WheelPose: Data Synthesis Techniques to Improve Pose Estimation Performance on Wheelchair Users

William Huang
University of California, Los Angeles
California, USA
whuang37@g.ucla.edu

Sam Ghahremani
University of California, Los Angeles
California, USA
samg2024@berkeley.edu

Siyou Pei
University of California, Los Angeles
California, USA
sypei@g.ucla.edu

Yang Zhang
University of California, Los Angeles
California, USA
yangzhang@ucla.edu

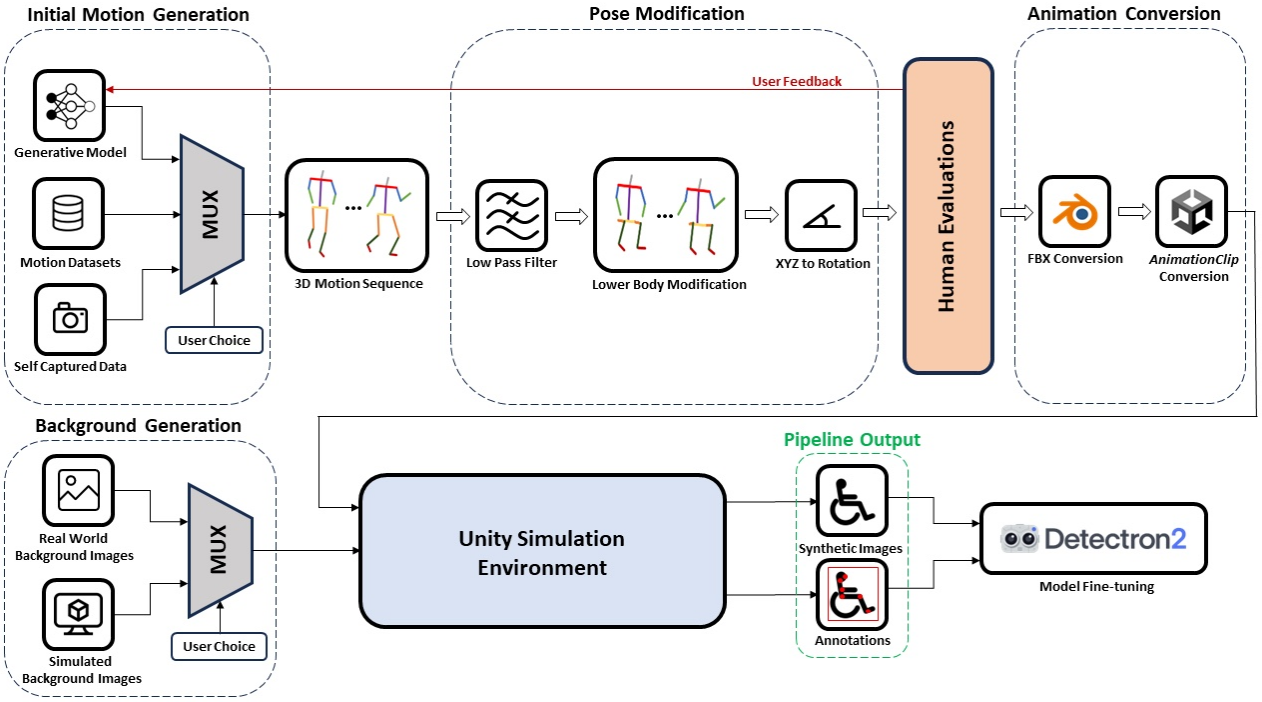


Figure 1: Overview of the full *WheelPose* data generation pipeline. Developers can choose different motion sources. Motion sequences are modified according to the specification stated below before being evaluated by human evaluators. Developers can regenerate, filter, and clean motion sequences from human evaluations before all motions are converted into Unity readable *AnimationClips*. Converted motion sequences, selected background images, and parameters are used in simulation to generate synthetic images and their related annotations for use in model boosting.

