

Supplementary Material: MagicFight: Personalized Martial Arts Combat Video Generation

Anonymous Authors

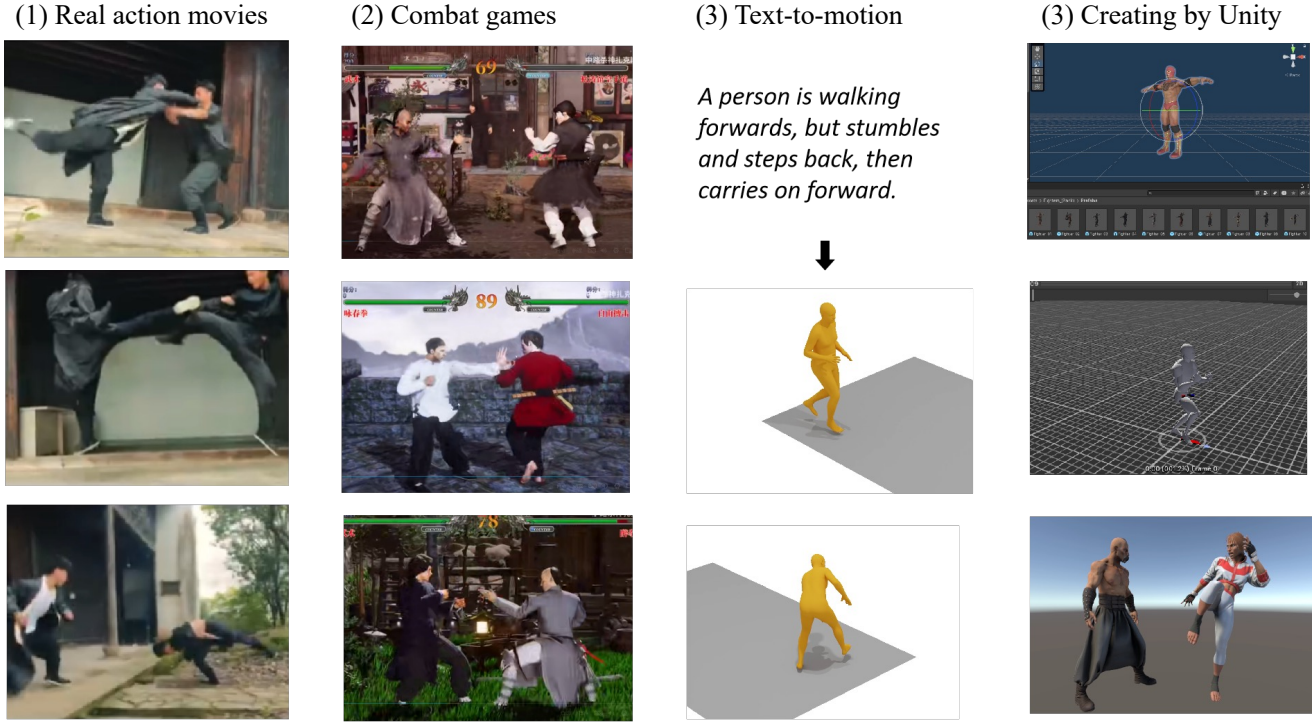


Figure 1: Comparison of dataset creation candidates. Real action videos in film and television have blurry picture quality and large camera shifts, and they are hard to extract pose maps. Synthesised combat videos in games have blurred picture quality, messy backgrounds, the presence of UI parts that obscure characters (blood bars, skills, etc.), and flat graphics. Text-to-motion generation (such as MotionGPT) did not address different character IDs modeling, so the motion has no texture. We chose to create a two-person martial arts video (column 4) by combining character, motion, and scene models using Unity.

1 DETAILS OF DATASET CREATION

1.1 Four Candidate Options

Here we detail our proposed 4 candidate options to create a combat dataset and explain their advantages and differences as shown in Fig. 1.

1) Use of a pure martial arts movie and television video dataset. The limited amount and diversity of video content available on the web may not fully meet the training needs. Besides, the problem of pose prediction accuracy in complex scenes affects the data quality.

