Hoi3DGen: Generating High-Quality Human-Object-Interactions in 3D

Supplementary Material

In this supplementary, we provide further details about our method implementation in Sec. 1 and experiment setup in Sec. 2. We then show additional results to demonstrate the controllability and generalization of our method in Sec. 3. Our code and models will be fully released to enable easy reproduction of our results.

1. Implementation Details

1.1. Segmentation and Reconstruction

In this Section we provide more details of the semantic separation of the human-object-interaction mesh. To separate the single watertight mesh, employ a three-stage mesh separation pipeline that relies on 1) controlled multi-view rendering, 2) open-vocabulary video segmentation, and 3) a per-vertex voting strategy from geometric consistency checks.

Multi-View Rendering. To obtain high-quality segmentation in the later stage, the rendering has to be of *high quality*, *clearly show the object*, and *have a smooth trajectory*. For high quality renders of our mesh, we use the open-source rendering engine of TRELLIS [1] and render 120 views per object. To facilitate high quality segmentation, we construct the camera trajectory as a multi-band spherical sweep around the mesh, which provides a broad view coverage and smooth viewpoint transitions. We partition the full set of views into several elevation bands from $[-60°, 60°]$ and perform a full $360°$ azimuthal sweep at a fixed elevation. To make the transition to the next elevation smooth, the azimuth direction alternates after each band.

All views are rendered at a constant distance and the same field of view, ensuring that the object stays at a consistent scale. After generating the full set of views, we cyclically rotate the sequence so that it begins at a diagonally elevated viewpoint, which we found to already give good enough segmentation quality. Otherwise, one could also run detections on each frame and chose the starting point based on the maximum confidence, however, we found this not to be neccessary. For each view, we then store the RGB image, the rendered depth map, and the camera transformation.

Open-Vocabulary Video Segmentation. The rendered RGB sequence is processed using Grounded Segment Anything 2 (GSAM2). We query the model twice: once using the text prompt "person" and once using the known object category. This yields two temporally coherent binary mask sequences, one for the human and one for object.

Vertex-Level Labeling and Mesh Separation. Given access to the depth map $\mathbf{D}_i$ and camera parameters for each view i, we project each mesh vertex $\mathbf{v}$ into all frames and record the views for which the vertex is 1) inside the image bounds and 2) passes a z-buffer consistency check to determine the closest vertex along the camera ray.

For each vertex, we inspect the object mask values at its projected location across all visible views. We compute the fraction of visible views where the vertex is marked as object. A vertex is then assigned to the object class, if this fraction exceeds a pre-defined threshold τ (in our case, we simply chose $\tau = 0.5$). Each triangle is considered part of the object, if two out of three connected vertices have been labeled as object. Finally, vertices and triangles are partitioned accordingly to obtained labels to obtain object mesh $\mathcal{O}$ and the remaining vertices and triangles form the human mesh $\mathcal{H}_m$.

1.2. SMPL fitting for base model

A common issue with the base image generation model is that it often generates only partial humans: sometimes only the torso is visible, sometimes only the lower body, and sometimes certain limbs are missing. Since CameraHMR always returns a full SMPL mesh $\mathcal{H}_s$, the Chamfer alignment fails due to mismatched scale and wrong matches.

To address this, we obtain a partial SMPL mesh $\mathcal{H}'_s$ that corresponds to the visible human regions. To obtain this partial mesh, we first use Grounded SAM 2 to compute a human mask $\mathbf{M}^h_{front}$ for the front-facing view of the 3D model (recall that CameraHMR can be utilized to calculate the front view). We then subset the SMPL vertices that fall inside this mask.

Formally, for an SMPL mesh with vertex set $\mathcal{V}$, we define the subset $\mathcal{V}'$ as:

$$\mathcal{V}' = \{\, \mathbf{v} \in \mathcal{V} \mid \mathbf{M}^h_{\text{front}}[\pi_{\text{front}}(\mathbf{v})] = 1 \,\}$$

where π_{front} denotes the camera projection for the front view. The vertices in $\mathcal{V}'$ define a partial SMPL mesh that can be reliably aligned with the partial generated mesh.

