LiveHPS: LiDAR-based Scene-level Human Pose and Shape Estimation in Free Environment

Supplementary Material

In the supplementary material, we first provide a detailed explanation of our data processing procedures. Our capture system for the newly collected dataset encompasses multiple sensors, such as LiDARs, cameras, and IMUs. In Section. 7, we meticulously present the details of data processing. In order to enhance the dataset's diversity of actions and shapes for pretraining, we generate a large amount of synthetic data. Section. 8 elaborates on the methodology employed for generating synthetic point cloud data. Additionally, we emphasize that the FreeMotion dataset encompasses various shapes and actions with occlusions and human-object interactions. In Section. 9, we showcase nearly twenty human scans and forty distinct action types in FreeMotion. Moreover, in Section. 10, we show more experiment and qualitative evaluations of our ablation study. Lastly, in Section. 11, we present multi-frame results of LiveHPS in diverse scenarios, containing indoor, outdoor, and night scenes, to further demonstrate the robustness and effectiveness of our method for free HPS in arbitrary scenarios.

7. Details of Data Processing

FreeMotion is collected in two capture systems, the first capture system is the indoor multi-camera panoptic studio and the second capture system is in various challenging large-scale scenarios with multiple types of sensors. In this section, we provide more details about our data acquisition, data pre-processing, multi-sensor synchronization and calibration. The whole FreeMotion dataset will be made publicly available.

7.1. Data Acquisition

When we capture data in both capture systems, we divide the capture processing into three stages. In the first stage, the actor performs different kinds of actions alone without occlusion and noise. In the second stage, we arrange other persons to interact with main actor and perform specific actions depending on the scene characteristics, such as basketball court, meeting room, etc. The interactions bring real occlusions. In the last stage, the main actor interacts with some objects, such as the balls, package skateboard, etc. The objects bring real noise.

7.2. Data Pre-Processing

Point Cloud. The raw point clouds are captured from 128-beam OUSTER-1 LiDAR. The LiDAR features a 360° horizon field of view (FOV) $\times$ 45° vertical FOV. For the training dataset, we first record the point cloud of static background, and then set a threshold to delete the background points to get the point clouds of actors. For the processed point clouds, we use DBSCAN [15] cluster to get instance point cloud of each person. Then we employ the Hungarian matching algorithm for point cloud-based 3D tracking. Moreover, to promise high-quality annotations of our dataset, we review the segmentation point cloud sequence of each person. For the challenge and complex scene, such as the meeting room, which contains many extra objects, we use manually annotation for each point cloud sequence. For the testing dataset, we train the instance segmentation model [56] in our training set and inference in our testing set while the tracking processing is the same by above methods.

SMPL annotations. For the first capture system, we use multi-camera method [1] to generate the ground-truth SMPL parameters(Poses and translations). The predicted shape parameters is not accurate by this method. As the Figure 8 shows, we use the multi-camera method [43] to reconstruct the human mesh and fit the human mesh to the SMPL model in template pose for more accurate shape parameters. For the second capture system, we use full set of Noitom [39] equipment(17 IMUs) to get the SMPL pose parameters and the shape parameters of performer are captured in the panoptic studio in advance. In outdoor capture scenes, we follow [13] to fit the SMPL mesh to the point cloud for accurate translation parameters.

7.3. Multi-sensor System Synchronization and Calibration

Synchronization The synchronization among various-view and modal sensors is accomplished by detecting the jumping peak shared across multiple devices. Thus, actors are asked to perform jumps before the capture. We subsequently manually identify the peaks in each capture device and use this timestamp as the start for time synchronization.

Calibration Our capture systems contain LiDARs, Cameras and IMUs. For the three LiDARs calibration, we select the static background point clouds in the same timestamp, and manually to do the point cloud registration. For multi-camera calibration, we use [43] to calibrate all cameras. The calibration of IMUs is the algorithm of Noitom [39]. For calibration of each sensor, in the first capture system, we generate the SMPL human mesh from multi-camera data, the coordinate of human mesh is the same with the multi-camera coordinate, then we follow LIP [45] to utilize the ICP [5] method between the segmen-

tation human point cloud and the human mesh vertices for LiDAR-camera calibration. In the second capture system, we utilize the Zhang's Camera Calibration Method [68] for LiDAR-camera calibration, and the human SMPL mesh's coordinate is the same with the coordinate of IMUs. We use the above ICP method for LiDAR-IMU calibration.

