REACTIVEGWM: FLEXIBLE CONTROL AND NPC REACTIVITY IN GAME WORLD MODELS

Anonymous authors
Paper under double-blind review

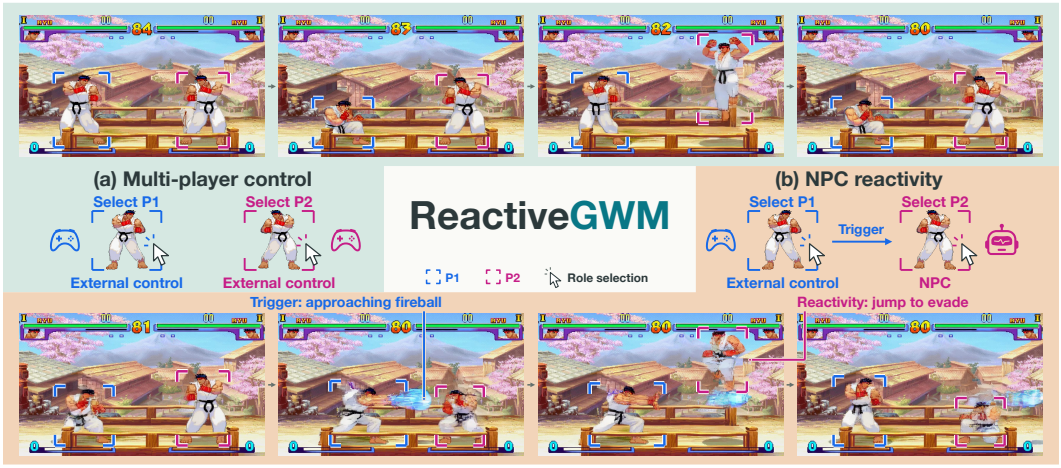


Figure 1: ReactiveGWM enables flexible role assignment and reactive game generation: (a) both characters follow external actions; (b) P1 is externally controlled, while P2 acts as an NPC and jumps to evade an approaching fireball.

ABSTRACT

Existing game world models typically adopt role-specific interactions, where player and NPC roles are bound to fixed characters. This limits their flexibility in multi-character games, where different characters may receive external control while NPCs must react to interactions triggered by players. This setting raises two key challenges: *how to flexibly assign control roles to individual characters, and how to support direct player control and reactive NPC behavior within a unified model*. These challenges are particularly pronounced in shared-view 2D games, where multiple, potentially visually identical characters share the same viewpoint, making camera cues insufficient to distinguish their roles. To address these challenges, we introduce **ReactiveGWM**, a reactive game world model that flexibly assigns control modes at initialization and jointly simulates externally controlled players and reactive NPCs. Specifically, ReactiveGWM introduces *Spatial Role Binding*, which grounds learned character handles to their corresponding regions in the initial frame using instance masks. Building on these handles, *Unified Agency Conditioning* unifies heterogeneous control signals across characters by encoding player actions and conditional NPC rules into character-specific control-token groups. Each group is then bound to its corresponding character handle, enabling the model to apply each control signal to its designated character. Meanwhile, causal self-attention restricts temporal context to the current and preceding latent frames when generating player actions and NPC responses. Experiments on two multi-character 2D games demonstrate that ReactiveGWM supports flexible character control across different player/NPC role assignments while jointly generating accurate player-controlled behaviors and reactive NPC responses, enabling more configurable and richer multi-character interactions. Anonymous project page is available at `https://reactivegwm.github.io/ReactiveGWM/`.

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.

1 INTRODUCTION

Recent advancements in world models (Ha & Schmidhuber, 2018; Parker-Holder et al., 2025) have established a new paradigm for simulating complex and interactive environments. By learning environment dynamics from large-scale offline gameplay videos, this paradigm naturally extends to game world models (Team et al., 2026; Skywork AI Matrix-Game Team, 2026; Hu et al., 2026), opening up new possibilities for interactive game generation.

To enable such interactive games, a key requirement is to model reactive interactions between players and Non-Player Characters (NPCs), where player inputs directly control selected characters, while player actions can automatically trigger corresponding NPC responses. In multi-character games, such interactions may involve different roles across characters. A character may be selected for external control in one rollout while serving as a reactive NPC in another. This flexible role assignment enables more diverse interaction patterns and greater gameplay flexibility.

However, most existing game world models (Hu et al., 2026; Zhu et al., 2026; Deng et al., 2026) commonly adopt a fixed interaction structure: external actions are bound to predefined player characters, while the remaining characters retain fixed NPC roles. We refer to this setting as *role-specific interaction*. It simplifies action grounding because the identity of the externally controlled character is known in advance. Hence, it restricts the range of interactions that a model can represent when different characters need to assume different control roles.

Moving beyond such fixed interaction structures introduces two closely related challenges. **First**, external actions must be grounded to the intended character rather than being tied to a predefined player/NPC role. **Second**, the model must support direct player control and reactive NPC behavior within a unified framework. These challenges are particularly pronounced in shared-view 2D games (Capcom, 1998), where multiple characters appear simultaneously under the same viewpoint, providing no camera-based cues for identifying their roles (Savva et al., 2026). Moreover, characters may have similar or even identical appearances, making visual identity alone insufficient to determine which character should receive external control and which should exhibit reactive behavior. Therefore, the model must explicitly associate external actions and reactive behaviors with individual characters while jointly modeling their dynamics within the same visual space.

To address these challenges, we introduce **ReactiveGWM**, a reactive game world model for multi-character 2D games. At rollout initialization, users assign each visible character either external control or an NPC behavior rule. The model then binds actions to the selected player characters and simulates the remaining NPCs according to their conditional rules. These role assignments remain fixed within a rollout, while the generated interaction evolves in response to player actions.

Specifically, ReactiveGWM comprises two complementary components. *Spatial Role Binding* associates each character with a learned handle and injects mask-weighted handles into the initial-frame video tokens. This establishes a correspondence between a character's spatial location and the controls assigned to it. *Unified Agency Conditioning* represents player actions and conditional NPC rules as control-token groups, binding each group to its character through the same handle. For NPCs, these rules encode trigger-response relationships that specify how they should react to particular player behaviors. Since NPC reactions are inherently conditioned on preceding player behaviors, we adopt causal self-attention to preserve this temporal dependency and prevent future video context from interfering with trigger interpretation. In our experiments, we evaluate ReactiveGWM on two multi-character 2D games with varying player/NPC role assignments and trigger-response interactions. The results demonstrate that ReactiveGWM enables flexible character control while jointly modeling reactive NPC behavior, highlighting its potential to support richer and more configurable interactions between players and NPCs.

In summary, our main contributions are as follows:

- We formulate multi-character reactive interaction beyond role-specific game world models, allowing different characters to assume externally controlled or reactive NPC roles.
- We introduce *Spatial Role Binding* and *Unified Agency Conditioning* to explicitly associate and model each character with its assigned roles within a unified video generation model.
- Experiments on two shared-view 2D games demonstrate flexible character control under different role assignments and effective player-triggered reactive NPC behaviors.

2 Related Works

2.1 Game World Models

Game world models (Parker-Holder et al., 2025; Yu et al., 2025; Skywork AI Matrix-Game Team, 2026; Tong et al., 2026) aim to construct interactive simulations of game environments by predicting future visual observations conditioned on player inputs. Early works such as GameNGen (Valevski et al., 2025) demonstrated that diffusion models can serve as real-time neural game engines for *DOOM*, while DIAMOND (Alonso et al., 2024) showed that the visual fidelity of diffusion-based world models can substantially affect downstream policy learning. Subsequent efforts, including Matrix-Game 2.0/3.0 (He et al., 2026; Skywork AI Matrix-Game Team, 2026), LingBot-World (Team et al., 2026; Gao et al., 2026), GameFactory (Yu et al., 2025), and Oasis (Decart et al., 2024), have further advanced game world modeling toward streaming, long-horizon, and open-domain generation.

Despite these advances, they largely focus on player-centric interaction, where external actions directly control a predefined player character to navigate or interact with the environment. However, interactive games involve not only player control but also reactive interactions between players and NPCs. Modeling such interactions requires the world model to capture how NPC behaviors respond to the actions of players, motivating recent efforts on interactive behavior modeling.

2.2 Interactive Behavior Modeling

Recent works have begun to model reactive interactions between players and NPCs. Incantation (Zhu et al., 2026), WanToFight (Hu et al., 2026), and WorldMind (Deng et al., 2026), for example, enable interactions among multiple characters by associating external actions and interactive behaviors with predefined character roles. These approaches substantially extend game world models beyond player-centric generation and enable richer character interactions.