Finally, once the alignment is computed, we apply the transformation from the CameraHMR coordinate system to the mesh coordinate system, $\mathbf{T}_{\text{camHMR}\to\text{mesh}}$, to the full SMPL mesh so that accurate contact computation can be performed.

1.3. Animation

Given the SMPL mesh $\mathcal{H}_s$ with pose and shape parameters θ and β, respectively, we copy only the pose parameters from the animation sequence while keeping the original shape fixed.

Because the animation is performed in the unaligned SMPL coordinate space, we first transform both the human

mesh $\mathcal{H}_m$ and the object mesh $\mathcal{O}$ using the transformation $\mathbf{T}_{\text{mesh}\to\text{camHMR}}$, which is the inverse of the alignment transform that maps the SMPL mesh to the segmented human mesh.

Next, we transfer the linear blend skinning (LBS) weights by finding, for each vertex in transformed $\mathcal{H}_m$, its nearest neighbor in $\mathcal{H}_s$. The LBS weights from the corresponding vertex in $\mathcal{H}_s$ are then assigned to the vertex in $\mathcal{H}_m$.

For the object mesh, we attach it to the nearest SMPL joint. If multiple joints have similar distances to the object, we randomly choose one of them as the attachment point. Examples of animation can be found at Fig. 1

2. Experiment Details

2.1. More Details On Experimental Setup

For image-to-shape generation, we use *Hunyuan3D-DiT-v2.0* for geometry and *Hunyuan3D-Paint-v2.1* for texture synthesis. The *2.0* model provides more stable geometric results in our experiments, while *2.1* yields higher-quality, PBR-compatible textures.

For Grounded-SAM, we use *sam2.1-hiera-large.pt* for the detection component and *grounding-dino-tiny* to generate the initial candidates. The confidence threshold for GroundingDINO is empirically set to 0.4.

2.2. View Conditioned Sampling

In Section 4.4 of the main paper, we qualitatively demonstrated that view conditioning is essential for producing correct contacts. In Fig. 2, we further illustrate this effect. Although all three 2D images are correct, in the front view image (third row), occlusions lead to failures during 3D lifting. This highlights that multiple views are critical for obtaining accurate 3D generations.

2.3. GPT Score

We provide here the prompts used to compute the GPT Score shown in Fig. 3. For the 3D scores, we randomly select one of three available views for our case. For the 2D scores, we sample both the base model and our model three times to ensure a fair comparison.

One important detail is that GPT has limited understanding of fine-grained contact information. As a result, explicit contact points in the user prompts can introduce ambiguity. To mitigate this, we replace specific contact descriptions with more generic phrasing. For example, a prompt such as *"holding a bag in the left hand"* is changed to *"holding a bag in one hand"*. We hence evaluate the consistency is a less strict way.

2.4. User Study

In Fig. 4, we show the instruction screenshots used for our user study. We ask participants to assess *quality* first so that they are not biased to select human object interaction images, and actually focus on the quality related details. Then, in the second part, we explicitly specify our task that is human object interaction generation.

Body Part	**SANA**	**Ours**
Left Hand (↑)	40%	**100%**
Right Hand (↑)	90%	**100%**
Both Hands (↑)	10%	**100%**
Left Leg (↑)	11.11%	**80%**
Right Leg (↑)	30%	**70%**
Both Legs (↑)	90%	**90%**

Table 1. **Per-part contact accuracy.** Our model generates correct contacts for various contact scenarios whereas base model SANA [2] can follow mainly 'right hand' and 'both legs' but fails in other body parts.

2.5. Contact Accuracy

In Table 1, we demonstrate that our model generalizes well across diverse contact configurations, whereas the base model exhibits a strong bias toward only a few of them. Notably, for the *Left Leg* configuration, the base model produced extremely poor generations for one of the prompts, leading to a complete failure of the 3D lifting stage. To maintain a fair comparison, we report the average score over the remaining nine successful generations. Even under this favorable evaluation, the base model achieves correct contact only 11.11% of the time.

