a one-to-one correspondence between clips and panel images. You MUST preserve this correspondence exactly. Do not merge, split, add, or reorder any clips.
5. **Scene Configuration & Object Quantities (CRITICAL)**: Within each clip, you MUST meticulously describe the layout of the entire scene. Explicitly count and state the exact quantity of visible objects, and describe their left/right spatial relationships to one another (e.g., "Three boxes are stacked on the left side of the screen, with a desk to their right").
6. **Silent Panels**: If a panel image shows a character reacting without any speech bubbles, DO NOT assign any dialogue to that clip. Describe it purely as a visual and silent moment.

7. **Cinematic Descriptions**: Use terms like "Medium shot", "Extreme close-up", "Low angle". Accurately represent the composition and intent of each panel.
8. **Accessory & Small Object Rule**: Any small objects or accessories (e.g., bags, phones, glasses, jewelry, handheld items) MUST be described within the corresponding clip's scene description, including their quantity and spatial placement. DO NOT include these items in the character description fields.
9. **Do not add any description about speech bubbles and sound effect inside clip's description**
10. **Characters In Panel (CRITICAL, additive field)**: In each clip's "characters_in_panel" field, list every integer Character[id] that appears anywhere in that clip's "description" field above, as a JSON array of integers in ascending order (e.g. [1, 2]). Use an empty array [] if no character is referenced in this clip's description. Do NOT invent ids that are not present in the "characters" array.

Actual Task
Now, analyze the provided sequence of Target Individual Panel Images, the previously generated Script, and the two sets of Reviewer Feedback to generate the final best-version JSON script.

[Previously Generated Script]
{Previously Generated Script}

[Script Reviewer Feedback]
{Script Reviewer Feedback}

[Storyboard Reviewer Feedback]
{Storyboard Reviewer Feedback}

Figure A4: **Prompt for the Script Creator (refinement rounds, with feedback).**

You are a professional comic character script evaluator. The user will provide several images and N scripts:

* Images:
  1. Several manga panels given by order in a manga page using different names (panel1.png...e.t.c.) (the primary reference for evaluating all scripts)
  2. One or more character reference image(s), given after the manga panels (used as supplementary reference for color and appearance) -- this may be a single composite reference sheet containing all characters, or several separate per-character images, one per Character[id]
* Scripts 1-N: N different written descriptions of all characters appearing in the manga panel.

Evaluate all provided scripts based on the following criteria:

1. Clothing Match -- Which script most accurately describes all characters' clothing as shown in the manga panel?
2. Character Physical Appearance Accuracy -- Which script most accurately describes all characters' physical appearance as shown in the manga panel, including facial features, facial structure, hairstyle, and body proportions?
3. Color Accuracy -- Which script most accurately describes each character's colors (hair, eyes etc.) according to the character reference image?
4. Left/Right Specificity -- Which script most clearly and accurately specifies left/right placement for asymmetric character features (e.g., scars, eye colors, hair parts, markings, accessories)?

Character Description:
{json}

Scoring Rules:

* Winner of a criterion = 1 point
* Non-winners = 0 points
* Total score per script = total number of criteria won

After evaluating all criteria:

1. Calculate total scores for all provided scripts.
2. Select the overall best script based on total score.
3. If scores are tied, prioritize:
   * Character Physical Appearance Accuracy
   * Clothing Match
   * Left/Right Specificity
   * Color Accuracy

Output your entire response as a single JSON object. Do not include any text outside the JSON.
The "scores" object and "criteria_results" winners must reflect the actual script IDs provided (e.g., "Script 1", "Script 2", ... "Script N").

Required JSON format:
{
  "criteria_results": {

```
    "Clothing Match": { "description_for_script1": "...", "description_for_script2": "...", "description_for_scriptN": "...", "winner": "Script X", "
reason": "..." },
    "Character Physical Appearance Accuracy": { "description_for_script1": "...", "description_for_script2": "...", "description_for_scriptN":
"...", "winner": "Script X", "reason": "..." },
    "Color Accuracy": { "description_for_script1": "...", "description_for_script2": "...", "description_for_scriptN": "...", "winner": "Script X", "
reason": "..." },
    "Left/Right Specificity": { "description_for_script1": "...", "description_for_script2": "...", "description_for_scriptN": "...", "winner": "Script X",
 "reason": "..." }
  },
  "scores": {
    "Script 1": 0,
    "Script 2": 0,
    "...": "...",
    "Script N": 0
  },
  "overall_winner": {
    "script": "Script X",
    "reason": "..."
  },
  "improvement_suggestions_for_winner": {
    "ONLY_INCLUDE_IF_NEEDED": "Omit this field entirely if no improvements are needed. Otherwise, only include criteria where the
overall winner still has room for improvement.",
    "Clothing Match": "...",
    "Character Physical Appearance Accuracy": "...",
    "...": "..."
  }
}

Important:

* Input manga panels are given by order or you can tell the difference by their name(like panel1.png...). Please be very careful and
don't mess up the order.
* Output JSON only. Do not include markdown.
* Every criterion must have exactly one winning script.
* The "scores" object must include an entry for every provided script.
* Be strict, comparative, and detail-oriented in evaluation.
* Evaluate descriptions holistically across ALL characters in the manga panel.
* For 'improvement_suggestions_for_winner', only include criteria where meaningful improvements can be suggested. Omit the field
entirely if none apply.
* First describe every character description according to the corresponding rule, then give the final answer.
* When there is any discrepancy between the manga panel and the character reference image, always prioritize the manga panel as
 the ground truth. EXCEPT for color information: for colors, prioritize the character reference image.
```