However, their interaction structures are typically role-specific: control signals and reactive behaviors are tied to predefined characters or roles during training. Consequently, the same character cannot be readily reassigned between controlled and reactive roles across different rollouts. This limits their flexibility in multi-character games, where both the controlled character and the interaction may vary. Although ActionParty (Pondaven et al., 2026) enables external control to be bound to a selected character, it does not explicitly model reactive interactions with NPCs. In contrast, our work focuses on configurable multi-character interactions, where control roles can be flexibly assigned across characters and reactive NPC behaviors can be conditioned on players' interactions.

3 Method

ReactiveGWM (Figure 2) assigns flexible control modes at initialization and jointly simulates externally controlled players and reactive NPCs. We first formulate selective agency in multi-character games, then present *Spatial Role Binding* and *Unified Agency Conditioning*.

3.1 Problem Formulation

In a multi-character game, external actions and reactive behavior instructions must be associated with the characters that execute them. Our goal is to make them explicit: users select which characters receive external control and assign reactive behaviors to the remaining NPCs. This is particularly challenge in shared-view 2D games, where characters can share the same appearance and viewpoint cues cannot distinguish their control roles.

Given an initial observation X_0 containing N characters, let $\mathbf{M}_0 = (M_1, \ldots, M_N)$ denote their initial-frame masks. At rollout initialization, we assign each character a control mode $\rho_i \in \{\text{external}, \text{NPC}\}$, collected in $\boldsymbol{\rho}$. An externally controlled character receives an action sequence $\mathbf{a}_i = (a_{i,0}, \ldots, a_{i,T-1})$ over T control intervals. Each command uses the text format `ACTION=<action>`, e.g., `ACTION=jump` or `ACTION=move_left + jump`. An NPC receives a conditional rule r_i, with expressed as `If [condition], then [response]`, e.g., `If a player throws a shuriken, then jump to dodge it.` The indexed con-

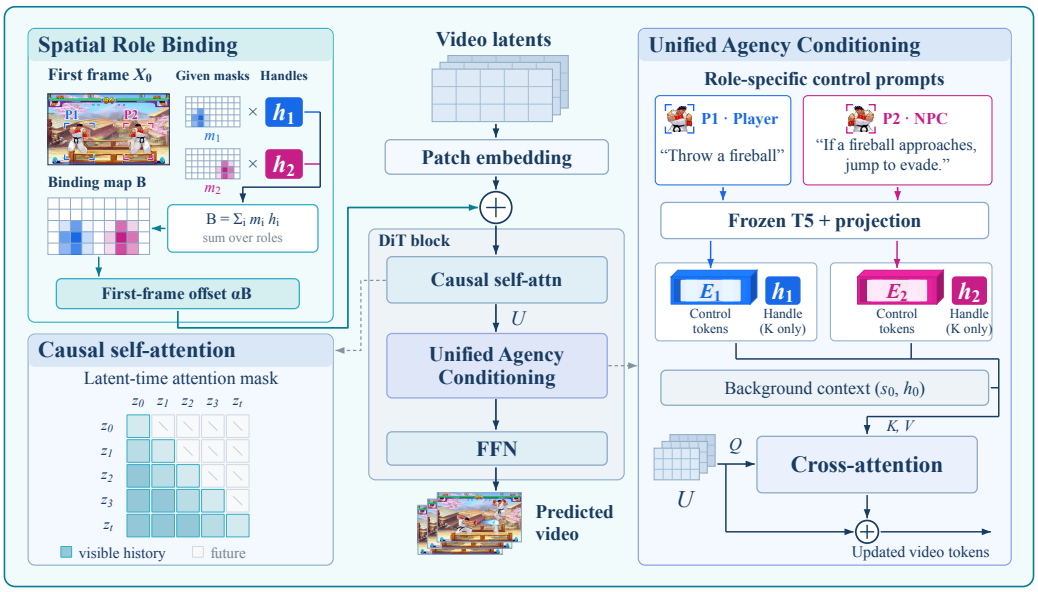


Figure 2: Overview of ReactiveGWM. Spatial Role Binding injects mask-weighted role handles into the first-frame tokens. Unified Agency Conditioning incorporates role-specific control tokens and background context through cross-attention, with actor handles injected into keys only. Causal self-attention limits temporal access to the current and past latent frames.

trol inputs are

$$\mathcal{A} = \{(i, \mathbf{a}_i) : \rho_i = \text{external}\}, \qquad \mathcal{R} = \{(i, r_i) : \rho_i = \text{NPC}\}. \tag{1}$$

The model jointly generates the future video according to

$$\mathbf{X}_{1:F} \sim p_\theta(\cdot \mid X_0, \mathbf{M}_0, \boldsymbol{\rho}, \mathcal{A}, \mathcal{R}), \tag{2}$$

where F is the number of output video frames. All characters evolve within the same generated scene, so their motions and interactions must remain jointly consistent.

For conditional NPC rule, it specifies a trigger condition and a response. The initial mask identifies the NPC that executes the response, while any externally controlled player may satisfy the trigger condition. For example, it may instruct an NPC to jump in response to a fireball in SF3 game. The initial mask identifies the character that should execute the response. The model must infer when the condition is met from the evolving video context and realize the response in the generated frames. Control modes and rules remain fixed within a rollout and can be reassigned when initializing another rollout.

This formulation combines external action control with conditional NPC rule, while allowing the control mode of each character to be specified independently of its appearance. Base on this formulation, we propose our method ReactiveGWM. For ReactiveGWM, Spatial Role Binding (Section 3.2) grounds character identities in the initial frame. Unified Agency Conditioning (Section 3.3) binds actions and rules to those identities and integrates them with causal video context.

3.2 Spatial Role Binding

To realize ReactiveGWM, the primary challenge is establishing reliable character binding. In a shared-view 2D game, visual appearance and camera perspective are insufficient to uniquely identify the targeted character—an ambiguity compounded when multiple entities share an identical appearance. We address this by utilizing initial-frame segmentation masks to map each character instance to a learned handle $h_i \in \mathbb{R}^d$, as shown in Figure 2. These handles are drawn from a shared codebook and normalized to unit root-mean-square magnitude. Each handle uniquely tracks an instance throughout a rollout and its index implies neither character class nor control modality (e.g., player vs. NPC). Finally, h_0 is reserved for the background.

Let $H_t(p) \in \mathbb{R}^d$ denote the patch embedding at spatial position p in latent frame t, with $t = 0$ corresponding to the initial frame. We resize each binary character mask to the patch grid by area averaging, obtaining soft coverage $\bar{m}_i(p) \in [0, 1]$. To prevent overlapping masks from increasing the total binding magnitude, we use

$$m_i(p) = \frac{\bar{m}_i(p)}{\max\Big(1, \sum_{j=1}^{N} \bar{m}_j(p)\Big)}. \tag{3}$$

The spatial binding map and the resulting patch embeddings are

$$B(p) = \sum_{i=1}^{N} m_i(p) h_i, \qquad \widetilde{H}_t(p) = H_t(p) + \mathbf{1}[t = 0]\, \alpha B(p), \tag{4}$$

where $\alpha = 0.25$ controls the injection strength. Thus, the same handle used to address a character in the control pathway is introduced at that character's initial spatial location. Patches outside the character masks receive zero offset. The binding is applied only to the initial frame, so generation requires no future character masks.

During training, we permute the assignment of characters to handle slots, jointly reordering their masks, control modes, action sequences, and NPC rules. This discourages fixed associations between handle indices, appearances, and control modes. The resulting interface allows a character to receive either external actions or an NPC rule while retaining the same spatial binding mechanism.

3.3 Unified Agency Conditioning

Building upon character binding, the subsequent challenge is unifying heterogeneous control signals into a single model while faithfully executing the designated behavior for each character. To this end, we propose Unified Agency Conditioning (UAC). UAC structures both external actions and reactive NPC rules as control-token groups addressed by character handles, and injects them into causal video features via a shared cross-attention mechanism.

Control tokens and cross-attention. For character i and control interval τ, let $c_{i,\tau}$ denote its control text. For an externally controlled character, $c_{i,\tau}$ is the action text for that interval. For an NPC, $c_{i,\tau} = r_i$ is a conditional rule describing its trigger and response, reused across control intervals. A frozen UMT5 (Chung et al., 2023) encoder $\mathcal{T}$ and a shared trainable projection $\mathcal{P}$ (Wan et al., 2025) produce

$$E_{i,\tau} = \mathcal{P}(\mathcal{T}(c_{i,\tau}))\,. \tag{5}$$

We preserve all valid text tokens and mask padding. The NPC rule is encoded jointly, allowing its condition and response to be interpreted together from video context.

We use the projected text sequence directly as the character's control group, $C_{i,\tau} = E_{i,\tau}$. The actor handle h_i binds this group to the character that executes the instruction through the cross-attention keys. We also include a background group $C_{0,\tau} = \{s_0\}$, where s_0 is a learnable background token shared across control intervals and paired with the background handle h_0.