8. Data Synthesis

In order to ensure the diversity of training data, we synthesize point cloud data on public datasets [34, 37, 51], which do not contain point cloud as input. In this section, we will explain our detailed implementations of data synthesis. LiDAR works in a time-of-flight way with simple principles of physics, which can be easily simulated with a small gap to real data. We generate simulated LiDAR point cloud by emitting regular lights from the LiDAR center according to its horizontal resolution and vertical resolution. The light can be reflected back when encountering obstacles, generating a point at the intersection on the surface of obstacles. We conduct the simulation according to the parameters of Ouster(OS1-128), which is also the device used in collecting FreeMotion. Its horizontal resolution is 2048 and its vertical resolution is 128 lines. Each emission direction is described by unit vector in spherical coordinate system $d = [\cos\varphi \sin\theta, \cos\varphi \cos\theta, \sin\varphi]$, where φ represents the angle between the emission direction and the plane XY, θ indicates the azimuth, and $c = [0, 0, 2]$ is the LiDAR center. The intersection point $p = [p_x, p_y, p_z]$ is calculated by

$$p = c + d\frac{n^T(q-c)}{n^T d}, \tag{10}$$

where n represents the normal vector of corresponding mesh and q denotes any vertex point of the mesh.

For the process of calculating the intersection of LiDAR and mesh surface, there are mainly three steps. The first step is to calculate the intersection of light and triangular patch, and the second step is to judge whether the intersection of light and plane is inside the triangle. Due to occlusions, one light should only have the intersection with the first touched mesh of object. Finally, we filter the intersections to only keep the ones first occurred in the LiDAR view.

Since the above datasets focus on single-person scenarios, we randomly crop some body parts of the point clouds to simulate the occlusions in real scenes:

$$\begin{aligned} index &= dis(p-o) > r, \\ pc_{crop} &= pc[index], \end{aligned} \tag{11}$$

where o represents a random point position in point cloud pc as the round dot, and r denotes the radius of the crop area. The function $dis()$ means the distance between the two point positions. The pc_{crop} is the synthetic point cloud with occlusion.

9. Details of Dataset

We collect FreeMotion in diverse multi-person scenarios, including various sports venues as well as daily scenes. We calculate the types of action and the shapes of human covered in different scenarios. As the Figure. 8 shows, FreeMotion contains diverse human shapes, and we use large synthetic data to obtain more human with different shapes, such as SURREAL [51]. As the Figure. 9 shows, FreeMotion comprises nearly forty distinct action types, providing point cloud data from three different views and corresponding ground-truth mesh. The action types are arranged from top to bottom and from left to right, we begin by showing some common warm-up actions performed in daily routines, such as the leg pressing, kick or lumbar movement, etc. These fundamental motions are essential for sports activities and represent simple motions with minimal self-occlusion. However, they are still affected by external occlusion, as depicted in the left point cloud of the "Lumbar Movement" and the right point cloud of the "Side Stretch". The more challenging daily motion type for point cloud is the squatting, which involves serious self-occlusion, such as the bend, crawling, and sitting down. Moreover, squatting motions are prone to lose information due to external occlusion, as shown in the middle point clouds of "Crawl" and "Long Jump". Furthermore, we present sports motions captured in large-scale multi-person sports scenes. These motions have greater challenges due to more self-occlusion, as defined by the LIP [45] benchmark. In FreeMotion, our sports motions exhibit more severe issues of external occlusion, resulting in point clouds with only a few points or even no points due to occlusion caused by other actors. This can be observed in the middle point clouds of "Football" and "Ping-Pong". Lastly, we also capture data in various daily walkway scenes, including crossing obstacles, carrying shoulder bags and backpacks, etc. These types of motion frequently occur in daily life, and the most challenging issues are lots of human-irrelevant noise and occlusion caused by the interactive objects. We adhere to privacy guidelines in our research. The LiDAR point clouds naturally protect privacy by omitting texture or facial details. Additionally, we mask faces in RGB images to uphold ethical standards in our dataset.

