# SIMS: Simulating Stylized Human-Scene Interactions with Retrieval-Augmented Script Generation

Supplementary Material

## 8. More Ablation Studies

### 8.1. Direct Generation *vs.* RASG.

We compare our RASG method with direct LLM generation using GPT-4 [1]. For direct LLM generation, we provide the LLM with all the available skills as input. To evaluate the narrative diversity and generation efficiency of our approach, we measure the cosine similarity of SBERT [34] embeddings and the generation time. Our method achieves lower cosine similarity among the generated stories, indicating that it produces more diverse scripts. For generation time, we require the LLM to generate approximately 20 keyframes for direct generation method. For the RASG method, we ask LLM to retrieve 4-5 short scripts, which are approximately 20 keyframes in total. The results are evaluated on 200 generated samples separately.

| Method | SBERT Similarity [34]↓ | Average Generation Time(s)↓ |
|---|---|---|
| LLM | 0.8167 | 12.2 |
| RASG | **0.7759** | **7.32** |

Table 11. Ablation on script generation methods.

### 8.2. Generalization on Unseen Objects

In Tab. 12, we show the physical performance of interaction skills on PartNet [25] and 3DFront [7]. Note that our policies are only trained on the objects from 3DFront. From the table, we can see our results could achieve as good performance on unseen objects, mainly due to the generalization ability of heightmap design.

| Datasets | Success Rate(%)↑ | | Contact Error↓ | |
|---|---|---|---|---|
| | Sit | Lie | Sit | Lie |
| PartNet [25] | 98.7 | 87.6 | 0.028 | 0.065 |
| 3DFront [7] | 96.9 | 89.7 | 0.014 | 0.030 |

Table 12. Results on PartNet and 3DFront. The policies are trained on 3DFront's furniture only.

### 8.3. Ablation of Policy Settings

In Tab. 13, we conducted an ablation study on different settings of our control policy, comparing the *Success Rate* and *Contact Error* for variations without heightmap and without text embedding. Both variants showed degraded performance. The height map provides essential information about the surrounding environment so the performance becomes worse when interacting with objects. When trained without text embedding, the APD metric shows an obvious degradation.

| Setting | Success Rate(%)↑ | | | APD↑ | | |
|---|---|---|---|---|---|---|
| | Sit | Lie | Carry | Sit | Lie | Carry |
| w/o text | 89.7 | 89.6 | 92.4 | 16.29±0.22 | 16.59±0.28 | 12.41±0.19 |
| w/o htmp | 88.7 | 79.8 | - | 16.18±0.19 | 16.94±0.29 | - |
| SIMS(ours) | 96.9 | 89.7 | 96.4 | 16.52±0.47 | 16.99±1.28 | 14.92±0.23 |

Table 13. Ablation on different policy settings.

## 9. Reward Templates

In this section, we introduce the reward functions in 3 parts: locomotion (Loco), human-scene interaction (HSI), and dynamic object interaction (DOI).

- **Loco Reward.** The locomotion reward is defined in Equation 1. The overall reward comprises the far $r_t^{far}$, near $r_t^{near}$, and standstill $r_t^{still}$ rewards. The standstill reward ensures that the humanoid remains static once the target position has been reached. Given a target position $x^*$ of the character's root $x^{root}$, a target direction $d_t^*$, and a target scalar velocity $g_t^{vel}$, the task reward is defined as:

$$r_t^G = \begin{cases} 0.4\, r_t^{near} + 0.5\, r_t^{far} + 0, \left\|x^* - x_t^{root}\right\|^2 > 0.5, \\ 0.4\, r_t^{near} + 0.5 + 0.1\, r_t^{still}, \text{otherwise.} \end{cases} \tag{1}$$

$$\begin{aligned} r_t^{far} = {} & 0.6 \exp\big( -0.5 \left\|x^* - x_t^{root}\right\|^2 \big) \\ & + 0.2 \exp\big( -2.0 \left\|g_t^{vel} - d_t^* \cdot \dot{x}_t^{root}\right\|^2 \big) \\ & + 0.2 \left\| d_t^* \cdot d_t^{facing} \right\|^2 \end{aligned} \tag{2}$$

$$r_t^{near} = \exp\big( -10.0 \left\|x^* - x_t^{root}\right\|^2 \big) \tag{3}$$

$$r_t^{still} = \exp\big( -2.0 \left\|\dot{x}_t^{root} - \dot{x}_{t-1}^{root}\right\|^2 \big) \tag{4}$$