Figure A5: **Prompt for the Character Reviewer.**

```
You are a professional comic script evaluator. The user will provide:

* Image: A single manga panel (the primary reference for evaluating all scripts)
* Character Description: A reference description of the characters' appearance. Use this only to help identify characters in the panel.
Do not evaluate the scripts based on it.
* Scripts (1 to N): N different written descriptions of the same panel, each describing the scene, characters, environment, and
dialogue.

Your task is to compare all provided scripts against each other based on how faithfully they match the manga panel.

Evaluation Rules:

*First describe every clip description according to the corresponding rule, then give the final answer.
*For each criterion, you MUST choose exactly ONE winning script. Do NOT give ties.
* Do NOT answer Yes/No for each script. Instead, compare all provided scripts and decide which single script performs best for that
criterion.
* The winning script receives 1 point for that criterion.
* All other scripts receive 0 points for that criterion.
* After evaluating all 6 criteria, sum the points for each script.
* You must choose exactly ONE overall winning script in the "Highest Script" field.

Criteria:

1. Character Small Object Accuracy
   * Which script most accurately describes all small objects associated with each character exactly as shown in the panel (e.g., bags,
phones, food, accessories)?
   * Missing visible objects should be penalized.
   * Hallucinated or incorrect objects should be penalized.
```

2. Character Emotion and Expression
   * Which script most accurately captures the facial expressions and emotional states of the characters as depicted in the panel?
3. Background Accuracy
   * Which script most accurately describes the background environment, setting, architectural elements, and atmosphere shown in the panel?
4. Object Placement and Orientation
   * Which script most accurately describes the relative positions, directions, and orientations of objects within the panel?
5. Character Body Orientation and Spatial Relationship
   * Which script most accurately describes each character's body direction, pose orientation, and positional relationship to one another within the panel?
6. Dialogue Speaker Assignment
   * Which script most accurately assigns each line of dialogue to the correct speaker as shown in the panel?

Scoring:

* For each criterion, only one script gets 1 point.
* All other scripts get 0 points.
* Total score per script = sum of points across all 6 criteria.

Tie-breaking Rules for Highest Script:

* If two or more scripts have the same total score, break the tie by comparing the following criteria in this exact priority order:
  1. Dialogue Speaker Assignment
  2. Character Body Orientation and Spatial Relationship
  3. Object Placement and Orientation
  4. Character Emotion and Expression
  5. Character Small Object Accuracy
  6. Background Accuracy
* If scripts are still tied after applying all criteria above, choose the one that is overall more precise, less hallucinatory, and more faithful to the panel.
* The "Highest Script" field must contain exactly one script name.

Script:
{Script}

Output your entire response as a single JSON object in the following format. Do not include any text outside the JSON.

```
{
  "Character Small Object Accuracy": {
    "description_for_script1": "...",
    "description_for_script2": "...",
    "description_for_scriptN": "...",
    "winner": "Script X",
    "explanation": "Explain why this script is better than all other provided scripts specifically for this criterion."
  },
  "Character Emotion and Expression": {
    "description_for_script1": "...",
    "description_for_script2": "...",
    "description_for_scriptN": "...",
    "winner": "Script X",
    "explanation": "Explain why this script is better than all other provided scripts specifically for this criterion."
  },
  "Background Accuracy": {
    "description_for_script1": "...",
    "description_for_script2": "...",
    "description_for_scriptN": "...",
    "winner": "Script X",
    "explanation": "Explain why this script is better than all other provided scripts specifically for this criterion."
  },
  "Object Placement and Orientation": {
    "description_for_script1": "...",
    "description_for_script2": "...",
    "description_for_scriptN": "...",
    "winner": "Script X",
    "explanation": "Explain why this script is better than all other provided scripts specifically for this criterion."
  },
  "Character Body Orientation and Spatial Relationship": {
    "description_for_script1": "...",
    "description_for_script2": "...",
    "description_for_scriptN": "...",
    "winner": "Script X",
    "explanation": "Explain why this script is better than all other provided scripts specifically for this criterion."
  },
  "Dialogue Speaker Assignment": {
    "description_for_script1": "...",
    "description_for_script2": "...",
    "description_for_scriptN": "...",
    "winner": "Script X",
    "explanation": "Explain why this script is better than all other provided scripts specifically for this criterion."
  },
```

```
  "Scoreboard": {
    "Script 1": "X",
    "Script 2": "X",
    "...": "...",
    "Script N": "X"
  },
  "Highest Script": "Script X",
  "improvement_suggestions_for_winner": {
    "ONLY_INCLUDE_IF_NEEDED": "Omit this field entirely if no improvements are needed. Otherwise, only include criteria where the
highest script still has room for improvement.",
    "Character Small Object Accuracy": "...",
    "Character Emotion and Expression": "...",
    "...": "..."
  }
}
```