10. More experiments

In this section, we show more experiments and more qualitative results of our ablation study.

our LiveHPS can provide high-quality results, notably in datasets like CIMI4D, where our results often align more closely with point clouds than global ground truth mesh. With SUCD metric, we can sift through a bunch of high-quality estimations as pseudo-labels to supplement lacking SMPL annotations in LiDARHuman26M [33], LIPD [45]

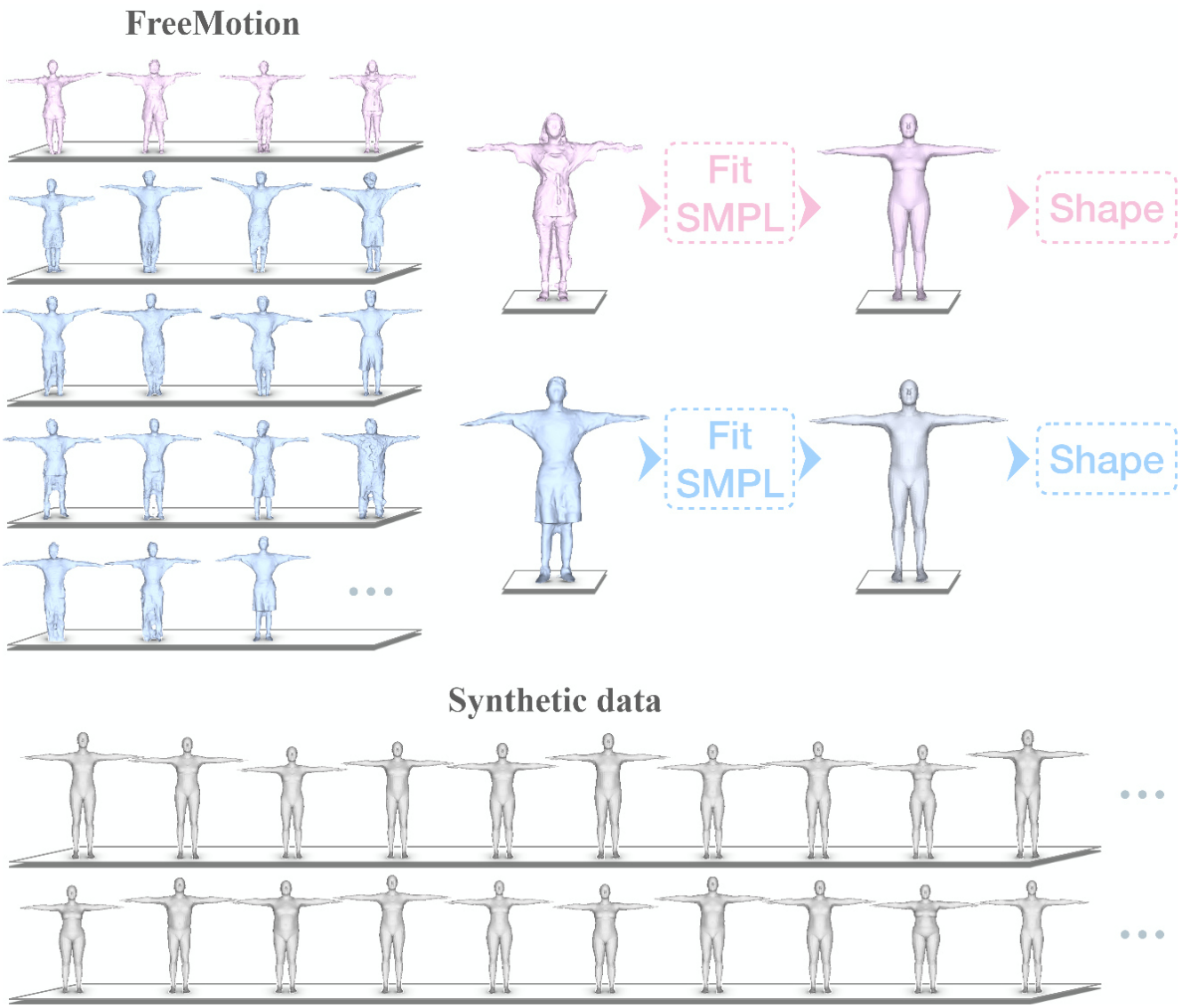

Figure 8. Diverse shapes of human in FreeMotion and synthetic dataset. We display the human reconstruction meshes and the process of generating shape parameters.

Table 6. Quantitative evaluation with fine-tune LiveHPS on each dataset. For the datasets with incomplete annotations, we use N/A for the metrics without annotation.

	LiveHPS				
	J/V Err(P)↓	J/V Err(PS)↓	J/V Err(PST)↓	And Err↓	SUCD↓
LiDARHuman26M [33]	65.34/81.17	-	-	18.69	2.62
LIPD [45]	56.71/70.53	-	-	13.97	1.75
CIMI4D [62]	85.82/105.77	86.33/106.14	-	25.02	3.65
FreeMotion	66.38/80.10	66.68/80.52	116.72/125.46	15.59	2.71

and parts of our FreeMotion, particularly in outdoor multi-person scenes lacking dense IMU annotation. These pseudo-labels are then incorporated into our training set to fine-tune our LiveHPS. Tab.6 displays the enhanced performance post fine-tuning, affirming LiveHPS's effectiveness and robustness, and its potential as an automated human pose and shape annotation tool.