3. Additional Results

3.1. Controllability

Through our fine-grained text annotation and data filtering, we successfully enhance the interaction awareness of image generation model Sana. Interestingly, it also learns the decoupling of key components for interaction. We show in Fig. 5 that one can change the text description of human, object, action label, or contact regions. After which the model precisely follows the new text description while keeping the other parts barely untouched. This shows the superior text following capability of our model and makes it possible to repurpose our model as an interaction data generator.

3.2. OOD Generalization

Although our fine-tuning set contains only 400 images and we train for roughly 1050 epochs, raising potential concerns about overfitting, our results show otherwise. As illustrated in Fig. 6, our model not only generalizes robustly to previously unseen subjects, but also synthesizes plausible interactions with out-of-distribution objects and can generate coherent out-of-distribution actions.

3.3. More Qualitative comparison

In Fig. 7, we provide additional qualitative comparisons with current 3D baselines. A key advantage of our method is its reliability at inference. InterFusion, which relies on score distillation sampling, requires users to manually fine-tune parameters for each new generation. In contrast, our approach only needs to be fine-tuned once after which the same network generalizes to different 3D outputs.

3.4. Text Annotation Examples

In the Fig. 8, we show a few examples from our annotation pipeline. Our decomposed annotations, allows accurate annotation for human, object, contact and interaction. These annotations, help us in subsequent filtering and effective fine-tuning.

References

[1] Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. *arXiv preprint arXiv:2412.01506*, 2024. 1

[2] Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, and Song Han. Sana: Efficient high-resolution image synthesis with linear diffusion transformer, 2024. 2

TEXT INPUT

GENERATED 3D

"A man wearing denim overalls with a white t-shirt is holding a bright green backpack in his left hand."

ANIMATION

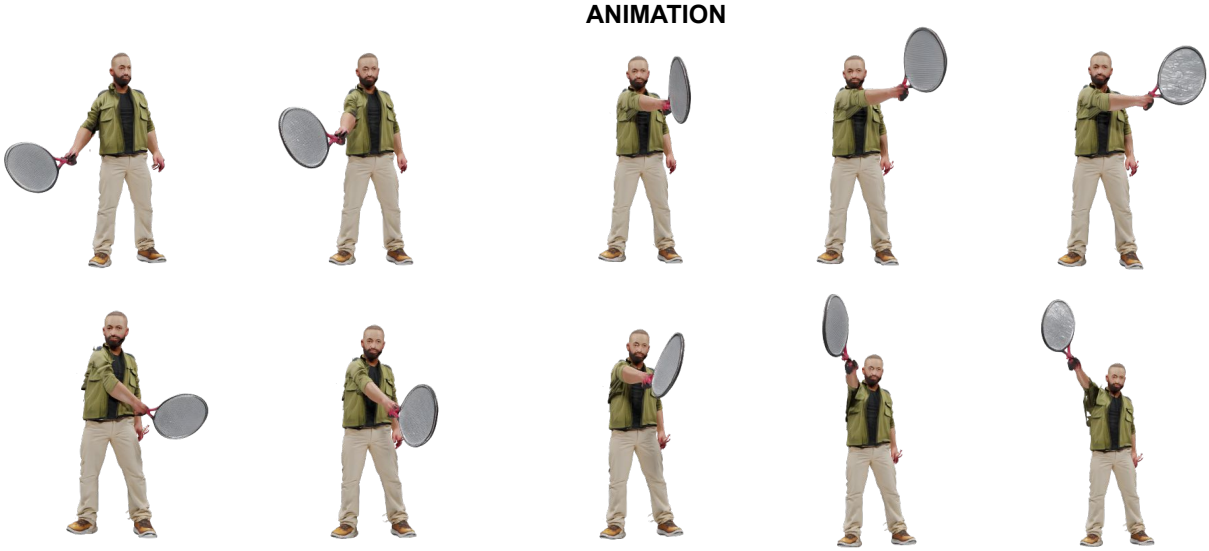
Animation Sequence 1: First the person right leg while spreading the arms, then lifts the left leg

TEXT INPUT

GENERATED 3D

"A man wearing khaki pants, a military jacket, a black t-shirt, and brown shoes is holding a tennis racket in his right hand."