For clarity, we suppress the block index. Let $U_t(p)$ denote the video features after self-attention and let $c_{i,\tau,\ell}$ be token ℓ in $C_{i,\tau}$. Queries are obtained from the normalized video features, while keys combine control content with the actor handle:

$$\begin{aligned} q_{t,p} &= \mathrm{RMSNorm}(W_Q\, \mathrm{LN}(U_t(p)))\,, \\ k_{i,\tau,\ell} &= \mathrm{RMSNorm}(W_K c_{i,\tau,\ell} + W_h h_i)\,, \\ v_{i,\tau,\ell} &= W_V c_{i,\tau,\ell}. \end{aligned} \tag{6}$$

The projection notation includes affine biases where present. RMS normalization precedes the split into attention heads; in the attention below, q, k, and v refer to the corresponding per-head slices. The actor handle is broadcast to all keys in its group, allowing queries to select controls by both their content and their bound character. It is injected only into keys and values are obtained from the projected control text.

Let $L_{i,\tau}$ denote the number of valid tokens in control group i. To account for different group lengths, we normalize the attention logits as

$$a_{t,p,i,\ell} = \underset{j,m}{\mathrm{softmax}}\left(\frac{q_{t,p}^{\top} k_{j,\tau,m}}{\sqrt{d_h}} - \log L_{j,\tau} + \beta_j\right)_{i,\ell}, \qquad \tau = t-1, \tag{7}$$

where d_h is the head dimension and the softmax ranges over all valid tokens in all groups, including the background. For each future latent frame $t \in \{1, \ldots, T\}$, we use control interval $\tau = t - 1$, since $t = 0$ is reserved for the initial frame. The length correction prevents a group from receiving a larger baseline attention mass solely because it contains more tokens. We set $\beta_i = 0$ for character groups and $\beta_0 = \log N$ for the background, balancing its baseline mass against the collection of N character groups. The background value is multiplied by a learned scale initialized to zero. The head outputs $\sum_{i,\ell} a_{t,p,i,\ell} v_{i,\tau,\ell}$ are concatenated, projected, and added to $U_t(p)$. The UAC residual is zero for the initial frame.

Causal video context. Reactive behavior requires access to the evolving scene. Before the cross-attention update, self-attention aggregates video context using a mask defined over latent-frame indices:

$$\mathcal{M}^{\mathrm{causal}}_{(t,p),(s,q)} = \begin{cases} 0, & s \le t, \\ -\infty, & s > t. \end{cases} \tag{8}$$

Consequently, every spatial token in frame t can attend to all positions in its current frame and preceding frames, including the initial spatial bindings. The temporal restriction applies between frames; spatial attention within a frame remains unrestricted. UAC uses the resulting features $U_t(p)$ to interpret an NPC rule in the context of the visible interaction. Control interval τ is aligned with future latent frame $t = \tau + 1$, and cross-attention at that frame reads only the corresponding control groups. Besides, we train the model with standard latent flow matching over future video latents, keeping the initial-frame latent clean. Together, these restrictions prevent future video tokens or later external controls from entering an earlier frame's conditioning pathway.

4 Experiments

4.1 Experimental Settings

Datasets. We utilize two 2D-game environments: Hikou no Mizu (HNM) (Hikou no mizu Project, 2024) and Street Fighter III: New Generation (SF3) (Capcom, 1998). Both protocols use five-second clips containing 101 frames at 20 FPS and 832×480 resolution. For HNM, it comprising 20,000 training clips with 2–6 characters and up to two NPCs. It combines direct action control with nine trigger–response rules and a no-op category. For SF3, it covers two-fighter interactions across eleven playable characters, targeting 20,000 training clips with NPC interactions covering nine reaction rules and a no-op category. The details are illustrated in Appendix A.

Models. We initialize ReactiveGWM from the pretrained Wan2.2-TI2V-5B model (Wan et al., 2025) and train separate models for HNM and SF3. The diffusion transformer and the proposed role-binding and conditioning modules are jointly fine-tuned, while the UMT5-XXL text encoder and Wan2.2 VAE remain frozen. We use AdamW with a learning rate of 5×10^{-5}, weight decay 0.01, and $(\beta_1, \beta_2) = (0.9, 0.999)$. Training uses BF16 precision and an effective batch size of 8 for both HNM and SF3, with each trained for 20K steps. We optimize the weighted flow-matching MSE over the 25 future latent frames, keeping the initial-frame latent clean and excluding it from the loss.

Evaluation Metrics. Action accuracy (AA) measures correct action execution by the instructed player without unrelated active behavior, and selection accuracy (SA) measures whether the selected player jumps while others remain idle. Joint reactivity (JR) measures reference-aligned waiting and response onset: the NPC must remain inactive before a fixed trigger boundary from the paired GT trajectory and initiate the prescribed response afterward. Note that the no-op cases require inactivity throughout the clip. These three metrics are case-level pass rates. Visual quality (VQ) equally weights clarity, character identity and structure, scene integrity, and temporal stability. Using the evaluation data collected for JR, VQ assesses the visual fidelity of generated player–NPC interactions. All scores use a 0–100 scale. We report mean $\pm$ sample standard deviation over three rounds

separately for GPT-6-astra (OpenAI, 2026) and Gemini 3.5 Flash (Team et al., 2025). The details are illustrated in Appendix B.

4.2 Main Results

Table 1: Comparison of model variants under Gemini 3.5 Flash (Gemini) and GPT-6-astra (GPT) evaluation for AA, SA, JR, and VQ. GT serves as an empirical upper-bound reference.

	AA ↑		SA ↑		JR ↑		VQ ↑	
	Gemini	GPT	Gemini	GPT	Gemini	GPT	Gemini	GPT
HNM-GT	97.53±2.14	100.00±0.00	97.78±0.96	100.00±0.00	98.89±1.92	100.00±0.00	98.69±0.18	100.00±0.00
ReactiveGWM	88.89±0.00	96.30±0.00	95.00±4.41	100.00±0.00	76.67±0.00	76.67±0.00	79.31±0.37	78.80±0.53
SF3-GT	100.00±0.00	96.97±0.00	100.00±0.00	100.00±0.00	96.67±3.33	100.00±0.00	99.65±0.12	99.79±0.00
ReactiveGWM	98.99±1.75	96.97±0.00	100.00±0.00	100.00±0.00	91.11±1.92	91.11±3.85	71.85±1.27	82.49±0.52

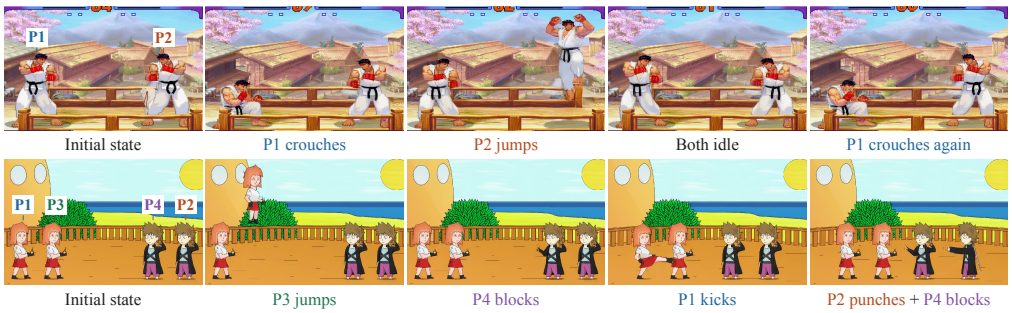


Figure 3: Multi-player interaction generated by ReactiveGWM in SF3 (top) and HNM (bottom). Colors match players to their actions.



Figure 4: NPC reactivity in SF3 (a,b) and HNM (c,d). Each generated sequence shows the initial roles, a player-triggered event, and the NPC response.

Character selection and action execution. This paragraph evaluates the performance of ReactiveGWM in character selection and player action injection. Table 1 compares ReactiveGWM

with GT under both judges. On SF3, AA reaches 98.99/96.97 (Gemini/GPT), close to the GT scores of 100.00/96.97, while HNM AA is 88.89/96.30 compared with 97.53/100.00 for GT. ReactiveGWM achieves SA of 95.00/100.00 on HNM and 100.00/100.00 on the SF3 selection cases. These results demonstrate that ReactiveGWM reliably selects the intended character in both environments while effectively incorporating player actions into the model.

Figure 3 complements these results by illustrating independent and simultaneous character control. In SF3, two identically dressed fighters separately execute P1's crouch and P2's jump, then return to idle before P1 crouches again. In HNM, four characters, including pairs with the same appearance, execute different action sequences. The final frame shows P2 punching while P4 blocks under simultaneous inputs. These examples illustrate both separate control of visually similar characters and concurrent execution of different actions.

NPC reactivity and visual quality. This paragraph evaluates ReactiveGWM in terms of NPC reactivity and visual quality in reactive game generation. ReactiveGWM achieves JR scores of 76.67/76.67 on HNM and 91.11/91.11 on SF3 (Gemini/GPT). Under this reference-aligned protocol, both judges assign higher JR scores to SF3 than to HNM, although scores remain below the GT references of 98.89/100.00 on HNM and 96.67/100.00 on SF3.