We show the qualitative evaluation of the ablation study, the red box means the incorrect local motion. As the Figure. 10 shows, the network without consecutive pose optimizer module is unable to capture coherence features in temporal and spatial, which causes the big mistake in rotation estimation even the joint positions are basically correct. And the network without vertex-guided adaptive distillation module also performs bad in local motions when the input point cloud is occlusion, such as the foots and hands. As shown in Figure. 11, the network without temporal information also cause the same mistake in rotation estimation. Our method performs better especially in local motions, such as the head and hands. As shown in Figure. 12, when the point cloud is occlusion, our method can maintain

stability, while others clearly affected by the occlusion of point clouds, such as the right hands in first row, the legs of STGCN result and left hand of GRU result in second row. Finally, we evaluate the translation estimation, the MOVIN provides the velocity estimation method for translation estimation, which causes severe cumulative errors, and other methods all can provide a basically correct result. In order to observe the accuracy of translation prediction more accurately, we subtract the predicted translation from the original point cloud to obtain the point cloud in the origin point. As the Figure. 13 shows, our method still maintain stability when the actor is squatting.

11. More Results of LiveHPS

In this section, we present additional multi-frame panoptic results of LiveHPS, which further illustrate the effectiveness and generalization capability of our method for estimating the accurate local pose in diverse challenging scenarios, encompassing indoor, outdoor, and night scenes.

Indoor scene. Most existing LiDAR-based mocap datasets [33, 45] primarily focus on capturing data in outdoor scenes to leverage the wide coverage capabilities of LiDAR sensors. Moreover, indoor data often present challenges such as lots of noise and occlusion in the point cloud caused by surrounding complex furniture in limited ranges. As shown in Figure. 14, three actors are engaged in indoor speech motions, involving tasks such as cleaning the desktop, adjusting the screen, giving a presentation, and writing on a whiteboard. We also display image reference in three views, in some case, actors may fall outside the camera's range. External occlusions arise due to the presence of other actors (purple actor in "Adjust the Screen") and indoor facilities (cyan actor in "Write on Whiteboard"). Despite these challenges, our method demonstrates reliable performance in handling occluded point clouds, showing its robustness in complex indoor environments.

Outdoor scene. Benefiting from long-range sensing properties of LiDAR, FreeMotion can capture multi-person motions in various large-scale scenes. As shown in Figure. 15, we collect the data in the basketball court measuring 28 meters in length and 15 meters in width. The sequence shows a fast-break tactic involving three actors, and external occlusion primarily occurs during instances when the actors scramble for the basketball, as observed in the right part of the sequence. Despite the challenging conditions, our method exhibits excellent performance even in situations where the point cloud becomes extremely sparse due to actors being distant from the LiDAR sensor or being occluded by others. These results highlight the potential applicability of LiveHPS in sports events.

Night scene. LiDAR sensors operate independently of environmental lighting conditions, which can work in the completely dark environments. In Figure. 16, we present the sequential results of LiveHPS captured on the square at night. The scene depicts a densely populated area with complex terrain. The LiDAR can capture seventeen persons in the region measuring 24 meters in length and 20 meters in width. In contrast, the image reference is significantly unclear due to the absence of adequate lighting. Nevertheless, our method demonstrates reliable performance in this challenging scene, demonstrating that LiveHPS can be applied for arbitrary real-world scenes.