ANIMATION



Animation Sequence 2: Person first hits a forehand, then a backhand

Figure 1. **Animation results.** Our fitted SMPL and segmented objects allow reanimation of the generated human object interaction mesh

Figure 2. **Advantage of view conditioned sampling.** Given same interaction prompt, our method generates three views that all correctly follow the contacts. Yet Hunyuan3D stuggle to reason the compositional shape under occlusion such as the leg occluded by the table in front view. By sampling side views as input to Hunyuan3D, we are able to generate at least one plausible 3D human-object interaction for each text prompt.

GPT 3D Text Alignment Prompt:
Given the prompt: <*Input Prompt*>
I will show you 3 different 3D model animations (shown as frames). Please select which animation best matches the prompt, and looks overall the best. Respond with only the number (0 to 3) of the best matching animation.
— Animation <animation num> —
<frame 1, frame 2, frame 3, frame 4>

GPT 3D Quality Score Prompt:
I will show you 3 different 3D model animations (shown as frames). Please select which animation has overall best quality, appeal, and physical plausibility. Respond as: Final Answer: <only the number (0 to 3)>.
— Animation <animation num> —
<frame 1, frame 2, frame 3, frame 4

GPT 2D Text Alignment Prompt:
Input Prompt: <*Input Prompt*>
Analyze the following 6 images and respond with ONLY the number (e.g., 1, 2, 3, etc.) of the image that best matches the 'Input Prompt'.
Image <image num>: <image>

GPT 2D Quality Score Prompt:
Given these 6 images and respond with ONLY the number (e.g., 1, 2, 3, etc.) of the image that best visual quality, realism, and plausibility.
Image <image num>: <image>

Figure 3. **Prompts used for calculating GPT scores.** We instruct GPT-4V to evaluate the text alignment and quality for generated 3D models (rendered as video) or 2D images.