Figure 4 illustrates the requested trigger–response reactivity in selected generated sequences. The SF3 examples show a projectile parry and a throw triggered when the player enters close range. In HNM, one example depicts an NPC following a departing player, while another shows an NPC advancing and kicking after a third character strikes the player. The latter demonstrates that NPC responses can be conditioned on interactions between other characters, extending beyond direct player–NPC exchanges. More reactivity results are provided in Appendix D, while additional results on multi-player interactions are presented in Appendix C. VQ scores are 79.31/78.80 on HNM and 71.85/82.49 on SF3, below the corresponding GT scores of 98.69/100.00 and 99.65/99.79. Thus, despite strong action control and reactive behavior, visual fidelity remains an area for improvement.

Table 2: Component ablations of ReactiveGWM on HNM game.

	AA ↑		SA ↑		JR ↑		VQ ↑	
	Gemini	GPT	Gemini	GPT	Gemini	GPT	Gemini	GPT
HNM-GT	97.53±2.14	100.00±0.00	97.78±0.96	100.00±0.00	98.89±1.92	100.00±0.00	98.69±0.18	100.00±0.00
ReactiveGWM	88.89±0.00	96.30±0.00	95.00±4.41	100.00±0.00	76.67±0.00	76.67±0.00	79.31±0.37	78.80±0.53
w/o Spatial Role Binding	66.67±0.00	54.32±2.14	15.00±0.00	15.00±0.00	26.67±0.00	23.33±0.00	79.39±0.83	78.24±0.18
w/o NPC Conditioning	88.89±0.00	87.65±2.14	95.56±0.96	98.33±0.00	17.78±5.09	13.33±0.00	60.30±0.65	63.12±0.18

4.3 Effectiveness of Spatial Role Binding

In this section, We investigate whether *Spatial Role Binding* helps associate video features with the correct character's controls and how it establishes this correspondence.

We first compare the full model with the no-binding variant on HNM using AA, SA, JR, and VQ. Table 2 shows that removing Spatial Role Binding reduces AA from 88.89/96.30 to 66.67/54.32, SA from 95.00/100.00 to 15.00/15.00, and JR from 76.67/76.67 to 26.67/23.33 (Gemini/GPT). In contrast, VQ remains similar, at 79.31/78.80 for the full model and 79.39/78.24 for the no-binding variant. These results suggest that Spatial Role Binding primarily benefits character-specific control and NPC reactivity, with little change in overall visual quality.

To investigate the mechanism underlying these gains, we analyze layer-wise cross-attention on 200 HNM validation videos with noised GT latents at noise fractions of 0.1, 0.5, and 0.9. At each layer, we sum attention probabilities over each character's control tokens and average across heads. Foreground actor-selection accuracy measures whether the highest-scoring character group matches the query's character, with the background group excluded from Top-1 selection.

We compare ReactiveGWM, the no-binding variant, and ReactiveGWM with first-frame handle injection disabled. Figure 5 shows a prominent layer-9 peak with binding, most pronounced at high noise. Disabling first-frame injection removes this peak and brings accuracy close to uniform actor

choice. Together, these findings support Spatial Role Binding as a spatial cue that links video queries to the appropriate character's control group.

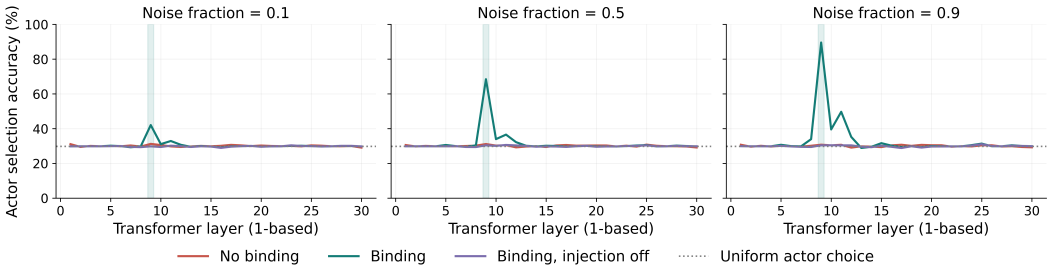


Figure 5: Layer-wise foreground actor-selection accuracy on 200 HNM validation videos with noised GT latents. Attention probabilities are summed by actor and averaged across heads; Top-1 excludes the background group. The dotted line denotes uniform actor choice.

4.4 Analysis on NPC Conditioning

We investigate whether NPC conditioning supports rule-guided reactions and how NPC video features attend to the condition and response clauses during generation. We first compare the full model with the variant without NPC conditioning on HNM. Table 2 shows that removing NPC conditioning reduces JR from 76.67/76.67 to 17.78/13.33 and VQ from 79.31/78.80 to 60.30/63.12 (Gemini/GPT). AA changes from 88.89/96.30 to 88.89/87.65, while SA remains high at 95.56/98.33 compared with 95.00/100.00 for the full model. These results suggest that NPC conditioning is particularly important for trigger–response behavior and also benefits the visual quality of generated player–NPC interactions.

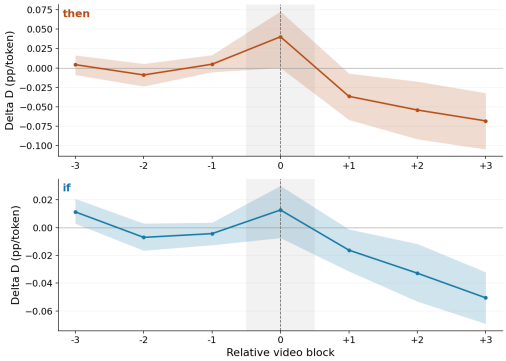


Figure 6: Layer-9 attention to NPC response (*then*, top) and condition (*if*, bottom) clauses. Curves show changes in attention per token relative to the three-block pre-trigger baseline, averaged with equal rule weights. Shading denotes 95% initial-frame-group cluster-bootstrap intervals. Block 0 contains the source GT trigger.

To examine attention to the these conditional text, we conduct a separate diagnostic analysis on HNM inputs. We analyze layer 9, selected from the preceding role-binding study. During free generation, we measure attention from generated NPC body regions to their own *if* and *then* clauses, averaging over 24 heads and 30 denoising steps. Attention is normalized over all valid conditioning keys. For each clause, we sum its token probabilities, divide by its token count, and subtract the mean over three complete pre-trigger blocks. We align these changes to the source GT trigger block (0.2 seconds); GT timestamps and future NPC actions are not supplied to the model.

Figure 6 shows the test results, with rules weighted equally. At the GT block, 58 measurable events (trigger-response events) yield a mean *then* increase of 0.04017 percentage points per token. The lower bound is close to zero, and the increase does not persist over subsequent blocks. The *if* change is 0.01271, providing no evidence of a consistent decrease. Together, these findings show the effectiveness of conditional NPC rule for reactive NPCs generation.

5 Conclusions

We presented ReactiveGWM, a reactive game world model for flexible role control and NPC reactivity. *Spatial Role Binding* grounds character handles in the initial frame, while *Unified Agency Conditioning* binds player actions and conditional NPC rules to corresponding characters. Results on HNM and SF3 demonstrate flexible role assignment and player-triggered NPC responses, supporting character-level conditioning for configurable multi-character interaction.

AI USE STATEMENT

In this work, we used GPT-6-astra (OpenAI, 2026) and Gemini 3.5 Flash (Team et al., 2025) as automated evaluators for action accuracy, selection accuracy, joint reactivity, and visual quality, following the protocols described in Section 4.1 and Appendix B. Additionally, we used generative AI tools to modify scientific figures, suggest experimental parameters, edit portions of the manuscript, search for information.

We have reviewed all AI-assisted work. We manually verified the literature identified through AI-assisted searches and carefully reviewed AI-assisted text, figures, and experimental suggestions before incorporating them into the manuscript. The automated evaluation procedures and score aggregation are documented in Appendix B. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in atari. *Advances in Neural Information Processing Systems*, 37:58757–58791, 2024.

Capcom. Street fighter Alpha 3. Video game, 1998. Arcade.

Hyung Won Chung, Xavier Garcia, Adam Roberts, Yi Tay, Orhan Firat, Sharan Narang, and Noah Constant. Unimax: Fairer and more effective language sampling for large-scale multilingual pretraining. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=kXwdL1cWOAi.

Etched Decart, Quinn McIntyre, Spruce Campbell, Xinlei Chen, and Robert Wachen. Oasis: A universe in a transformer. *URL: https://oasis-model. github. io*, 2(3):6, 2024.

Zhiyang Deng, Boran Zhang, Danze Chen, and Yeying Jin. Worldmind: Decoupled game world model for state-aware npc behavior. *arXiv preprint arXiv:2608.21439*, 2026.