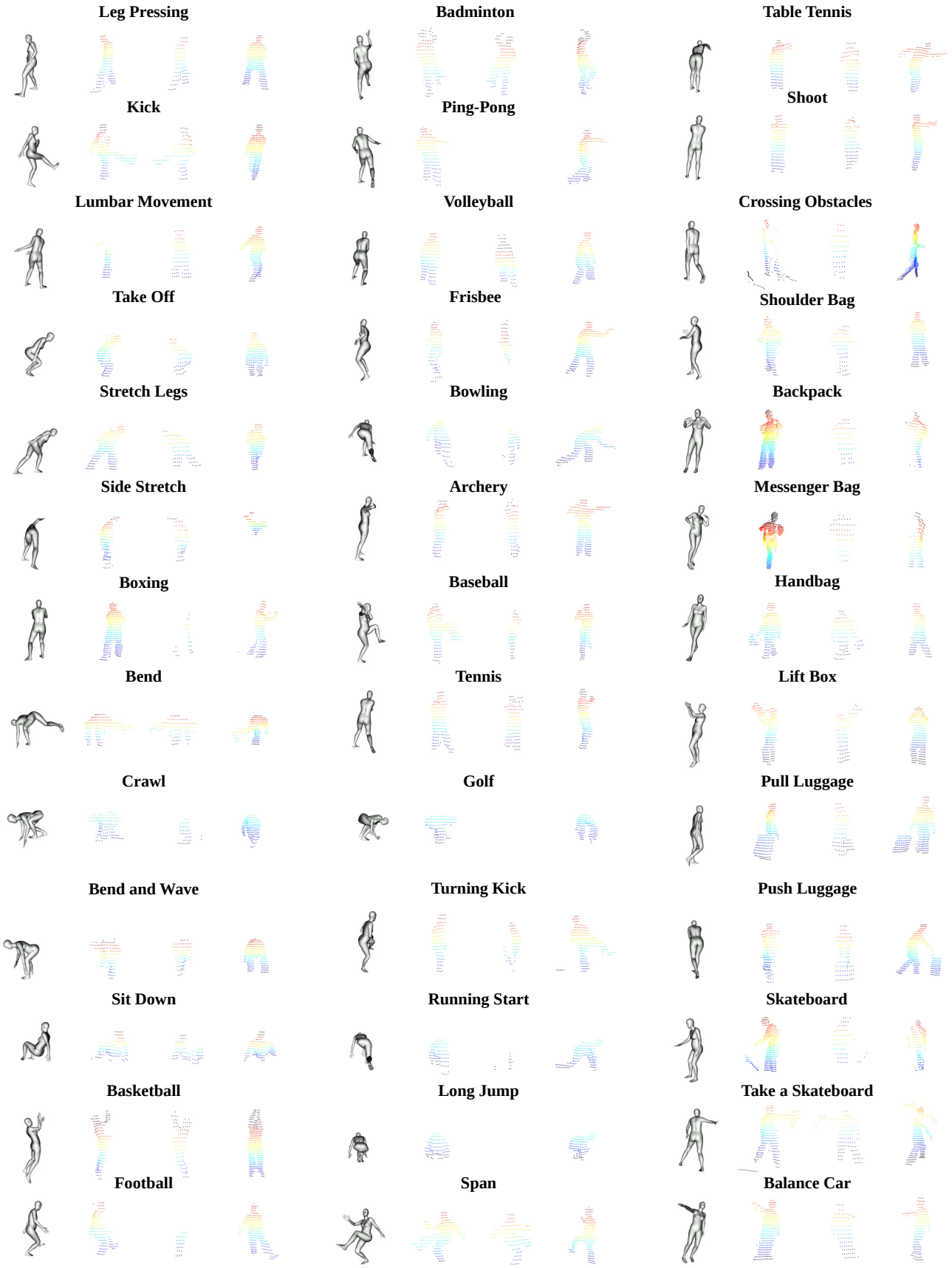

Figure 9. Different types of action in FreeMotion. We display the point cloud from three different views and corresponding ground-truth mesh for each type of action. The left point cloud is the main view, which is matching with the ground-truth mesh, the middle point cloud is the back view and the right point cloud is the side view.

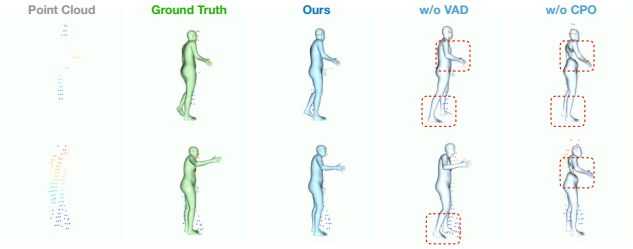

Figure 10. Qualitative evaluation on our network modules.

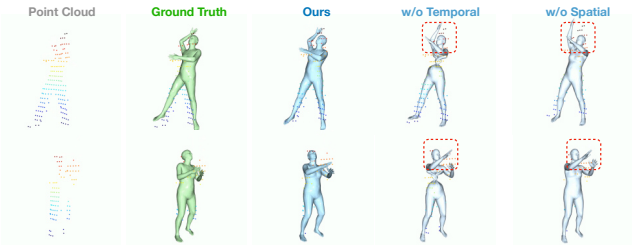

Figure 11. Qualitative evaluation on different optimization configurations of CPO module.