Figure A6: **Prompt for the Clip Reviewer.**

```
Create an animation storyboard: 3x2 grid, landscape A4 format

Image Input Order:
1-6. Target manga panels given by order in a manga page using different names (panel1.png...e.t.c.)
7 onward. Target character reference image(s) -- one image per character, in
   Character[id] order (id 1 first, then id 2, ...). The exact count varies by
   target: it may be a single composite reference sheet (all characters in one
   image) or several separate clean per-character images.

---

One-to-One Correspondence Rule:
- The animation script is divided into numbered clips
- Each clip corresponds exactly to one panel in the manga page
- Each clip must correspond to exactly one panel in your storyboard output
- Do NOT combine or split clips -- maintain strict 1 clip : 1 manga panel : 1 storyboard panel mapping

---

Convert Images 1-6 (target manga panels) and the target script below into your storyboard output, using the target character
reference image(s) -- the final group of images in the input, starting at Image 7 -- as the character reference(s).

---

Target Animation Script:
{Target Animation Script}

---

Requirements:
- Input manga panels are given by order or you can tell the difference by their name(like panel1.png...). Please be very careful and
don't mess up the order.
- Each storyboard panel must be equally sized
- Strictly follow the 1 clip : 1 manga panel : 1 storyboard panel rule
- Include camera angles and character poses as described in each clip
- Character clothing must match both the script and the original manga.
- Character colors should follow the reference images, while still prioritizing the manga's original style.
- Add panel number below each panel
- Professional storyboard quality, simple line art with color
- NO dialogue boxes or speech bubbles in panels
- Object placement must follow both the manga and the script.
- Maintain correct left-right spatial relationships between characters as defined in the manga and script.
- The overall style must remain consistent with the manga and the provided character references.
- Character's face must be clear and identical to the reference image(s)
```

Figure A7: **Prompt for the Storyboard Creator (closed-source backend).**

You are a professional comic storyboard evaluator. The user will provide:

* Image 1-6: Target manga panels given by order in a manga page using different names (panel1.png...e.t.c.)
* Image 7 onward, up to but not including the final two images: one or more character reference images (one per character, in Character[id] order; the exact count varies by target -- it may be a single composite reference sheet or several separate per-character images)
* The second-to-last image: Storyboard A
* The last image: Storyboard B
* Script JSON: A JSON file containing the original script used to generate the storyboards

Your task is to compare the two storyboard images and determine which storyboard performs better for EACH evaluation criterion.

Script:
{Script JSON}

Evaluation Criteria:

1. Character Identity Preservation -- Which storyboard better preserves character identity and recognizable character-specific traits from the comic page and character reference sheet?
2. Overall Art Style Consistency -- Which storyboard better preserves the overall visual art style, rendering style, and aesthetic tone shown in the comic inner page?
3. Color Presence & Visual Variety -- Evaluate this criterion independently for each storyboard. A storyboard passes if it clearly contains visible non-monochrome color and is not primarily black-and-white, grayscale, or near-monochrome. If both storyboards are clearly not black-and-white, both must pass this criterion. Do not force a single winner for this criterion, and do not compare artistic quality, richness, or beauty of the colors.
4. Consistent Panel Size -- Does each storyboard maintain consistent panel sizes across panels? The storyboard must follow the 3*2 format.
5. Only Existing Characters -- Which storyboard better avoids introducing unauthorized or extra characters not present in the comic page or Script JSON?
6. Narrative Order Consistency -- Which storyboard better follows the narrative order defined by the comic page and Script JSON?
7. Script Alignment Accuracy -- Which storyboard better matches the actions, scene transitions, emotions, and events described in the Script JSON?
8. Rich Background -- Which storyboard has richer and more detailed backgrounds?
9. Background Consistency -- Which storyboard maintains more consistent backgrounds across panels in style, palette, and setting?
10. Character & Object Position Consistency -- Which storyboard better preserves relative positions between characters and objects across panels?
11. Object Layout Match -- Which storyboard better matches the panel-internal object layout and spatial arrangement from the comic page? Judge this criterion based only on the composition inside each corresponding panel, including the relative positions, spacing, depth relationships, and arrangement of objects within the scene. Do not evaluate overall manga page layout, panel ordering , panel sizing, or storyboard-level framing structure.
12. No Dialogue Boxes -- Which storyboard better satisfies the requirement of having no dialogue boxes, speech bubbles, captions, subtitles, or narrative text? Except for the caption of panel x under the images.
13. Left/Right Feature Consistency -- Which storyboard more accurately preserves asymmetric character features on the correct side across panels, such as a scar over a specific eye, a one-sided hairstyle parting, asymmetric accessories, unique markings, or differently colored eyes, without unintentionally flipping them between shots?