2) Creating synthetic videos based on existing martial arts games can be limited by the content of the game itself (e.g., characters, actions, scenes, shots, etc.). The most serious problem is that almost all action games have UI parts that obscure the main screen, such as common blood bars, air bars, character names, skill move buttons, etc., which seriously affects the realism of the screen and model training.

3) Constructing datasets based on Text-to-Motion technology. We can create a motion dataset by utilizing Text-to-Motion technique, especially by using advanced models such as MotionGPT [4], which is a scheme to convert text descriptions directly into motion sequences. The structure of a motion sequence is defined as [frames, 22, 3], where "frames" represents the number of frames, "22" refers to the number of keypoints in the human body, and "3" represents the position of each keypoint in the 3D world coordinate system. The goal of this scheme is to use these generated action sequences as conditions for video generation, thus enabling precise control of the video content of a two-person fight scene. However, it is a technical challenge to further transform these motion sequences into RGB videos with a realistic feel. This process involves multiple complex steps such as motion rendering, character model building, environment setting, etc., and requires interdisciplinary expertise and technology. In addition, higher computational resources and specialized graphics processing software may be required to realize this process.

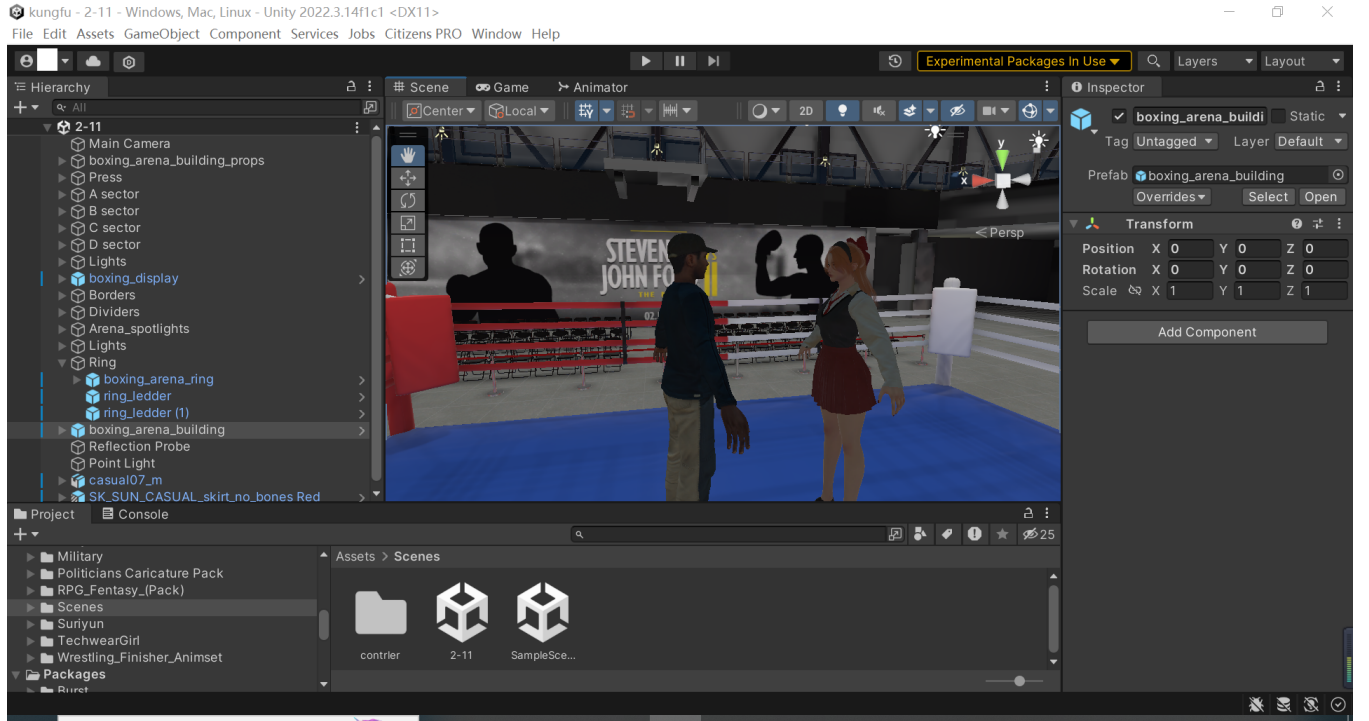


Figure 2: The interface of Unity, the dataset creation tool. A combination of character, action and scene models can be rendered to create a video of a character executing an action, which we use to simulate a live action martial arts video.