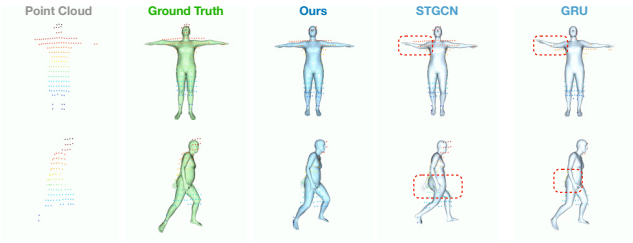

Figure 12. Qualitative evaluation on our attention-based Inverse Kinematic Solver.

Figure 13. Qualitative evaluation on our network modules. The blue line means the height of point cloud with ground-truth translation, the green line means the height of point cloud with our predicted translation, and the red line means the point cloud with LIP predicted translation or average locations.

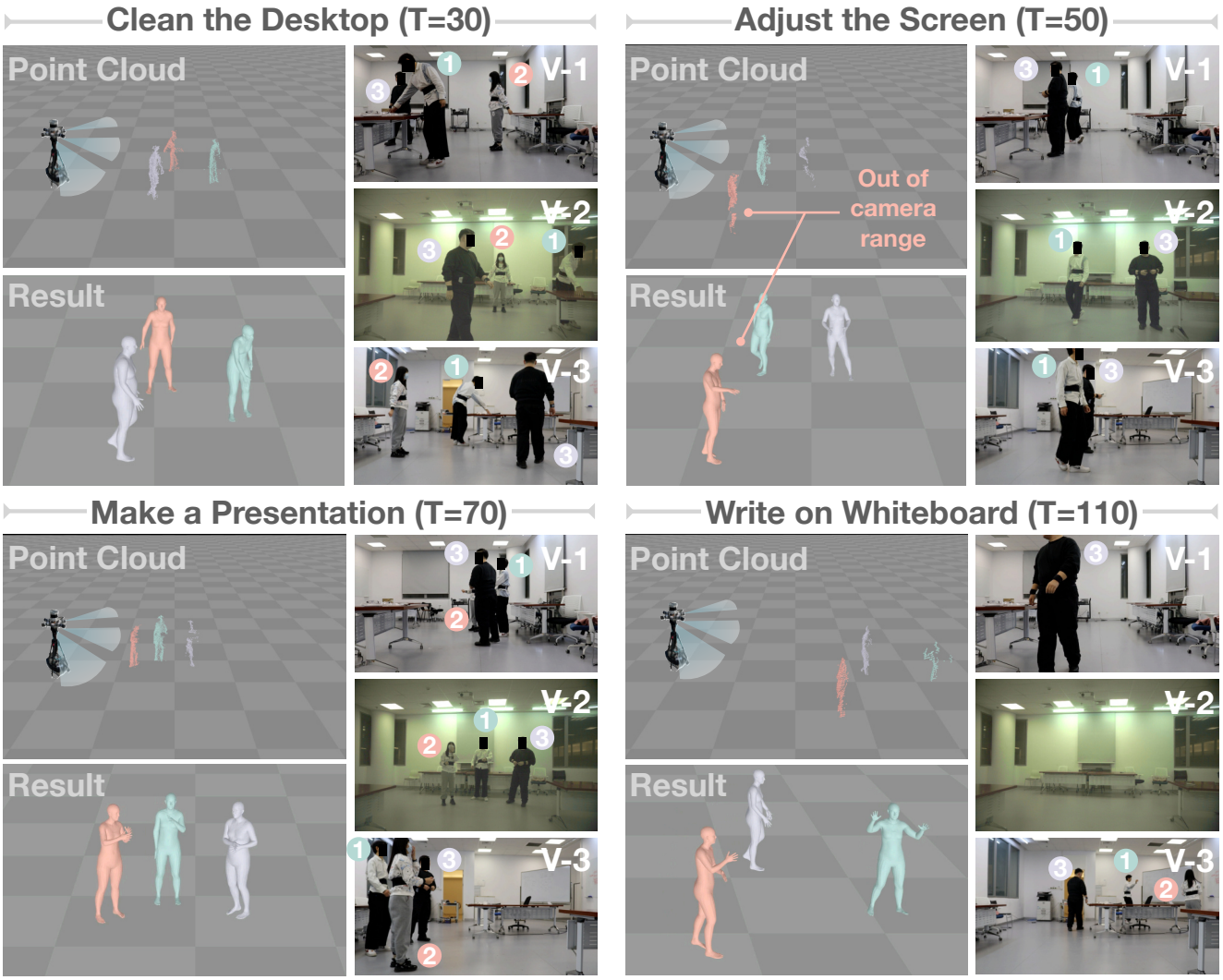

Figure 14. Sequential result of LiveHPS in indoor scene. We select four time points to show the indoor speech motion. The "T" means the timestamp, and the time unit is seconds. For each timestamp, we show point cloud, the corresponding result of LiveHPS and the image reference, "V-x" with same color means the three views of camera in the same timestamp. Digital labels with different colors on the image corresponding the point clouds with the same color. Note, for showing more details, we zoom in the results.