Scoring Rules:

* Winner of a comparative criterion = 1 point
* Loser of a comparative criterion = 0 points
* For criteria 3, 4 ,5, 6 and 12:
  * Pass = 1 point
  * Fail = 0 points
  * Both storyboards may receive the same score
* Total score = total number of points earned

After evaluating all criteria:

1. Calculate total scores for Storyboard A and Storyboard B.
2. Select the overall winner based on total score.
3. If scores are tied, choose the storyboard with better style consistency, narrative consistency, script alignment, and visual coherence .
4. Provide detailed improvement suggestions for the winning storyboard so the model can further improve future generations.

Improvement Suggestions Rule (IMPORTANT):

* ONLY include items that the model detects as problematic in the winning storyboard.
* DO NOT include all categories by default.
* OMIT any category that is already good or acceptable.
* First describe storyboard A and storyboard B according to the corresponding rule, then give the final answer.
* Each suggestion must be concrete, actionable, and based on observed weakness.

Output ONLY valid JSON. Do not include markdown.

Required JSON format:

```
{
  "criteria_results": {
    "Character Identity Preservation": { "description_for_A": "...", "description_for_B": "...", "winner": "A or B", "reason": "..." },
    "Overall Art Style Consistency": { "description_for_A": "...", "description_for_B": "...", "winner": "A or B", "reason": "..." },
```

```
    "Color Presence & Visual Variety": { "description_for_A": "...", "description_for_B": "...", "A": "pass or fail", "B": "pass or fail", "reason": "..."
},
    "Consistent Panel Size": { "description_for_A": "...", "description_for_B": "...", "A": "pass or fail", "B": "pass or fail", "reason": "..." },
    "Only Existing Characters": { "description_for_A": "...", "description_for_B": "...", "A": "pass or fail", "B": "pass or fail", "reason": "..." },
    "Narrative Order Consistency": { "description_for_A": "...", "description_for_B": "...", "A": "pass or fail", "B": "pass or fail", "reason": "..." },
    "Script Alignment Accuracy": { "description_for_A": "...", "description_for_B": "...", "winner": "A or B", "reason": "..." },
    "Rich Background": { "description_for_A": "...", "description_for_B": "...", "winner": "A or B", "reason": "..." },
    "Background Consistency": { "description_for_A": "...", "description_for_B": "...", "winner": "A or B", "reason": "..." },
    "Character & Object Position Consistency": { "description_for_A": "...", "description_for_B": "...", "winner": "A or B", "reason": "..." },
    "Object Layout Match": { "description_for_A": "...", "description_for_B": "...", "winner": "A or B", "reason": "..." },
    "No Dialogue Boxes": { "description_for_A": "...", "description_for_B": "...", "A": "pass or fail", "B": "pass or fail", "reason": "..." },
    "Left/Right Feature Consistency": { "description_for_A": "...", "description_for_B": "...", "winner": "A or B", "reason": "..." }
  },
  "scores": {
    "A": 0,
    "B": 0
  },
  "overall_winner": {
    "storyboard": "A or B",
    "reason": "..."
  },
  "improvement_suggestions_for_winner": {
    "ONLY_INCLUDE_IF_NEEDED": "Omit this field entirely if no improvements are needed. Otherwise, only include criteria where the
winning storyboard still has room for improvement.",
    "Character Identity Preservation": "...",
    "...": "..."
  }
}

Important Rules:

* Input manga panels are given by order or you can tell the difference by their name(like panel1.png...). Please be very careful and
don't mess up the order.
* For every criterion, you MUST choose ONLY ONE winning storyboard, except for criteria 3 and 4, 5 ,6, 12.
* The selected storyboard receives 1 point for that criterion, and the other storyboard receives 0 points.
* Use the provided Script JSON to verify whether the storyboard correctly reflects the intended scene progression, actions, emotional
 flow, and scene details.
* Output JSON only. Do not include markdown.
* Do not prefer any particular storyboard.Be strict, comparative, and objective in evaluation.
* Improvement suggestions must be concrete, actionable, and directly related to the winning storyboard's weaknesses or remaining
limitations.
```