  The main difference between Walk and Idle reward is that we allow a large distance threshold for Idle. We restrict the Walk skill to reach the target coordinate as close as possible, but only restrict Idle to maintain inside 3 meters distance.
- **HSI Reward.** The HSI reward is defined in Eq 5. The far reward $r_t^{far}$ is to encourage the humanoid's pelvis $x^{root}$ to reach the target coordinate $x^*$ with the target speed $g_t^{vel}$ and target direction $d_t^*$. Like UniHSI [50], the near reward $r_t^{near}$ encourages the humanoid's certain joint to contact

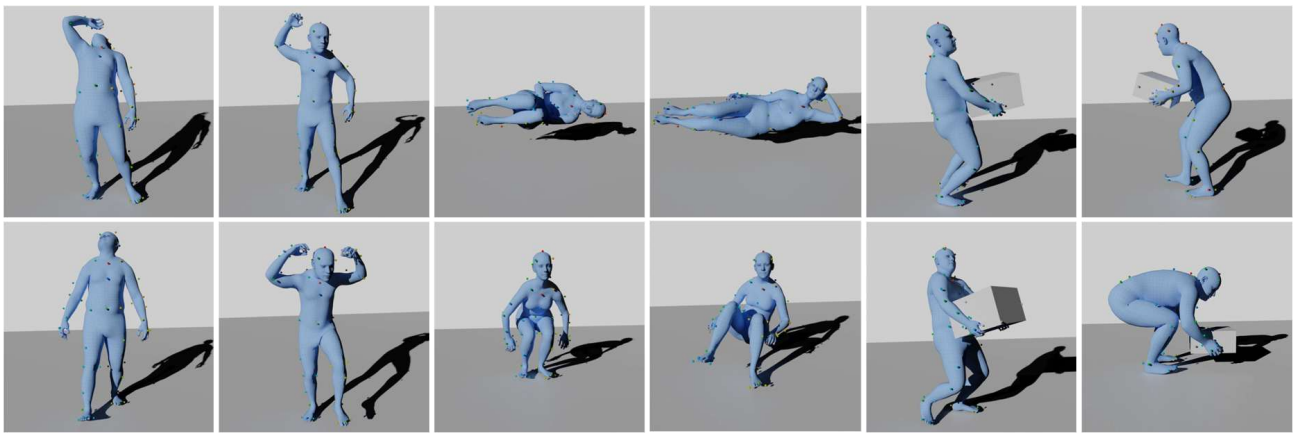


Figure 6. ViconStyle demos.

the nearest point in an interactable part $p$ of the target object. For Sit we require pelvis to contact the target sitting point, while for Lie we require pelvis to reach the nearest point on the bed's surface. For Reach, either left or right hand is supposed to reach the object's surface. The task reward is defined as:

$$r_t^G = \begin{cases} 0.7\, r_t^{near} + 0.3\, r_t^{far}, \left\| x_t^* - x_t^{root} \right\|^2 > 0.5 \\ 0.7\, r_t^{near} + 0.3, \text{otherwise} \end{cases} \tag{5}$$

$$r_t^{far} = \exp\left( -2.0 \left\| g_t^{vel} - d_t^* \cdot \dot{x}_t^{root} \right\|^2 \right) \tag{6}$$

$$r_t^{near} = \exp\left( -10.0 \left\| x_t^* - x_t^{root} \right\|^2 \right) \tag{7}$$

**Getup Reward.** The GetUp skill is developed through step goals, which combine walk and contact rewards. If the contact goal has not been reached, the reward encourages the humanoid to sit or lie on the object. Conversely, when the contact goal is achieved, the reward motivates the humanoid to elevate its pelvis to a standing position. The formulation for this reward system aligns with that of the contact reward $r_t^{near}$.