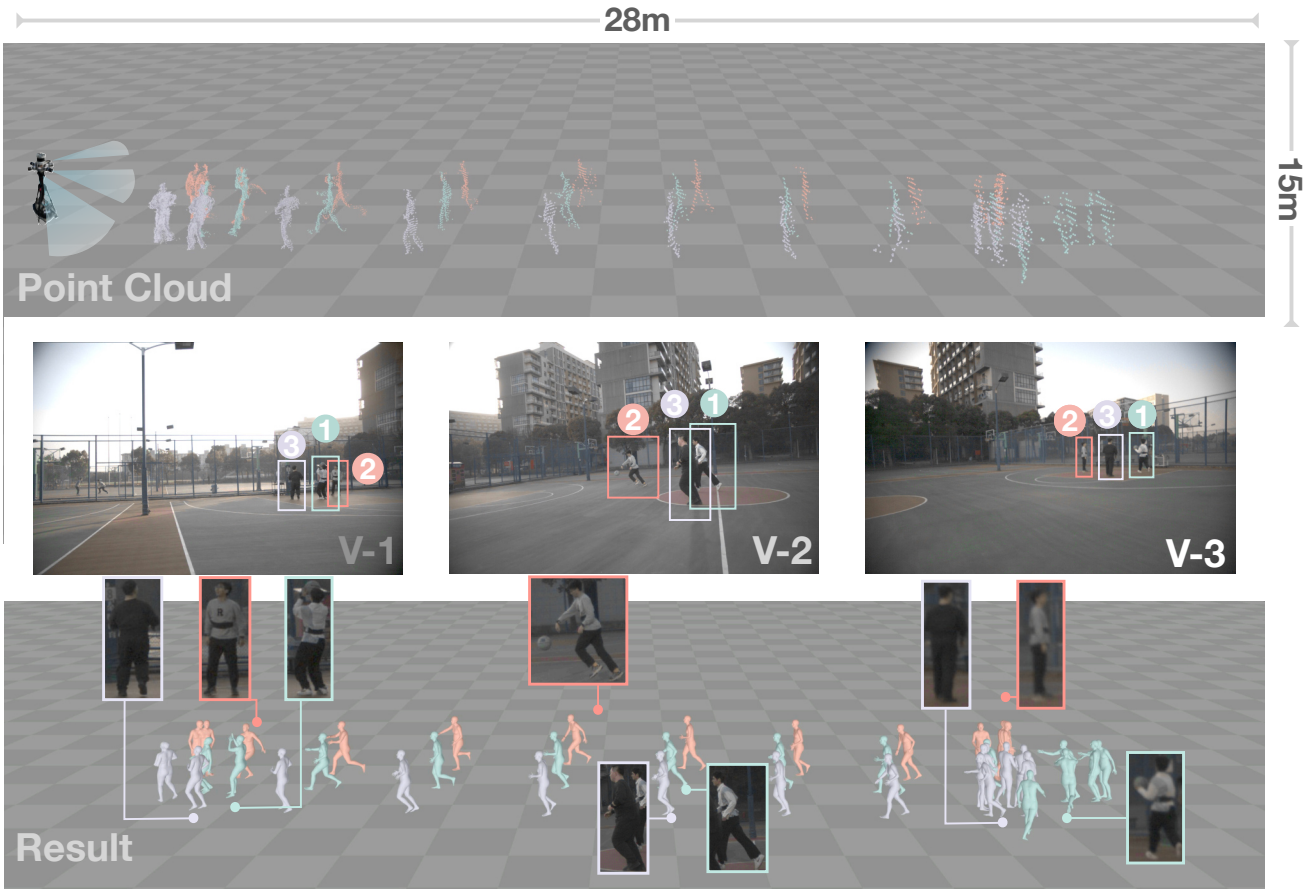

Figure 15. Sequential result of LiveHPS in outdoor scene. We show the sequential point cloud and results with time step of 10 seconds. For showing the image reference in three views, we select three different timestamp for each view, the "V-x" with different colors means different timestamp in view-x(x=1,2,3). Digital labels with different colors on the image corresponding the point clouds with the same color.