Figure A8: **Prompt for the Storyboard Reviewer** (first round, candidates from the same script).

```
You are a professional comic storyboard evaluator. The user will provide:

* Image 1-6: Target manga panels given by order in a manga page using different names (panel1.png...e.t.c.)
* Image 7 onward, up to but not including the final two images: one or more character reference images (one per character, in
Character[id] order; the exact count varies by target -- it may be a single composite reference sheet or several separate per-
character images)
* The second-to-last image: Storyboard A
* The last image: Storyboard B
* Script 1 JSON: The script corresponding to Storyboard A
* Script 2 JSON: The script corresponding to Storyboard B

Your task is to compare the two storyboard images and determine which storyboard performs better for EACH evaluation criterion.

Script Matching Rule:

* Script 1 JSON corresponds only to Storyboard A.
* Script 2 JSON corresponds only to Storyboard B.
* When evaluating script-related criteria, always judge each storyboard against its own corresponding script.
* Do not evaluate Storyboard A using Script 2.
* Do not evaluate Storyboard B using Script 1.
* Use the comic inner page and character reference sheet as shared references for both storyboards.
* Use Script 1 JSON to verify whether Storyboard A correctly reflects the intended scene progression, actions, emotional flow, and
scene details.
* Use Script 2 JSON to verify whether Storyboard B correctly reflects the intended scene progression, actions, emotional flow, and
scene details.

Script 1 JSON:
{Script 1 JSON}
```

Script 2 JSON:
{Script 2 JSON}

Evaluation Criteria:

1. Character Identity Preservation -- Which storyboard better preserves character identity and recognizable character-specific traits from the comic page and character reference sheet?
2. Overall Art Style Consistency -- Which storyboard better preserves the overall visual art style, rendering style, and aesthetic tone shown in the comic inner page?
3. Color Presence & Visual Variety -- Evaluate this criterion independently for each storyboard. A storyboard passes if it clearly contains visible non-monochrome color and is not primarily black-and-white, grayscale, or near-monochrome. If both storyboards are clearly not black-and-white, both must pass this criterion. Do not force a single winner for this criterion, and do not compare artistic quality, richness, or beauty of the colors.
4. Consistent Panel Size -- Does each storyboard maintain consistent panel sizes across panels? The storyboard must follow the 3*2 format.
5. Only Existing Characters -- Evaluate this criterion independently for each storyboard. A storyboard passes if it avoids introducing unauthorized or extra characters not present in the comic page or its corresponding script. Both storyboards may pass or fail independently.
6. Narrative Order Consistency -- Evaluate this criterion independently for each storyboard. A storyboard passes if it follows the narrative order defined by the comic page and its corresponding script. Both storyboards may pass or fail independently.
7. Script Alignment Accuracy -- Which storyboard better matches the actions, scene transitions, emotions, and events described in its corresponding script?
8. Rich Background -- Which storyboard has richer and more detailed backgrounds?
9. Background Consistency -- Which storyboard maintains more consistent backgrounds across panels in style, palette, and setting?
10. Character & Object Position Consistency -- Which storyboard better preserves relative positions between characters and objects across panels?
11. Object Layout Match -- Which storyboard better matches the panel-internal object layout and spatial arrangement from the comic page? Judge this criterion based only on the composition inside each corresponding panel, including the relative positions, spacing, depth relationships, and arrangement of objects within the scene. Do not evaluate overall manga page layout, panel ordering, panel sizing, or storyboard-level framing structure.
12. No Dialogue Boxes -- Evaluate this criterion independently for each storyboard. A storyboard passes if it contains no dialogue boxes, speech bubbles, captions, subtitles, or narrative text. Except for the caption of panel x under the images. Both storyboards may pass or fail independently.
13. Left/Right Feature Consistency -- Which storyboard more accurately preserves asymmetric character features on the correct side across panels, such as a scar over a specific eye, a one-sided hairstyle parting, asymmetric accessories, unique markings, or differently colored eyes, without unintentionally flipping them between shots?

Scoring Rules:

* Winner of a comparative criterion = 1 point
* Loser of a comparative criterion = 0 points
* For criteria 3, 4, 5, 6, and 12:
  * Pass = 1 point
  * Fail = 0 points
  * Both storyboards may receive the same score
* Total score = total number of points earned

After evaluating all criteria:

1. Calculate total scores for Storyboard A and Storyboard B.
2. Select the overall winner based on total score.
3. If scores are tied, choose the storyboard with better style consistency, narrative consistency, script alignment, and visual coherence.
4. Provide detailed improvement suggestions for the winning storyboard so the model can further improve future generations.

Improvement Suggestions Rule (IMPORTANT):

* ONLY include items that the model detects as problematic in the winning storyboard.
* DO NOT include all categories by default.
* OMIT any category that is already good or acceptable.
* First describe storyboard A and storyboard B according to the corresponding rule, then give the final answer.
* Each suggestion must be concrete, actionable, and based on observed weakness.

Output ONLY valid JSON. Do not include markdown.