Zelin Gao, Qiuyu Wang, Jiapeng Zhu, Jingye Chen, Zichen Liu, Qingyan Bai, Jiahao Wang, Yufeng Yuan, Hanlin Wang, Yichong Lu, Ka Leong Cheng, Haojie Zhang, Jian Gao, Tianrui Feng, Yuzheng Liu, Yao Yao, Yinghao Xu, Xing Zhu, Yujun Shen, and Hao Ouyang. Infinite worlds with versatile interactions. *arXiv preprint arXiv:2607.07534*, 2026.

David Ha and Jürgen Schmidhuber. World Models. *arXiv e-prints*, art. arXiv:1803.10122, March 2018. doi: 10.48550/arXiv.1803.10122.

Xianglong He, Chunli Peng, Zexiang Liu, Boyang Wang, Yifan Zhang, Qi Cui, Fei Kang, Biao Jiang, Mengyin An, Yangyang Ren, Baixin Xu, Hao-Xiang Guo, Kaixiong Gong, Size Wu, Wei Li, Xuchen Song, Yang Liu, Yangguang Li, and Yahui Zhou. Matrix-game 2.0: An open-source real-time and streaming interactive world model, 2026. URL https://arxiv.org/abs/2508.13009.

Hikou no mizu Project. Hikou no mizu: An open-source platform fighting game. https://hikounomizu.org/, 2024. Version 1.1, accessed September 2026.

Li Hu, Guangyuan Wang, Peng Zhang, and Bang Zhang. Wantofight: Real-time generative game engine for multi-player combat interaction, 2026. URL https://arxiv.org/abs/2607.12592.

OpenAI. Gpt-6 astra: A new generation of intelligence. https://openai.com/index/gpt-6-astra/, 2026. Accessed: 2026-09-25.

Jack Parker-Holder, Shlomi Fruchter, et al. Genie 3: A new frontier for world models. *Google DeepMind Blog*, 4, 2025.

Alexander Pondaven, Ziyi Wu, Igor Gilitschenski, Philip Torr, Sergey Tulyakov, Fabio Pizzati, and Aliaksandr Siarohin. Actionparty: Multi-subject action binding in generative video games. In *European Conference on Computer Vision*, pp. 74–92. Springer, 2026.

Georgy Savva, Oscar Michel, Daohan Lu, Suppakit Waiwitlikhit, Timothy Meehan, Dhairya Mishra, Srivats Poddar, Jack Lu, and Saining Xie. Solaris: Building a multiplayer video world model in minecraft, 2026. URL https://arxiv.org/abs/2602.22208.

Skywork AI Matrix-Game Team. Matrix-game 3.0: Real-time and streaming interactive world model with long-horizon memory. Technical report, 2026. URL https://github.com/SkyworkAI/Matrix-Game/blob/main/Matrix-Game-3/assets/pdf/report.pdf.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, and et al. Gemini: A family of highly capable multimodal models, 2025. URL https://arxiv.org/abs/2312.11805.

Robbyant Team, Zelin Gao, Qiuyu Wang, Yanhong Zeng, Jiapeng Zhu, Ka Leong Cheng, Yixuan Li, Hanlin Wang, Yinghao Xu, Shuailei Ma, Yihang Chen, Jie Liu, Yansong Cheng, Yao Yao, Jiayi Zhu, Yihao Meng, Kecheng Zheng, Qingyan Bai, Jingye Chen, Zehong Shen, Yue Yu, Xing Zhu, Yujun Shen, and Hao Ouyang. Advancing open-source world models, 2026. URL https://arxiv.org/abs/2601.20540.

Zizhao Tong, Hongfeng Lai, Zeqing Wang, Zhaohu Xing, Kexu Cheng, Haoran Xu, Zhao Pu, Shangwen Zhu, Ruili Feng, Jian Zhao, Yan Zhang, Hao Tang, Yeying Jin, and Ling Shao. Scope: Simulating cross-game operations in playable environments for fps world models, 2026. URL https://arxiv.org/abs/2605.23345.

Dani Valevski, Yaniv Leviathan, Moab Arar, and Shlomi Fruchter. Diffusion models are real-time game engines. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 73754–73776, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/b71ecea210f7159f31e46631fe5c838f-Paper-Conference.pdf.

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.

Jiwen Yu, Yiran Qin, Xintao Wang, Pengfei Wan, Di Zhang, and Xihui Liu. Gamefactory: Creating new games with generative interactive videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11590–11599, October 2025.

Shangwen Zhu, Qianyu Peng, Zhao Pu, Zhilei Shu, Xiangrui Ke, Zhaohu Xing, Zizhao Tong, Zeqing Wang, Xinyu Cui, Zian Zheng, Huangji Wang, Jian Zhao, Yeying Jin, Fan Cheng, and Ruili Feng. Incantation: Natural language as the action interface for multi-entity video world models, 2026. URL https://arxiv.org/abs/2605.18601.

A Details of Datasets

A.1 Shared Data Format

Both datasets contain five-second game trajectories, represented by an initial observation and 100 future RGB frames at 20 FPS and 832×480 resolution. Each trajectory has 25 external-control intervals of 0.2 seconds, with each interval aligned to four future video frames. We store the initial RGB image, one initial-frame mask per character instance, control-mode assignments, external action sequences, conditional NPC rule specifications, and the target video. Character masks are obtained from native instance-ID rendering and remain associated with physical character instances even when appearances are identical. Control assignments remain fixed within a trajectory.

The model is conditioned on the initial image and masks, external actions, and static NPC rule. Realized NPC inputs, native game events, and trigger/response timestamps are retained for collection verification and are not supplied as future conditioning. Recorded target identities likewise do not introduce a separate target-slot input to ReactiveGWM. Table 3 summarizes the dataset splits.

Table 3: Training, validation, and test set sizes in video clips.

Dataset	Train	Validation	Test
HNM	20,000	200	200
SF3	20,000	200	200

A.2 Hikou no Mizu

Scene composition and actions. HNM trajectories contain two to six characters, with zero, one, or two NPCs. The training set contains 20,000 clips covering external-only control and NPC interactions; each validation and test split contains 100 clips of each type. We vary character appearances, initial layouts, action sequences, and role assignments. The external action vocabulary has 21 commands: no-op; jump, punch, kick, and shuriken throw; crouch alone or combined with punch, kick, or shuriken throw; block alone or combined with crouch; and left/right movement alone or combined with jump, punch, kick, or shuriken throw.

NPC Reactive behaviors. Table 4 lists nine trigger–response rule families (H01–H09) and one passive no-op condition (H10). The missed-attack and jump-in rules each have separate punch and kick policies, yielding 11 active rule/policy combinations. The no-op condition releases all controls throughout the clip. NPC instructions express the trigger and response in conditional language, for example, *"If the target player throws a shuriken, then jump to dodge it."* Native controllers verify the triggering event, the corresponding response, and recovery before a subsequent response. The collection includes repeated reactions, passive NPCs, and cases in which the specified trigger does not occur.

Training-set filtering. The training set includes external-only clips, passive-NPC clips, and clips with valid active reactions. For a training clip containing active NPCs, every active NPC must have a valid trigger and complete its observed trigger–response–recovery cycles, with controls released outside the response. If any active NPC has no valid trigger, the entire clip is excluded. Retained clips preserve their full waiting and recovery segments.

A.3 Street Fighter III: New Generation

Scene composition and roles. SF3 trajectories contain two fighters drawn from eleven playable characters, with either two externally controlled players or one player and one NPC. The released training set contains 10,000 clips of each type; validation and test each contain 100 clips of each type. Within the NPC subset, each of the nine reactive rules and the no-op category contributes 1,000 training clips and ten clips to each held-out split. For each rule and split, the NPC occupies each physical player slot equally often. This assignment follows the physical controller port and character instance throughout the clip, including when the fighters cross on screen.

Table 4: HNM NPC rule families and the no-op condition. The target is the external player designated by the collection controller. These descriptions summarize the rule semantics. Although the original prompt templates use the phrase "the target player," this phrase is not accompanied by a player identifier or a target-selection input.

Rule	Trigger	Response
H01	Target moves away.	Approach the target until close.
H02	Target approaches and enters close range.	Retreat until at a safe distance.
H03	Target's punch or kick misses.	Approach and punch or kick during the target's recovery.
H04	Target throws a shuriken.	Jump to dodge the projectile.
H05	Another external player hits the target.	Approach and kick as the target recovers.
H06	Target hits the NPC.	Recover, then approach and kick once.
H07	Target punches or kicks the NPC.	Block standing for high attacks and crouching for low attacks.
H08	Target jumps toward the NPC.	Punch or kick the target while airborne.
H09	Target's punch or kick contacts another external player.	Approach the target and kick once.
H10	None (no-op).	Keep all controls released throughout the clip.