- **DOI Reward.** In this version, we only implement Carry skill in DOI task. However, our DOI reward could serve as a universal template for dynamic object interactions, like push, throw, etc. The reward is split into 3 parts: walk reward $r_t^{walk}$, encourages the humanoid walk to the object first; hand contact reward $r_t^{hand}$, encourages the humanoid place its hand on the object before the task been completed; moving reward $r_t^{carry}$, encourages to the object to the target position.

$$r_t^G = \begin{cases} 0.3\, r_t^{walk} + 0.5\, r_t^{carry} + 0.2\, r_t^{hand}, & \|x_t^{obj} - x_t^{goal}\|^2 > 0.5, \\ 0.3\, r_t^{walk} + 0.5\, r_t^{carry} + 0.2, & \text{otherwise.} \end{cases} \tag{8}$$

$$\begin{aligned} r_t^{walk} = {} & 0.8 \cdot \exp\left( -10.0 \cdot \|x_t^{root} - x_t^{obj}\|^2 \right) \\ & + 0.2 \cdot \exp\left( -2.0 \cdot \|v_t^{root} - v_t^{goal}\|^2 \right), \end{aligned} \tag{9}$$

$$r_t^{hand} = \exp\left( -0.5 \cdot \|x_t^{hand} - x_t^{obj}\|^2 \right) \tag{10}$$

$$\begin{aligned} r_t^{carry} = {} & 0.7 \cdot \exp\left( -10.0 \cdot \|x_t^{obj} - x_t^{goal}\|^2 \right) \\ & + 0.3 \cdot \exp\left( -2.0 \cdot \|v_t^{obj} - v_t^{goal}\|^2 \right). \end{aligned} \tag{11}$$

## 10. Re-implemented MotionCLIP

To control the policy language constraints, we aim to construct an embedding space fed into the policy network, where the embedding aligns motion representation with their corresponding natural language descriptions. To do this, we follow [17, 40], where a transformer auto-encoder is trained to encode motion sequences into a latent representation that aligns with the language embedding from a pre-trained CLIP text encoder [33]. Given a motion clip $\hat{\mathbf{m}} = (\hat{\mathbf{q}}_1, \ldots, \hat{\mathbf{q}}_n)$, a motion encoder $\mathbf{z} = \text{Enc}_m(\hat{\mathbf{m}})$ maps the motion to an embedding $\mathbf{z}$. The embedding is normalized to lie on a unit sphere $\|\mathbf{z}\| = 1$. We set the embedding size $\mathbf{z}$ to 64 to save the computation cost. For the text embedding, we first extract the feature with CLIP Encoder [33] $\text{Enc}_l$ from caption $\mathbf{c}$, then use a multilayer perception $\text{MLP}_d$ to downsize the 512 dim CLIP feature to 64

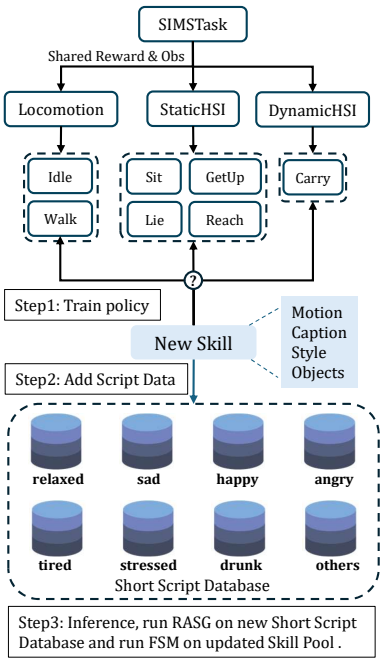


Figure 7. Scalability on new skills.