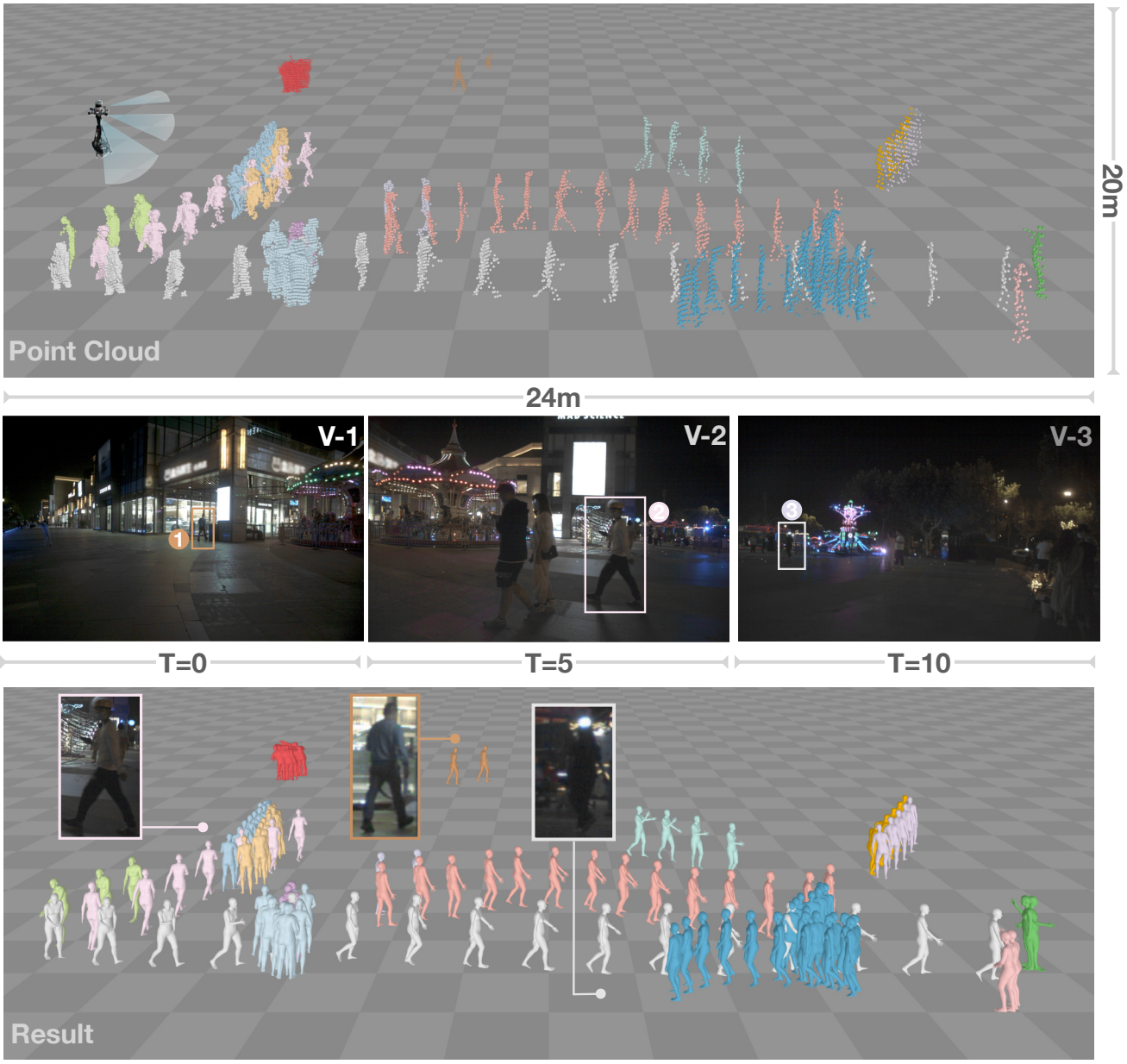

Figure 16. Sequential result of LiveHPS in night scene. The time step of the sequential point cloud and results is 10 seconds. The "V-x" with different colors means different timestamp in view-x(x=1,2,3). T means the timestamp of the image and the time unit is seconds. Digital labels with different colors on the image corresponding the point clouds with the same color.