Actions and temporal alignment. The player action vocabulary contains 20 commands: no-op, left/right movement, jump, crouch, three strengths each of punch and kick, left/right movement combined with jump, left/right movement with a light punch, crouching heavy punch, crouching light/heavy kick, and left/right crouching inputs. NPC collection uses a separate 18-command vocabulary covering movement, jump, guards, parries, attacks, and throws; these realized NPC commands are used for replay and auditing rather than model conditioning.

Reactive behaviors. Table 5 summarizes the nine conditional NPC rules (R01–R09) and the no-op category (R10). The collection includes valid zero-trigger cases, single responses, and repeated responses where supported by the rule and character configuration. Zero-trigger cases retain the same static rule instruction and require the NPC to release all controls; a missed response to a valid trigger is a failed sample, not a zero-trigger example. The passive no-op category instead specifies released controls throughout the clip. SF3 retains its valid zero-trigger cases and does not use the HNM training-set filter.

Table 5: SF3 NPC rule families and the no-op category. The opponent is the externally controlled fighter. Native game events are used to verify the corresponding response.

Rule	Trigger	Response
R01	Opponent moves away.	Approach until close.
R02	Opponent enters close range.	Retreat to a safe distance.
R03	Opponent's attack misses.	Punish the same attack during recovery.
R04	Opponent's projectile threatens the NPC.	Jump to evade that projectile.
R05	Opponent hits the NPC.	Recover before counterattacking.
R06	Opponent attacks the NPC.	Guard according to the attack type.
R07	Opponent jumps toward the NPC.	Anti-air the opponent while airborne.
R08	Opponent's projectile approaches.	Parry the projectile.
R09	Opponent enters throw range.	Execute a normal throw.
R10	None (no-op).	Keep all controls released throughout the clip.

Quality control. The initial RGB image matches video frame zero, and each mask contains the visible primary character body, excluding projectiles, shadows, effects, and interface elements. We verify that recorded action IDs reconstruct the complete native input sequence and that native events

confirm trigger and response outcomes. Duplicate checks cover RGB content, character-ID renderings, full inputs, and character trajectories, including mirrored or role-swapped variants.

B Details of Evaluation

Following Section 3, $F = 100$ denotes the number of future RGB frames, $T = 25$ the number of control intervals, and $\tau \in \{0, \dots, T-1\}$ a control-interval index. We use $f \in \{0, \dots, F\}$ for RGB frames, with $f = 0$ corresponding to X_0, and retain t for latent frames. Evaluation cases, judges, and rounds are indexed by n, g, and $e \in \{1, 2, 3\}$, respectively. Scores are computed separately for each game and model variant.

For AA, SA, and JR, the judge returns `pass`, `fail`, or `unknown`, together with frame-indexed observations and an explanation. Both `fail` and `unknown` remain in the denominator and receive zero credit. Let K_μ be the number of cases for metric $\mu \in \{\mathrm{AA}, \mathrm{SA}, \mathrm{JR}\}$, and let $y^{\mu}_{n,g,e}$ denote the case-level judgment. The round score is

$$S^{\mu}_{g,e} = \frac{100}{K_\mu} \sum_{n=1}^{K_\mu} \mathbf{1}\left[y^{\mu}_{n,g,e} = \texttt{pass}\right]. \tag{9}$$

We first aggregate cases within each round, then report the mean and sample standard deviation across the three round scores, separately for each judge. The same procedure is used for the round-level VQ scores defined below.

B.1 Action Accuracy (AA)

Objective and data. AA measures whether an action instruction is visibly executed by its designated character without unrelated active behavior. HNM tests nine basic actions: left/right movement, jump, punch, kick, shuriken throw, block, crouch, and no-op. Each action has three independently constructed scene variants, giving 27 cases. Scenes contain two visually distinct characters in an open layout: one receives the tested action, while the other receives no-op. Brief actions are requested once; sustained actions occupy a short control interval surrounded by no-op intervals. SF3 tests 11 actions—no-op, left/right movement, jump, crouch, and light/medium/heavy punches and kicks—in three scene variants with Ryu, Yang, or Ken as the tested fighter, giving 33 cases.

Evidence and decision. Each judge receives 25 full-clip frames at $f_u = \mathrm{round}(Fu/24)$ for sample index $u \in \{0, \dots, 24\}$, together with a dense action window; duplicate frame indices are removed. For HNM, an action scheduled at control interval τ_{act} uses every frame from $4\tau_{\mathrm{act}} - 2$ through $4\tau_{\mathrm{act}} + 18$, clipped to $[0, F]$. The factor of four is the number of future RGB frames per control interval. SF3 AA uses the frozen interval 16–44 (0.8–2.2 seconds), around the scheduled 1.0-second action start. No-op receives the same sampling procedure. Each case receives one direct judgment per round. The dense window captures brief actions, while full-clip frames expose additional activity before or after the requested action.

Passing requires the designated instance to perform the requested action with no unrelated new active action in the supplied evidence. Movement requires its own displacement in the requested image direction; jumping requires takeoff; punches and kicks require the specified strike but not a hit. HNM throwing also requires a visible shuriken, while block and crouch require the corresponding posture. For SF3, an unresolved punch/kick strength is `unknown`, not credited as a generic strike. Ordinary idle animation, passive recoil, natural physics, and action recovery are allowed. The instructed action or its name is not treated as evidence that it occurred.

B.2 Selection Accuracy (SA)

SA isolates instance selection by using the same requested action, a single jump, for every tested role. HNM has three independent scenes for each population size from two to six. Within each scene, every character is selected once, yielding $3(2 + 3 + 4 + 5 + 6) = 60$ cases. Paired selections share the initial scene, identity assignments, timing, and generation seed, changing only the selected character. This includes visually identical characters as separate instances. SF3 uses three two-fighter scenes, including an identical-character pair; selecting each fighter once yields six cases.

Both games use the 25 full-clip frames and the input-timed dense window $[4\tau_{\mathrm{act}} - 2, 4\tau_{\mathrm{act}} + 18]$, clipped to $[0, F]$, where τ_{act} is the scheduled jump interval. A case passes only if the selected physical instance jumps once and every other character performs no new active action. Selecting the wrong jumper, multiple jumpers, no jumper, or another active action fails. Natural idle motion and completion of a pre-existing animation are allowed. SA is averaged over all cases, so the HNM aggregate weights population sizes by their case counts rather than equally weighting the five population sizes. The six-case SF3 suite provides a limited selection diagnostic.

B.3 Joint Reactivity (JR)

Objective and cases. The reported JR tests whether the designated NPC waits before a fixed GT trigger boundary and initiates the prescribed response afterward. Each game uses 30 cases: three GT-trigger cases for each of the nine active reaction categories, plus three passive no-op cases. Active categories follow H01–H09 for HNM and R01–R09 for SF3; the no-op categories are H10 and R10, respectively. For no-op cases, waiting is required on both sides of a fixed random split. The category semantics follow Tables 4 and 5. Table 6 states the post-boundary action criteria used for evaluation.

Table 6: Response actions used by the visual JR evaluator. An eligible action must begin in a scored post-boundary window. No-op instead requires continued waiting.

Rules	Accepted response
HNM H01 / H02	NPC's own movement toward / away from any external player
HNM H03, H08	Punch
HNM H04	Jump
HNM H05, H06, H09	Kick
HNM H07	Standing or crouching block
SF3 R01 / R02	NPC's own movement toward / away from the other fighter
SF3 R03, R05, R07	Attack, without requiring a particular strike strength
SF3 R04 / R06	Jump / standing or crouching block
SF3 R08 / R09	Parry / normal throw
HNM H10, SF3 R10	Wait throughout the scored segment

Fixed temporal alignment. For each active case, let f_n^{bdry} denote the RGB frame corresponding to the trigger timestamp in its native GT event log. This boundary is fixed for both GT and generated videos and is not adjusted to events in the generated clip. Frames $1, \ldots, f_n^{\mathrm{bdry}} - 1$ form the waiting segment, and frames $f_n^{\mathrm{bdry}}, \ldots, F$ form the response segment. No-op cases use a fixed random split. GT event times and realized NPC actions are not provided to the model.

Window judgments and aggregation. JR covers all 100 future frames. Each segment is partitioned into consecutive windows of at most 44 scored frames, preceded by up to five context frames, plus the identity reference. Context helps identify ongoing actions and new onsets but is not scored again. A WAIT window passes only when the NPC shows no deliberate activity. New movement, jumps, attacks, guard, crouch, parry, or throws violate waiting; ordinary idle animation, passive recoil, and natural physics do not. An ongoing deliberate action still violates waiting even if it began in context.

A RESPOND window passes only if the required action newly starts in its scored frames at or after f_n^{bdry}. An action already underway in context cannot be counted again; a later separate onset can count. The response need not land a hit, successfully dodge or intercept, satisfy an exact recovery time, or end at a precise location. Once a valid onset is identified, later extra actions or repetitions do not erase that post-segment success. HNM approach/retreat accepts movement relative to any external player; movement of the other player alone is insufficient.