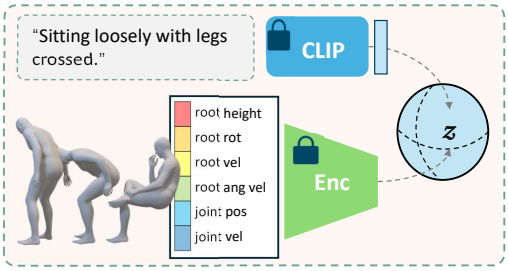


Figure 8. Our re-implemented MotionClip.

dim and use an extra one $\text{MLP}_u$ to upsample it to 512 dim to maintain the semantic feature. The embedding $\mathbf{z}$ should be aligned with the downsized CLIP feature. See details in Fig. 8 Following [40], $\text{Enc}_m(\mathbf{m})$ is modeled by a bidirectional transformer [4]. The motion decoder is jointly trained with the encoder to produce a reconstruction sequence $\mathbf{m} = (\mathbf{q}_1, \ldots, \mathbf{q}_n)$ to recover $\hat{\mathbf{m}}$ from $\mathbf{z}$. The motion representation $\mathbf{q}$ we use is a set of character motion features, following the discriminator observation used in AMP [30]. The auto-encoder is trained with the loss:

$$\mathcal{L}_{\text{AE}} = \mathcal{L}^{m}_{\text{recon}} + \mathcal{L}^{m,t}_{\text{align}} + \mathcal{L}^{t}_{\text{recon}}. \quad (12)$$

The reconstruction loss $\mathcal{L}^{m}_{\text{recon}}$ measures the MSE error between the reconstructed sequence and original motion.

The alignment loss $\mathcal{L}^{m,t}_{\text{align}}$ measures the cosine distance between the motion embedding and the downsized CLIP feature:

$$\mathcal{L}^{m,t}_{\text{align}} = 1 - d_{\cos}\left(\text{Enc}_m\left(\hat{\mathbf{m}}\right), \text{MLP}_d(\text{Enc}_l(\mathbf{c}))\right). \quad (13)$$

Figure 9. The motion capture environment of Vicon optical motion capture system.

The text embedding reconstruction loss $\mathcal{L}^{t}_{\text{recon}}$ measures the MSE distance between the reconstructed CLIP embedding and the original one:

$$\mathcal{L}^{t}_{\text{recon}} = \|\text{MLP}_u(\text{MLP}_d(\text{Enc}_l(\mathbf{c}))) - \text{Enc}_l(\mathbf{c})\|_2 \quad (14)$$

The weights of $\text{Enc}_l$ are fixed during training. To maintain the semantic information, we follow the sampling strategy used in MotionCLIP [40]. We sample 300 frames from the 30fps motion data and use skip sampling for the motion clips that are longer than 10 seconds so that all the information is included.

## 11. New Skill Scalability

In Fig. 7, we show the easy scalability of our framework. When new skills of new styles come, we need to train the corresponding skill based on the 3 kinds of templates, and expand the scripts database following the instruction of Sec. 3.1.

## 12. ViconStyle Dataset

We propose a comprehensive motion dataset called ViconStyle, in which well-labeled reconstructed motion clips with diverse styles and multiple skills are provided.

### 12.1. Capture Setting

The motion clips are captured with Vicon, an optical motion capture system, as shown in figure 9. All motion clips are captured with 120 fps. During the capture, we asked actors to interact with scene objects of different sizes and weights, such as lying on the sofa or carrying boxes.

We used SOMA [10] to fit the SMPL [19] body model and its pose parameters. The mocap data are then annotated with text descriptions containing motion details such as "hands on the thighs" and "lean back" and motion styles and emotions.

| Actors No. | Age | Gender | Height | Weight |
|---|---|---|---|---|
| 1 | 22 | Female | 168 | 55 |
| 2 | 22 | Male | 182 | 71 |
| 3 | 30 | Male | 175 | 85 |

Table 14. Actor information.