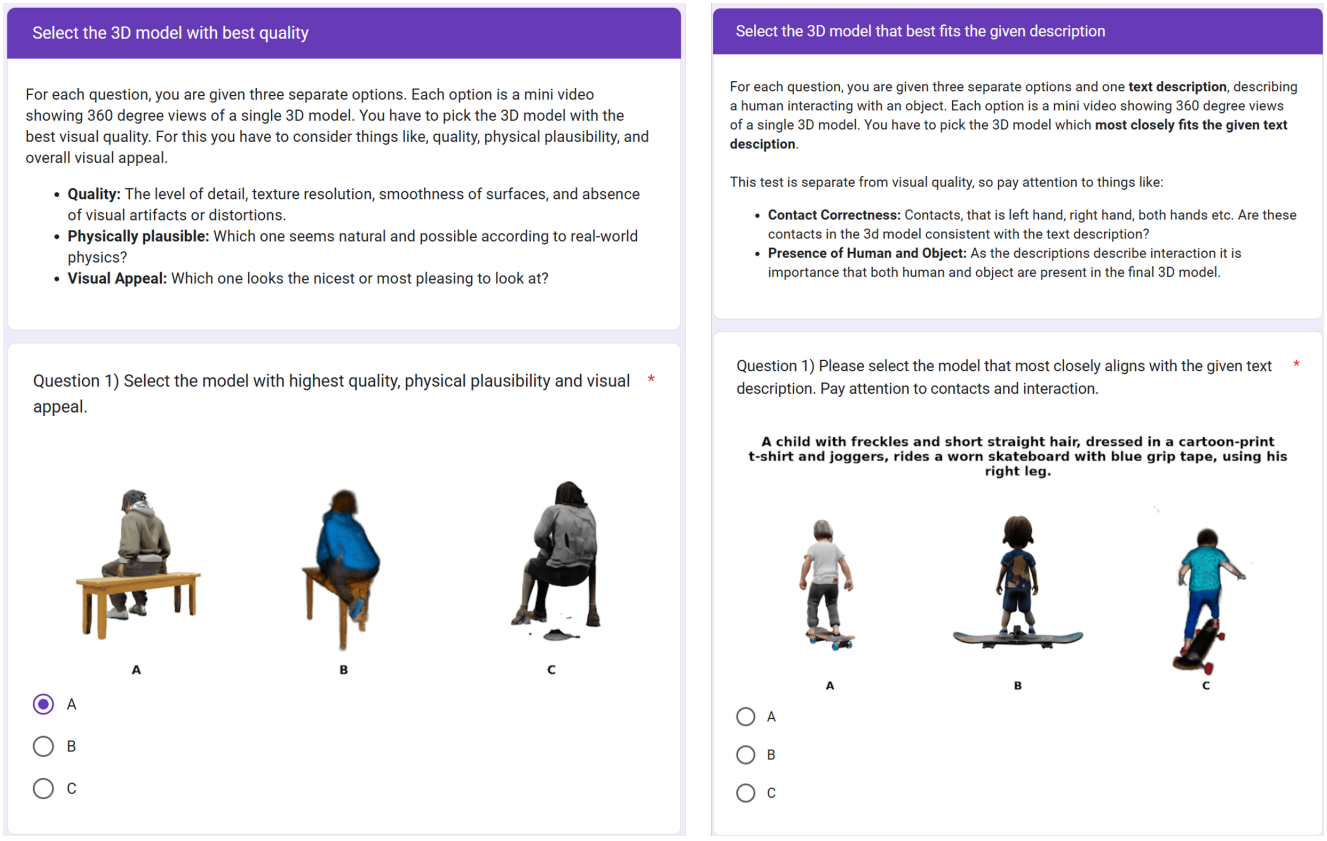


Figure 4. **User study instruction and example questions.** We guide participants to evaluate the quality or consistency to input text prompt. We randomly select 20 examples for each aspect and users are asked to choose the best one from three options.