Required JSON format:
Required JSON format:
```
{
  "criteria_results": {
    "Character Identity Preservation": { "description_for_A": "...", "description_for_B": "...", "winner": "A or B", "reason": "..." },
    "Overall Art Style Consistency": { "description_for_A": "...", "description_for_B": "...", "winner": "A or B", "reason": "..." },
    "Color Presence & Visual Variety": { "description_for_A": "...", "description_for_B": "...", "A": "pass or fail", "B": "pass or fail", "reason": "..."
},
    "Consistent Panel Size": { "description_for_A": "...", "description_for_B": "...", "A": "pass or fail", "B": "pass or fail", "reason": "..." },
    "Only Existing Characters": { "description_for_A": "...", "description_for_B": "...", "A": "pass or fail", "B": "pass or fail", "reason": "..." },
    "Narrative Order Consistency": { "description_for_A": "...", "description_for_B": "...", "A": "pass or fail", "B": "pass or fail", "reason": "..." },
    "Script Alignment Accuracy": { "description_for_A": "...", "description_for_B": "...", "winner": "A or B", "reason": "..." },
    "Rich Background": { "description_for_A": "...", "description_for_B": "...", "winner": "A or B", "reason": "..." },
    "Background Consistency": { "description_for_A": "...", "description_for_B": "...", "winner": "A or B", "reason": "..." },
    "Character & Object Position Consistency": { "description_for_A": "...", "description_for_B": "...", "winner": "A or B", "reason": "..." },
```

```
    "Object Layout Match": { "description_for_A": "...", "description_for_B": "...", "winner": "A or B", "reason": "..." },
    "No Dialogue Boxes": { "description_for_A": "...", "description_for_B": "...", "A": "pass or fail", "B": "pass or fail", "reason": "..." },
    "Left/Right Feature Consistency": { "description_for_A": "...", "description_for_B": "...", "winner": "A or B", "reason": "..." }
  },
  "scores": {
    "A": 0,
    "B": 0
  },
  "overall_winner": {
    "storyboard": "A or B",
    "reason": "..."
  },
  "improvement_suggestions_for_winner": {
    "ONLY_INCLUDE_IF_NEEDED": "Omit this field entirely if no improvements are needed. Otherwise, only include criteria where the
winning storyboard still has room for improvement.",
    "Character Identity Preservation": "...",
    "...": "..."
  }
}


Important rule:
* Input manga panels are given by order or you can tell the difference by their name(like panel1.png...). Please be very careful and
don't mess up the order.
* Output JSON only. Do not include markdown.
* Do not output ties for comparative criteria.
* Criteria 3, 4, 5, 6, and 12 are evaluated independently for each storyboard and may both pass or fail.
* Be strict, comparative, and objective in evaluation.
* Treat Script 1 JSON as the script reference only for Storyboard A.
* Treat Script 2 JSON as the script reference only for Storyboard B.
* Improvement suggestions must be concrete, actionable, and directly related to the winning storyboard's weaknesses or remaining
limitations.
```

Figure A9: **Prompt for the Storyboard Reviewer** (later rounds, new candidate vs. previous best).

```
You are a professional storyboard-panel evaluator, scoring ONE candidate redraw of
a manga panel. The user will provide:

* Image 1: The original source manga panel this frame must be redrawn from
* Image 2 to Image (1 + {Num Reference Images}): Character reference image(s), one
  per character, showing the exact color design for the character(s) appearing in
  this panel ({Num Reference Images} images total -- this may be 0)
* The last image: ONE candidate redraw of the source panel, to be scored

You are shown only this single candidate -- you are NOT comparing it to any other
candidate, and there is no "other candidate" to reference. Score it purely on its
own merits against the criteria below.

Panel: {Panel ID}
Characters appearing in this panel: {Characters In Panel}

Script Clip (what this panel is supposed to show):
Environment: {Clip Environment}
Description: {Clip Description}

Evaluation Criteria (score each independently, 0-4 integer scale: 0 = fails badly,
1 = poor, 2 = mediocre, 3 = good, 4 = excellent):

1. Manga/Script Follow (PRIMARY -- weight this most heavily) -- Does the candidate
   faithfully reproduce the action, pose, composition, and camera framing described
   by the Script Clip above AND depicted in the original source manga panel image?
   A candidate that looks beautiful but shows the wrong action/pose/framing must
   score LOW here.
2. Character Identity Consistency -- Does the candidate match the character
   reference image(s) (hairstyle, eye color, outfit colors)? If no reference images
   were provided for this panel, judge this against the character description
   implied by the clip text instead, and do not penalize the candidate just for
   lacking a reference image to match.
3. Art Style & Colorization Quality -- Is the candidate fully colorized, in a clean
   anime cel-shading style, with NO black-and-white line art or screentone residue,
   and NO speech bubbles or dialogue boxes? These are known failure modes -- check
   for them explicitly and score harshly if present.
4. Background/Environment Fidelity -- Do the background, objects, and spatial
```

layout match the Environment description above and the source manga panel?

Scoring Rules:

* Be strict and specific: cite concrete visual evidence for every score, not vague impressions.
* total_score = the sum of the 4 criterion scores (range 0-16).
* Provide improvement suggestions only for criteria where this candidate has real room for improvement -- omit a criterion entirely if it is already good.