4) The last candidate is our Unity-based way, which is the final option to create our dataset in the main paper. We give some example images of the video in our KFF dataset in Fig. 3.

1.2 Our Unity-Based Way

After a detailed analysis of the different dataset creation options, we finally chose this option: using the physics modeling tool Unity and its action and character libraries to construct a two-person combat video and export it as a composite dataset as shown in Fig. 2. This decision was based on several key considerations, and the reasons for this choice are explained in detail below.

Technical feasibility and controllability: Unity, as a mature game development platform, provides a wealth of resources and tools that can help us efficiently build and render fight scenes with diverse perspectives and action representations. Compared with other solutions, Unity has a clear advantage in the controllability of actions and scenes, making the process of building data sets more flexible and precise. We were able to design specific fight moves, adjust character interactions, and even control lighting and environmental effects to create the ideal dataset for our training needs.

Diversity and Consistency: the datasets built through Unity are not only highly diverse in terms of viewpoints and actions, but also maintain a high degree of consistency in terms of visual style and action execution. This is crucial for training high-performance video generation models, as datasets with high consistency can reduce noise during model training and improve learning efficiency.

2 DETAIL ABOUT BACKGROUND CONTROLLING

As discussed in the main paper, our approach incorporates two distinct methods for background conditioning to accommodate user preferences in video generation. 1) Users have the option to directly provide a background image. This simple background can be incorporated seamlessly into the video through an end-to-end process, where the denoising U-Net is conditioned with the provided image, allowing precise background control throughout the video generation. This method ensures that the user's chosen background is accurately reflected in the generated video, enhancing the visual consistency and thematic relevance of the final output. 2) For users preferring text input, the initial conditioned background is set to pure white. This setup simplifies the initial generation process, where the denoising U-Net produces a video with a uniformly white background during the first stage. In the subsequent stage, the Background Crafter, developed specifically for more complex background generation, comes into play. Here, users can input a text prompt describing the desired background, which the Background Crafter then uses to generate detailed and contextually appropriate backdrops for the two-person combat scenes.

Recognizing the limitations of our MagicFight model in generating intricate backgrounds directly, we designed the Background Crafter to enhance user control over the background aesthetics. Based on the SDXL-Inpainting [3] with minor modifications, the Background Crafter utilizes three key elements: the original foreground image of the two-person fight, a background mask—readily



Figure 3: Examples video frame of our KFF dataset.

obtained from the initial white background but refined using a robust video matting method [1], and the user’s text prompt.

However, given that the original SDXL-Inpainting model [3] was not tailored for dual-person background tasks and fell short of our requirements, we finetuned it on a specialized dataset. This dataset included two-person foregrounds and complex scene backgrounds described by text. To address video temporal modeling, we integrated a temporal layer from AnimateDiff into the SDXL-Inpainting framework, transforming it into our Background Crafter. During sampling, we adopted a method similar to the one described in [2] to ensure inter-frame background consistency, employing shared latent space variables (z_T) across all frames and transmitting Key and Value information through the self-attention layers to maintain uniformity and continuity in the background across the video sequence. This integration not only enhances the model’s ability to handle dynamic backgrounds but also ensures that the

backgrounds are visually coherent throughout the video, aligning with the evolving narrative and actions within the scene.

3 MORE VIDEO RESULTS

In Figs. 4–8, we further showcase a variety of generated combat videos with IDs in an open-set situation. Our method excels in producing these results, demonstrating exceptional quality and creativity. The videos generated not only exhibit a high degree of diversity, capturing a range of dynamic interactions and combat styles, but also underscore the robustness of our approach under varying conditions and scenarios. This ability to consistently produce realistic and varied content highlights the advanced capabilities of our model, making it a significant contribution to the field of video generation.



Figure 4: More testing results of our MagicFight.



Figure 5: More testing results of our MagicFight.



Figure 6: More testing results of our MagicFight.