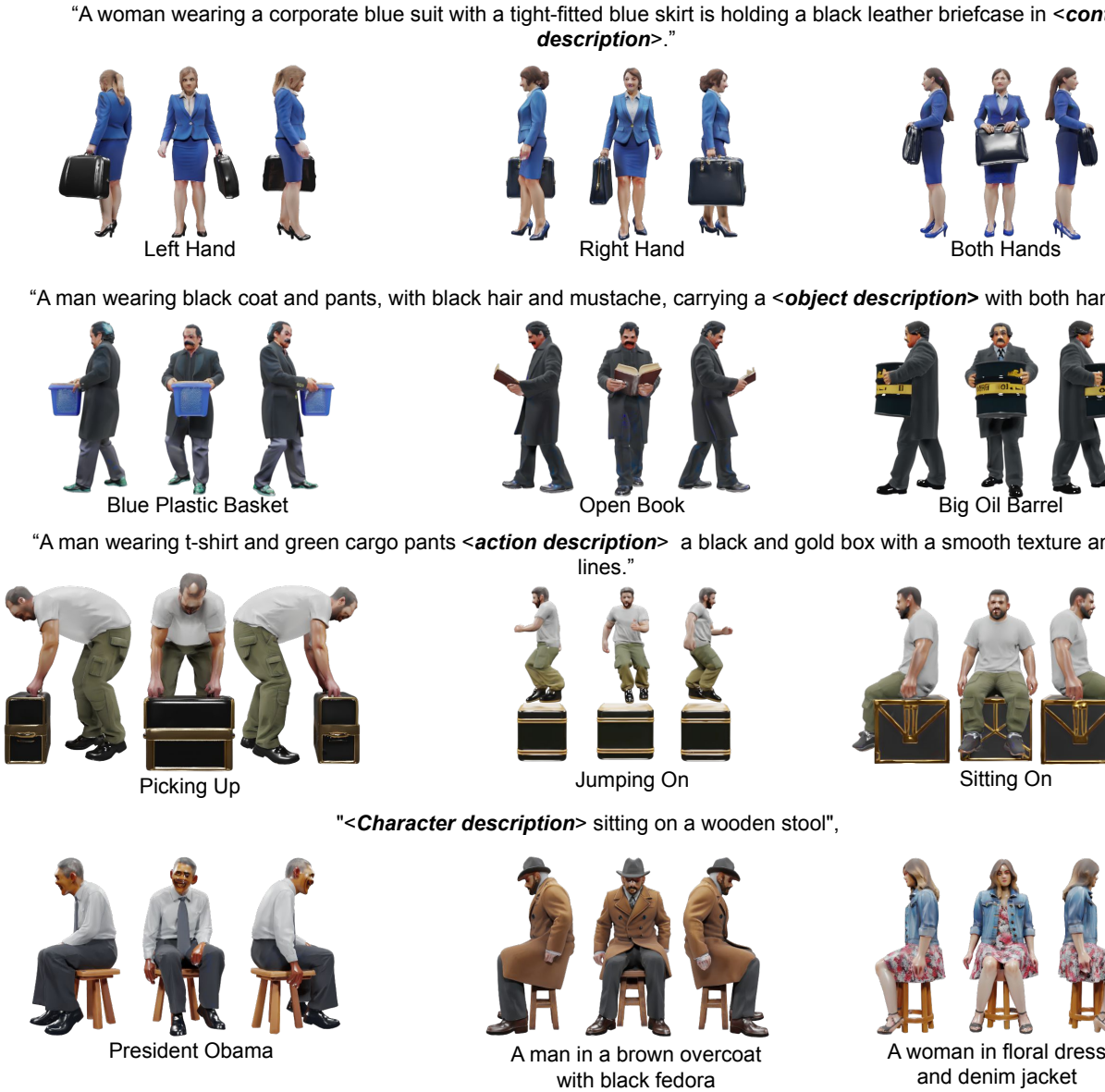


Figure 5. **Controllable interaction generation via text.** We can control the contact, object, action or human description in our text prompt with minimum changes to the other parts. This makes it possible to use our model as a data generator.

Figure 6. **OOD generalization.** Fine tuned only on interaction data with 100 humans and 15 object categories, our method generalizes well to different objects, human descriptions and new actions.

Figure 7. **More qualitative comparison.** Our method consistently produces high quality results with correct contact and details.

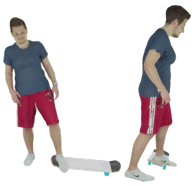
Dataset Renderings	Text Annotations	Dataset Renderings	Text Annotations	Dataset Renderings	Text Annotations
	“A middle-aged male with short hair, wearing a long-sleeve shirt with a logo, plaid shorts, and flip-flops, is carrying a green backpack using his right hand.”		“A female with a blue tank top, black leggings, and a ponytail hairstyle picks up a yellow backpack with black straps and a face design on the front with her right hand.”		“A male wearing a red t-shirt, blue jeans, and black shoes is carrying a blue backpack with white straps and zippers, contacting it with his hands, right shoulder, torso, right forearm, and right arm.”
	“Male with short brown hair, wearing a black and purple jacket, black pants, and black shoes, is holding a brown wooden box with visible grain texture and diagonal support structure using his right hand.”		“A young woman with short hair, wearing a grey sweater, a burgundy apron, black leggings, carries a wooden crate featuring a textured surface using her hands, hips, torso and thighs.		“A male with short dark hair, wearing a dark green "Hayes" t-shirt and black pants, picks up a beige box with a brown wooden frame using his left hand.”
	“A male with short brown hair, wearing a black t-shirt with a graphic, black shorts with a red design, and black shoes, sits on a dark wooden table, featuring a metal frame and four black caster legs, making contact with his hips, thighs, feet and hands”		“A male with short dark hair, light grey t-shirt, dark green shorts, and black shoes is lifting a minimalist wooden table with black metal legs and a light brown surface, contacting it with legs and hands.		“A male with short dark hair, wearing a green t-shirt, dark pants, and black shoes, lifts a light-colored wood dining table with his torso, hips, hands and thighs”
	“A young boy with short brown hair, wearing a dark blue t-shirt with a graphic and text, green cargo shorts, and colorful sneakers, balances on a white skateboard with black grip tape and four white wheels, using his right foot.”		“Male with black t-shirt, dark jeans, and glasses jumping on a white skateboard with smooth surface and four wheels, with no body part contacts.”		“A male with a dark blue t-shirt, red shorts with white stripes, white sneakers, and short dark hair is balancing on a skateboard with his left foot.”

Figure 8. **More examples from our annotated data.** Our annotation pipeline generates detailed descriptions about human, object, interaction action and contacts.