For a fixed judge g and round e, let $b_{n,g,e}^{\mathrm{pre}} \in \{0, 1\}$ indicate that every pre-boundary WAIT window passes, and let $b_{n,g,e}^{\mathrm{post}} \in \{0, 1\}$ indicate that at least one post-boundary RESPOND window passes. For a split no-op case, $b_{n,g,e}^{\mathrm{post}} = 1$ instead requires every post-boundary WAIT window to pass. The

case passes only when both indicators are one, giving

$$\mathrm{JR}_{g,e} = \frac{100}{K_{\mathrm{JR}}} \sum_{n=1}^{K_{\mathrm{JR}}} b^{\mathrm{pre}}_{n,g,e} b^{\mathrm{post}}_{n,g,e}, \qquad K_{\mathrm{JR}} = 30. \tag{10}$$

This corresponds to $S^{\mathrm{JR}}_{g,e}$ in Equation 9. Cases judged as `unknown` receive zero credit and remain in the denominator.

B.4 Visual Quality (VQ)

Objective, data, and evidence. VQ evaluates visual readability and consistency independently of behavioral correctness on 30 reactivity-evaluation case conditions (from JR) per game. The judge receives the initial identity/appearance reference and three continuous future windows: 1–34, 35–67, and 68–100, with frames 0, 34, and 67 as their respective preceding context frames. Thus, all 100 future frames are covered with 34/33/33 scored frames, and each request has at most 36 images. No action instruction, trigger label, JR score, or future GT reference is provided.

Each window receives four independent integer scores from 0 to 4: clarity (clar), covering readable silhouettes and detail; identity/body structure (id), covering character identity, appearance, and coherent anatomy; background integrity (bg), covering scene structure and layout; and temporal stability (temp), covering continuity across adjacent frames. Clarity and identity/body structure require inspection of every character so that a damaged actor is not hidden by a clean background. Static waiting, native sprite flips, exaggerated poses, legitimate effects, and ordinary game animation are allowed. Incorrect but clearly rendered behavior affects behavior metrics, not VQ. Temporal deductions require evidence from consecutive frames.

Game-specific rubrics. HNM assigns 4 for no discernible defect, 3 for a minor/local/brief defect, 2 for an obvious or recurring defect, 1 for severe or persistent degradation, and 0 for an essentially unusable dimension. The released SF3 scores use a more lenient gameplay-demo rubric: 4 permits cosmetic blur, small texture defects, mild ghosting, or minor jitter when fighters and the scene remain readable and continuously trackable. A score of 3 denotes noticeable but generally interpretable artifacts; 2 denotes substantial disruption; 1 denotes severe loss of readability; and 0 denotes unusable evidence. Major identity replacement, body fragmentation, disappearance, or scene destruction still lowers SF3 scores. Therefore, the HNM and SF3 VQ values do not share an identical severity scale, although GT and generated videos within a game use the same rubric.

Aggregation. For a fixed judge g and round e, we suppress these two indices in the per-video scores. Let $z^{\mathrm{VQ}}_{n,\delta,w} \in \{0,1,2,3,4\}$ denote the score for video n, quality dimension $\delta \in \mathcal{D}_{\mathrm{VQ}} = \{\mathrm{clar}, \mathrm{id}, \mathrm{bg}, \mathrm{temp}\}$, and window $w \in \{1,2,3\}$. The scored frame counts are $(\nu_1, \nu_2, \nu_3) = (34, 33, 33)$, so $\sum_{w=1}^{3} \nu_w = F$. We first weight windows by their scored frame counts, then equally weight the four dimensions and the $K_{\mathrm{VQ}} = 30$ videos:

$$\begin{aligned} \bar{z}^{\mathrm{VQ}}_{n,\delta} &= \sum_{w=1}^{3} \frac{\nu_w}{F} z^{\mathrm{VQ}}_{n,\delta,w}, \\ \mathrm{VQ}_n &= \frac{100}{16} \sum_{\delta \in \mathcal{D}_{\mathrm{VQ}}} \bar{z}^{\mathrm{VQ}}_{n,\delta}, \qquad \mathrm{VQ}_{g,e} = \frac{1}{K_{\mathrm{VQ}}} \sum_{n=1}^{K_{\mathrm{VQ}}} \mathrm{VQ}_n. \end{aligned} \tag{11}$$

The denominator $16 = 4 \times 4$ accounts for four dimensions, each with a maximum score of four. The resulting 0–100 score is a rubric aggregate, not a success probability. Each judge's three round-level VQ scores are summarized using the same mean/sample-standard-deviation procedure as the behavioral metrics.

C MULTIPLE NPCS

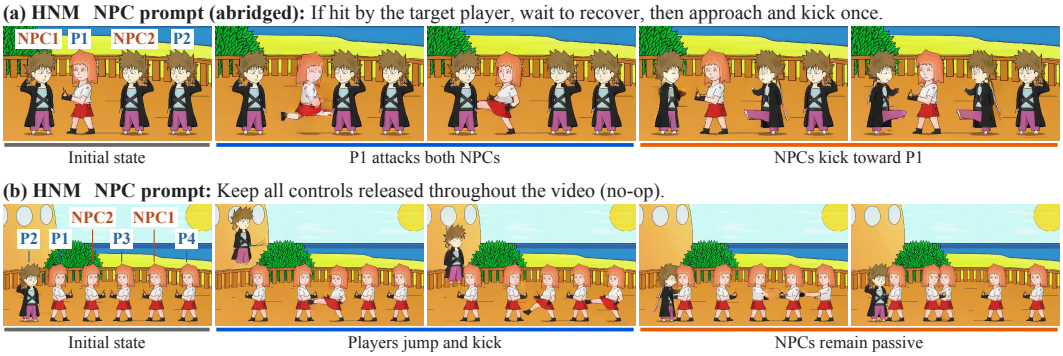


Figure 7: Multiple NPCs generated by ReactiveGWM in HNM. (a) P1 attacks NPC2 and NPC1 in turn; the NPCs then kick toward P1. (b) Two no-op NPCs remain passive while four controlled players act. Each row shows an excerpt from one generated sequence; the prompt applies to both NPCs and is abbreviated in (a). Labels indicate initial roles.

Figure 7 presents two HNM examples with two NPCs in the same generated scene. In (a), a player attacks the NPCs on either side, and both NPCs subsequently kick toward that player. In (b), four players perform different actions while two NPCs retain their passive behavior under no-op instructions. The two NPCs receive the same instruction within each example; their character bindings distinguish which instances the instruction applies to.

D QUALITATIVE RESULTS ACROSS REACTIVITY TYPES

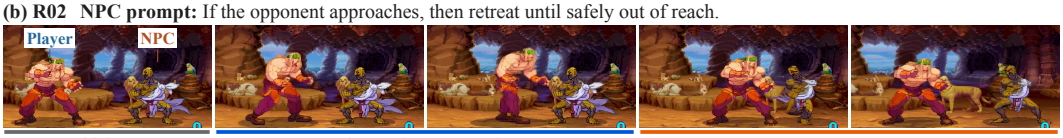
Figures 8–11 show one generated example for every rule family and the no-op condition in each game. Rule IDs follow Tables 5 and 4. Each row contains five chronological frames: the initial roles, a player event or cue interval, and the subsequent NPC behavior. Prompts marked "abridged" summarize the conditioning instruction for better displaying.

(a) R01 NPC prompt: If the opponent retreats, then approach until close.



Initial state | Trigger: Player retreats | Response: NPC approaches

(b) R02 NPC prompt: If the opponent approaches, then retreat until safely out of reach.



Initial state | Trigger: Player approaches | Response: NPC retreats

(c) R03 NPC prompt (abridged): If the opponent whiffs, then approach and punish before recovery.



Initial state | Cue: Player attacks | Outcome: Attack exchange

(d) R04 NPC prompt (abridged): If a threatening projectile gets close, then jump once to evade it.



Initial state | Trigger: Player throws a projectile | Response: NPC jumps

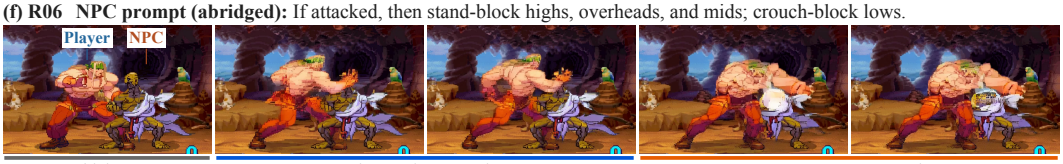
(e) R05 NPC prompt (abridged): If hit without blocking, then recover, approach, and counterattack once.



Initial state | Trigger: Player hits NPC | Response: NPC counterattacks

Figure 8: SF3 reactivity examples, R01–R05: approach, retreat, missed-attack punishment, projectile evasion, and counterattack after being hit. The opening event and subsequent behavior are grouped by the blue and orange rules.