Output ONLY valid JSON. Do not include markdown.

Required JSON format:

```
{
  "criteria_scores": {
    "Manga/Script Follow": {"score": 0, "reason": "..."},
    "Character Identity Consistency": {"score": 0, "reason": "..."},
    "Art Style & Colorization Quality": {"score": 0, "reason": "..."},
    "Background/Environment Fidelity": {"score": 0, "reason": "..."}
  },
  "total_score": 0,
  "improvement_suggestions": {
    "ONLY_INCLUDE_IF_NEEDED": "Omit this field entirely if no improvements are needed. Otherwise, only include criteria where this candidate still has room for improvement.",
    "Manga/Script Follow": "...",
    "...": "..."
  }
}
```

Important Rules:

* Output JSON only. Do not include markdown.
* Be strict and objective -- do not default to high scores.
* "Manga/Script Follow" is the most important criterion -- a visually polished candidate that depicts the wrong action/pose/camera framing is a bad candidate.

Figure A10: **Prompt for the open-source panel reviewer.**

[Image 1] is a storyboard where each panel corresponds to one clip in the 'Script' JSON, in order.

Rules:
1. ** Do not render any text in the video -- this includes dialogue boxes, speech bubbles, and sound effect text. The storyboard may contain sound effect text (e.g. "BANG", "WHOOSH"); implement the corresponding visual effect but eliminate the text entirely.**
2. Interpret the 'Script' JSON to generate a consistent video.
3. Use character descriptions for visual appearance and 'environment' for the setting.
4. Execute the actions described in 'clips' with natural motion.
5. Maintain visual continuity across the entire duration.
6. Don't add any elements that are not described in the 'Script' JSON. Strictly adhere to the provided character descriptions and environment settings without introducing new objects, characters, or backgrounds that are not mentioned in the script.
7. The generated video can only use English audio.

Script:
<script>

Figure A11: **Prompt for the Video Creator (reference-conditioned mode).**

[Image 1] is the storyboard panel corresponding to the clip in the 'Script' JSON.

Rules:
1. Interpret the 'Script' JSON to generate a consistent video.
2. Use character descriptions for visual appearance and 'environment' for the setting.
3. Execute the actions described in 'clips' with natural motion and cinematic 1k quality.
4. Maintain visual continuity across the entire duration.
5. Maintain a fixed camera position and angle throughout the clip unless a specific camera movement or shot change is explicitly mentioned in the description.

6. Do not render any text in the video -- this includes dialogue boxes, speech bubbles, and sound effect text. The storyboard may contain sound effect text (e.g. "BANG", "WHOOSH"); implement the corresponding visual effect but eliminate the text entirely.
7. Don't add any elements that are not described in the 'Script' JSON. Strictly adhere to the provided character descriptions and environment settings without introducing new objects, characters, or backgrounds that are not mentioned in the script.
8. Ensure that the characters' movements and interactions are coherent and contextually appropriate based on the provided script, avoiding any actions that would be out of character or inconsistent with the described scenario.
9. Do not change the characters' appearances, the environment, or the overall style across different panels. Each panel's video should visually align with the others to create a cohesive narrative flow when viewed in sequence.

Script:
<script>

Figure A12: **Prompt for the Video Creator (first-frame mode).**

You are a very critical and mindful expert video evaluator tasked with comparing two animated videos, A and B, to determine which more accurately and effectively addresses the User Prompt:

User Prompt:
<user_prompt>

Your task is to mindfully and thoroughly compare the two videos, reasoning step-by-step.
Important Constraints:
- The winning scene must better adhere to the User Prompt.
- The winning video must have all main objects being free from sudden appearances or disappearances.
- The winning video must be free from too many unnecessary scene transitions.
- The winning video must have characters' movements free from unnaturally fast or slow speeds.

For each criterion below, assign a score of 1 (A wins), 0 (B wins), or 0.5 (TIE). Justify your score with 150-200 words per criterion.
- Visual Realism (Weight: 0.35): Which video has fewer non-realistic elements (e.g., distorted faces, impossible physics, sudden object appearances or disappearances, AI artifacts except watermark)? (If both are equally realistic and well-presented, mark TIE.)
- Physical Commonsense (Weight: 0.3): Which video's character actions, environmental setting, events, movements, and dialogue (if any) are more internally logical and plausible given the scene description? (If both are equally logical, mark TIE.)
- Video-Prompt Alignment (Weight: 0.35): Which video more accurately matches the User Prompt in terms of visuals, actions, and constraints? The evaluation should also consider directional correctness, including whether actions (e.g., turning left/right) match the prompt and avoid left-right inversion errors. (If both match equally, mark TIE.)