Figure 7: More testing results of our MagicFight.

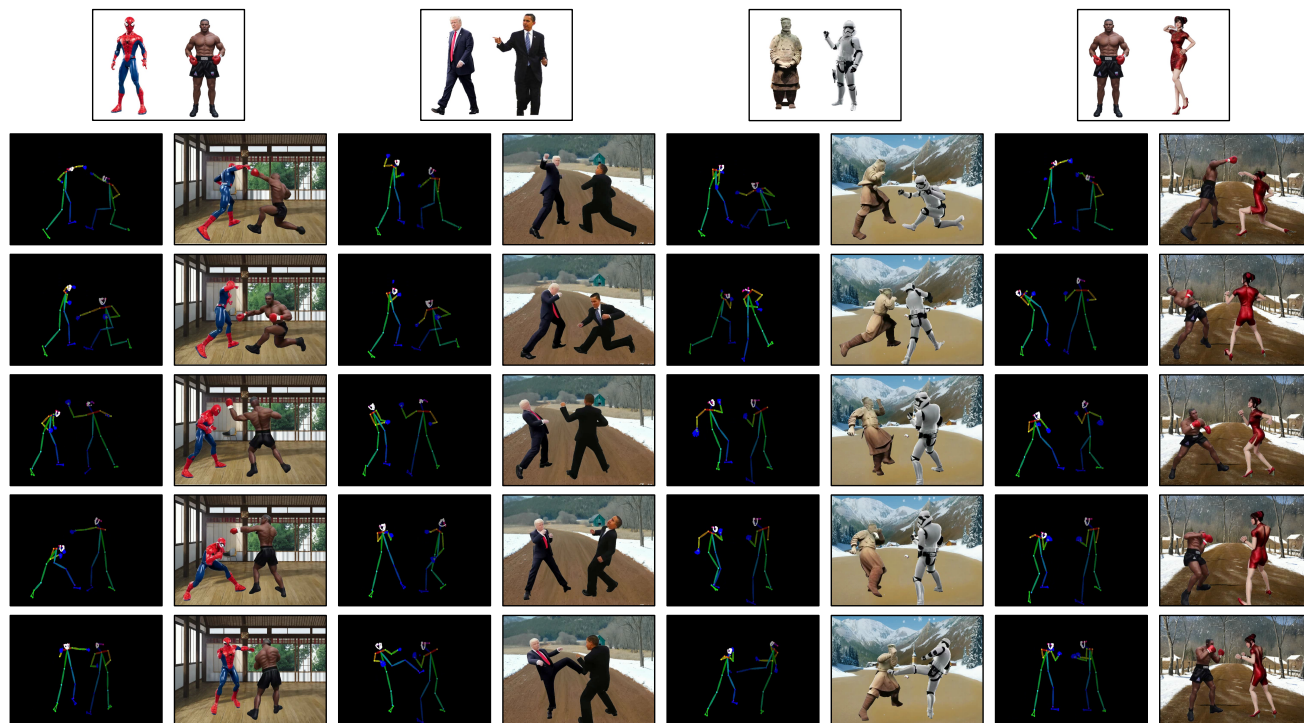


Figure 8: More testing results of our MagicFight.

REFERENCES

- [1] Shanchuan Lin, Linjie Yang, Imran Saleemi, and Soumyadip Sengupta. 2022. Robust high-resolution video matting with temporal guidance. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 238–247.
- [2] Jiayi Lv, Yi Huang, Mingfu Yan, Jiancheng Huang, Jianzhuang Liu, Yifan Liu, Yafei Wen, Xiaoxin Chen, and Shifeng Chen. 2024. GPT4Motion: Scripting Physical Motions in Text-to-Video Generation via Blender-Oriented GPT Planning. *CVPR* workshop (2024).
- [3] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. 2024. SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis. In *The Twelfth International Conference on Learning Representations*.
- [4] Yaqi Zhang, Di Huang, Bin Liu, Shixiang Tang, Yan Lu, Lu Chen, Lei Bai, Qi Chu, Nenghai Yu, and Wanli Ouyang. 2023. Motiongpt: Finetuned llms are general-purpose motion generators. *arXiv preprint arXiv:2306.10900* (2023).