(f) R06 NPC prompt (abridged): If attacked, then stand-block highs, overheads, and mids; crouch-block lows.



Initial state | Trigger: Player punches | Response: NPC guards

(g) R07 NPC prompt (abridged): If the opponent jumps in, then anti-air once as they descend within reach.



Initial state | Trigger: Player jumps in | Response: NPC anti-airs

(h) R08 NPC prompt (abridged): If the opponent's projectile gets close, then parry once and release.



Initial state | Trigger: Projectile approaches | Response: NPC parries

(i) R09 NPC prompt (abridged): If both are grounded and throwable, then throw once as the opponent enters range.



Initial state | Trigger: Player enters throw range | Response: NPC throws

(j) R10 NPC prompt: If this NPC is assigned no-op, then keep all controls released.



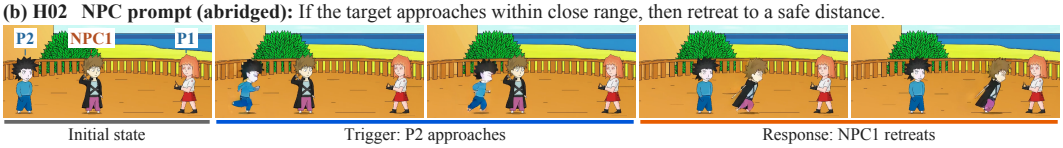
Initial state | Player activity | NPC remains passive

Figure 9: SF3 reactivity examples, R06–R10: guard, anti-air, parry, throw, and no-op. R06 shows a standing guard, and R07 shows an attack against the airborne player. R08 reuses the parry sequence from Figure 4(a), showing a blue parry effect followed by a return to stance. R09 shows a throw, while R10 shows the passive NPC during the player's movement.

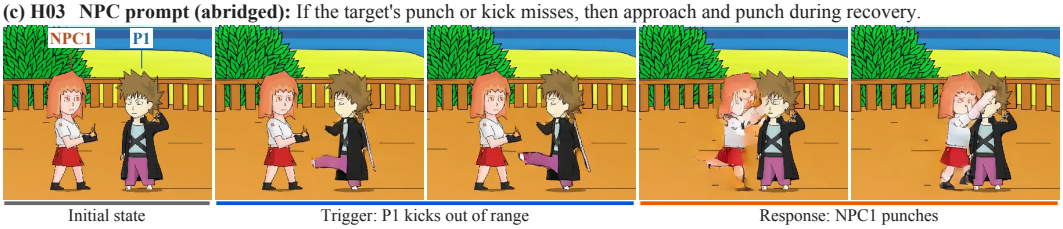
(a) H01 NPC prompt (abridged): If the target moves away, then approach until close.



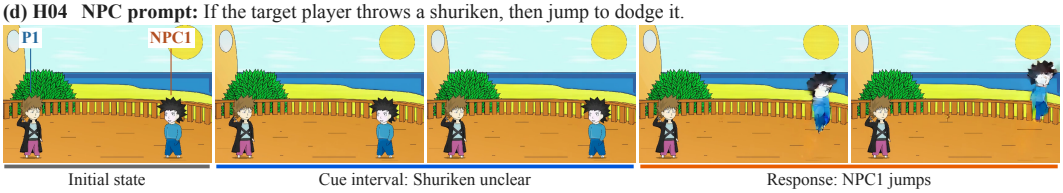
(b) H02 NPC prompt (abridged): If the target approaches within close range, then retreat to a safe distance.



(c) H03 NPC prompt (abridged): If the target's punch or kick misses, then approach and punch during recovery.



(d) H04 NPC prompt: If the target player throws a shuriken, then jump to dodge it.

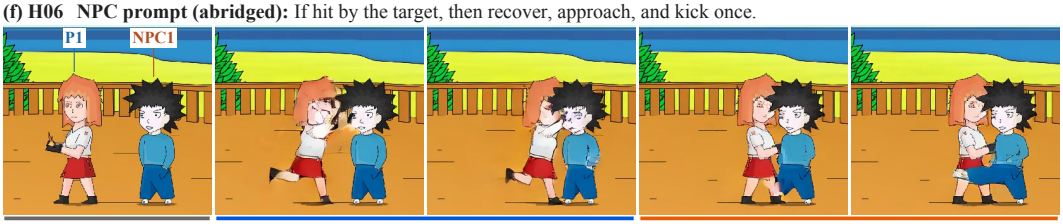


(e) H05 NPC prompt (abridged): If another player hits the target, then approach and kick as the target recovers.

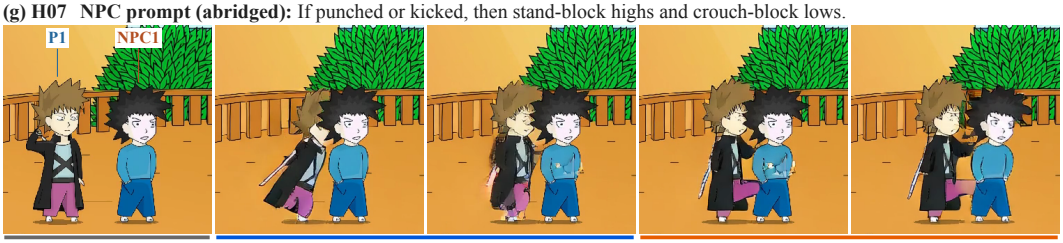


Figure 10: HNM reactivity examples, H01–H05: approach, retreat, missed-attack punishment, shuriken evasion, and response to another player hitting the target. H03 shows an out-of-range kick followed by an NPC punch, without establishing exact recovery timing. In H04 the jump is visible, but the shuriken is not clearly resolved. H05 retains all four characters to show the player interaction preceding the NPC's kick.

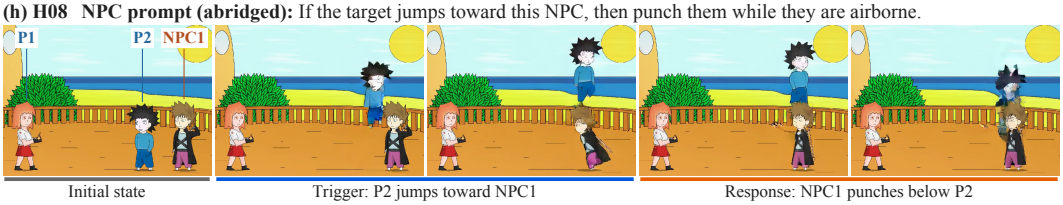
(f) H06 NPC prompt (abridged): If hit by the target, then recover, approach, and kick once.



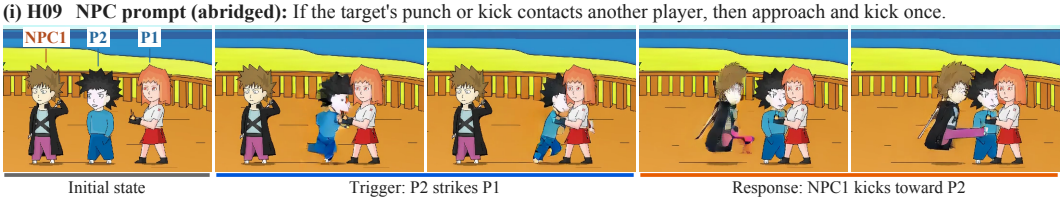
(g) H07 NPC prompt (abridged): If punched or kicked, then stand-block highs and crouch-block lows.



(h) H08 NPC prompt (abridged): If the target jumps toward this NPC, then punch them while they are airborne.



(i) H09 NPC prompt (abridged): If the target's punch or kick contacts another player, then approach and kick once.



(j) H10 NPC prompt: Keep all controls released throughout the video (no-op).

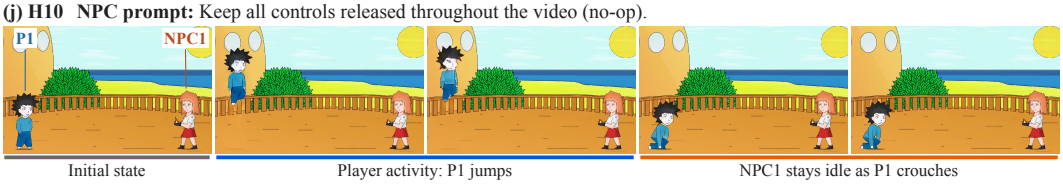


Figure 11: HNM reactivity examples, H06–H10: counterattack after being hit, guard by attack height, anti-air, response to the target hitting another player, and no-op. H07 shows a subtle forearm raise; correct attack-height matching is inconclusive. H08 shows a punch while P2 is airborne, with no clear contact. H09 shows a kick toward P2 after the player interaction, and H10 shows the NPC remaining idle while P1 jumps and crouches.