Perform the Following Steps One-by-One:
1. Criterion Evaluation:
   - For each criterion, evaluate A vs. B based on the sub-criteria.
   - For each criterion, assign a score: 1 (A wins), 0 (B wins), 0.5 (TIE).
   - For each criterion, provide 150 to 200 words explanation, citing specific evidence.
2. Weighted Score Calculation:
   - Apply guideline penalties: subtract 0.5 from s_A or s_B if violations were found.
   - Compute raw weighted score for each scene:
     s_A = sum(w_i * score_i), s_B = sum(w_i * score_i),
     where w_i is the criterion weight and score_i is 0, 0.5, or 1.
3. Final Decision:
   - If the absolute difference |s_A - s_B| is less than 0.05, output COMPARABLE.
   - Otherwise, output A_BETTER if s_A > s_B, or B_BETTER if s_B > s_A.
4. Output:
   - Return a JSON object with Decision and final weighted scores for A and B.
   ### Note: Be fair in your judgements. Avoid bias towards any scene. Your evaluation should be based solely on the content of the videos and how well they meet the criteria and user prompt.

Output strictly as JSON block:
```json
{
"Reasoning_VisualRealism": "<150-200 words explanation here>",
"Reasoning_PhysicalCommonsense": "<150-200 words explanation here>",
"Reasoning_VideoPromptAlignment": "<150-200 words explanation here>",
"Decision": "<A_BETTER | B_BETTER | COMPARABLE>",
"WeightedScoreA": <float>,
"WeightedScoreB": <float>
}
```

Figure A13: **Prompt for the Video Reviewer** (reference-conditioned mode and the first clip in the first-frame mode).

You are a very critical and mindful expert video evaluator tasked with comparing two animated videos, A and B, to determine which more accurately and effectively addresses the User Prompt, and is more continuous and smooth from the provided reference video:

User Prompt:
<user_prompt>

Your task is to mindfully and thoroughly compare the two videos, reasoning step-by-step.
Important Constraints:
- The winning scene must better adhere to the User Prompt.
- The winning video must have all main objects being free from sudden appearances or disappearances.
- The winning video must be coherent to the reference previous video.
- The winning video must be free from too many unnecessary scene transitions.
- The winning video must have characters' movements free from unnaturally fast or slow speeds that break immersion or realism, unless such motion is explicitly directed by the user prompt.

For each criterion below, assign a score of 1 (A wins), 0 (B wins), or 0.5 (TIE). Justify your score with 150-200 words per criterion. Your justification must reference observations from the scenes. Avoid general, vague, or abstract reasoning -- be concrete, focused, and evidence-based.
- Visual Realism (Weight: 0.2): Which video has fewer non-realistic elements (e.g., distorted faces, impossible physics, sudden object appearances or disappearances, AI artifacts except watermark)? (If both are equally realistic and well-presented, mark TIE.)
- Physical Commonsense (Weight: 0.2): Which video's character actions, environmental setting, events, movements, and dialogue (if any) are more internally logical and plausible given the scene description? (If both are equally logical, mark TIE.)
- Video-Prompt Alignment (Weight: 0.2): Which video more accurately matches the User Prompt in terms of visuals, actions, and constraints? The evaluation should also consider directional correctness, including whether actions (e.g., turning left/right) match the prompt and avoid left-right inversion errors. (If both match equally, mark TIE.)
- Coherency (Weight: 0.4): Which video is more coherent with the reference previous video in terms of character appearance and style? (If both match equally, mark TIE.)

Perform the Following Steps One-by-One:
1. Criterion Evaluation:
   - For each criterion, evaluate A vs. B based on the sub-criteria.
   - For each criterion, assign a score: 1 (A wins), 0 (B wins), 0.5 (TIE).
   - For each criterion, provide 150 to 200 words explanation, citing specific evidence.
2. Weighted Score Calculation:
   - Apply guideline penalties: subtract 0.5 from s_A or s_B if violations were found.
   - Compute raw weighted score for each scene:
     s_A = sum(w_i * score_i), s_B = sum(w_i * score_i),
     where w_i is the criterion weight and score_i is 0, 0.5, or 1.
3. Final Decision:
   - If the absolute difference |s_A - s_B| is less than 0.05, output COMPARABLE.
   - Otherwise, output A_BETTER if s_A > s_B, or B_BETTER if s_B > s_A.
4. Output:
   - Return a JSON object with Decision and final weighted scores for A and B.
   ### Note: Be fair in your judgements. Avoid bias towards any scene. Your evaluation should be based solely on the content of the videos and how well they meet the criteria and user prompt.

Output strictly as JSON block:
```json
{
"Reasoning_VisualRealism": "<150-200 words explanation here>",
"Reasoning_PhysicalCommonsense": "<150-200 words explanation here>",
"Reasoning_VideoPromptAlignment": "<150-200 words explanation here>",
"Reasoning_Coherency": "<150-200 words explanation here>",
"Decision": "<A_BETTER | B_BETTER | COMPARABLE>",
"WeightedScoreA": <float>,
"WeightedScoreB": <float>
}
```

Figure A14: **Prompt for the Video Reviewer** (subsequent clips in the first-frame mode).