We also used a method to calculate the transformation and orientation and fit the size of the scene objects that we captured. We divide the reconstruction problem into two stages. In the first stage, we need to approximate the initial state of the scene objects. Since the scene objects are mainly boxes, the state estimation problem can be converted into an axis regression problem. We first regress the most suitable local coordinate by rotating the axis to minimum the max distance from the captured marker points to the axis. Then we move the origin point to the center of the bounding boxes of the marker points, and the scale can also be easily calculated. In the second stage, we trivially represent the subsequent transformation and orientation in the form of displacements and rotations relative to the initial frame.

### 12.2. Dataset Statistics

We recruited three actors to capture the dataset. The motion clips we captured contain 7 skills and actors are asked to perform in different styles and add details in every motion clip. The motion data set is 71.6 minutes in length and has 415 clips in total. The information of the actors is listed in table Tab. 14, and the detailed statistics of the data set are listed in table Tab. 2.

### 12.3. Qualitative Results

The captured motion containing diverse styles of Idle, Lie, Carry and GetUp skills. See Fig. 6 for demonstration.

## 13. Short Script Examples

We show some vivid examples in Tab. 15 for all the emotions/styles we use. Please check the skills, style label, object type, and captions, which are essential for FSM control.

| Summary: The character enjoys a **relaxed** afternoon in the living room. | | | |
|---|---|---|---|
| skill | style | object | captions |
| loco | neutral | - | smoothly forward walk |
| idle | relaxed | - | relaxing body |
| sit | relaxed | sofa | leaning back, legs straight, hands supporting head |
| getup | neutral | sofa | - |
| touch | - | shelf | - |
| **Summary: The character rushed **anxiously** through the living room.** | | | |
| skill | style | object | captions |
| loco | anxious | - | rush anxiously forward |
| touch | - | shelf | - |
| idle | anxious | - | pace around nervously |
| loco | hurried | table | walk with large steps |
| **Summary: Character felt utterly **tired** and sleep in the bedroom.** | | | |
| skill | style | object | captions |
| idle | tired | - | bent over with hands on knees |
| loco | tired | lamp | head bowed and body bent while walking |
| touch | - | lamp | - |
| loco | neutral | - | moving backward while walking |
| lie | tired | bed | lying down, legs straight |
| **Summary: The character **happily** played and relaxed around the bedroom** | | | |
| skill | style | object | captions |
| loco | happy | wardrobe | excited walk |
| carry | happy | toy | carry object happily |
| loco | happy | sofa | excited walk |
| sitdown | relaxed | sofa | hands support body, cross-legged |
| **Summary: The character is **angry** and knocks on the table, then sit.** | | | |
| skill | style | object | captions |
| loco | angry | - | angrily walking |
| idle | angry | - | stomp angrily against the ground |
| touch | table | - | |
| sit | angry | armchair | crossing arms |
| **Summary: The character gets **drunk** and stumbles around the living room.** | | | |
| skill | style | object | captions |
| idle | drunk | - | stand drunkenly |
| loco | drunk | sofa | walking drunkenly |
| sit | drunk | sofa | right leg held, left leg stretched out |
| touch | sofa | - | |
| loco | drunk | sofa | walking drunkenly |
| lie | tired | sofa | lying down, legs straight |
| **Summary: The character feels **stressed** and seeks comfort in the living room.** | | | |
| skill | style | object | captions |
| sit | stressed | armchair | sitting with head bowed, hands resting on thighs |
| touch | armchair | - | |
| loco | stressed | sofa | walking slowly, hands behind back |
| lie | stressed | sofa | side-lie on left with left arm as pillow, legs bent |
| Summary: The character discovered an old vase on the shelf, settled on the sofa. | | | |
| skill | style | object | captions |
| loco | neutral | | side-stepping |
| touch | neutral | shelf | - |
| carry | neutral | vase | carry object calmly |
| liedown | neutral | sofa | legs bend |

Table 15. Examples in the Short Script Database.