ABSTRACT

Existing pose estimation models perform poorly on wheelchair users due to a lack of representation in training data. We present a data synthesis pipeline to address this disparity in data collection and subsequently improve pose estimation performance for wheelchair users. Our configurable pipeline generates synthetic data of wheelchair users using motion capture data and motion generation outputs simulated in the Unity game engine. We validated our pipeline by conducting a human evaluation, investigating perceived realism, diversity, and an AI performance evaluation on a set of synthetic datasets from our pipeline that synthesized different backgrounds, models, and postures. We found our generated datasets were perceived as realistic by human evaluators, had more diversity than existing image datasets, and had improved person

CHI '24, May 11–16, 2024, Honolulu, HI, USA

ACM ISBN 979-8-4007-0330-0/24/05
https://doi.org/10.1145/3613904.3642555

detection and pose estimation performance when fine-tuned on existing pose estimation models. Through this work, we hope to create a foothold for future efforts in tackling the inclusiveness of AI in a data-centric and human-centric manner with the data synthesis techniques demonstrated in this work. Finally, for future works to extend upon, we open source all code in this research and provide a fully configurable Unity Environment used to generate our datasets. In the case of any models we are unable to share due to redistribution and licensing policies, we provide detailed instructions on how to source and replace said models. All materials can be found at https://github.com/hilab-open-source/wheelpose.

CCS CONCEPTS

• **Human-centered computing → Accessibility systems and tools**.

KEYWORDS

Accessibility, Data Synthesis, Wheelchair Users, Pose Estimation

ACM Reference Format:

William Huang, Sam Ghahremani, Siyou Pei, and Yang Zhang. 2024. WheelPose: Data Synthesis Techniques to Improve Pose Estimation Performance on Wheelchair Users. In *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '24), May 11–16, 2024, Honolulu, HI, USA.* ACM, New York, NY, USA, 25 pages. https://doi.org/10.1145/3613904.3642555

1 INTRODUCTION

The inclusiveness of AI depends on the quality and diversity of data used to train AI models. We focus on pose estimation models, which have found widespread use in health care, environmental safety, entertainment, context-aware smart environments, and more. These models are a major concern in the push for AI fairness due to the disparity in their accuracy of predicted postures between able-bodied users and users with disabilities [38, 95, 106]. Focusing on human movement, Olugbade et al. [73] surveyed 704 open datasets and found a major lack of diversity. The authors found no datasets that included people with disabilities performing sports, engaging in artistic expressions, or simply performing everyday tasks. We suspect that the lack of diversity has contributed to biases and poor performance on users with disabilities in many popular AI models trained on common human movement datasets like Detectron2 ImageNet [108]. This is especially apparent in users who use mobility-assistive technologies (Figure 2). The lack of disability representation in training data can be directly attributed to poor accessibility in the data collection process for people with disabilities [38, 74]. People with disabilities often must overcome more challenges in data collection during their commute and communications in the recruitment and participation process. One such example and the focus of our work are wheelchair users, who may not be able to navigate through motion capture rigs easily. Additionally, certain poses that able-bodied users could easily perform might be difficult or even dangerous for people with motor impairments. To improve disability representation in training data in the push for inclusive AI, we must make data collection equitable across people with *all* levels of capabilities.

Prior work has combated this issue by proposing guidelines in the design of studies [14] and online data collection systems that could be more accessible to people with disabilities [74]. We propose an alternative solution to the data collection problem: synthetic data. Like how synthesized materials could help preserve scarce natural resources – synthetic rubbers were developed in part because of concerns about the availability of natural rubber – synthetic data could be valuable in supplementing insufficient data collection from people with disabilities. While traditional motion capture procedures lie on a spectrum of difficulty dictated by the motion of the user and exacerbated by the innate difficulties of data collection with users with assistive technologies, synthetic data offers a more accessible alternative where different actions, settings, and assistive technologies, ranging from cooking at home to performing backflips in the forest, are equal in the difficulty of implementation and feasibility.

In this research, we present a novel data synthesis pipeline which leverages motion generation models to simulate highly customizable image data of wheelchair users. Our approach includes steps for user-defined parameters, data screening, and developer feedback. This pipeline yields image data which can be used to improve the performance of AI models for wheelchair users. We evaluate this in the case of pose estimation by fine-tuning common pose estimation models, trained on common human movement datasets, with our synthetic data. Fine-tuned models are then tested on a new dataset of wheelchair users to analyze the degree of improvement from adding synthetic data to training datasets.

Finally, as we are cautious about the problematic simulation of disabilities (e.g., blindfolded participants to simulate people without vision or with low vision), synthesized wheelchair user postures are carefully reviewed in human evaluations to avoid inadvertently exacerbating existing equitability problems our approach attempts [14]. Our goal is not to exclude wheelchair users from AI training, but rather present a data collection solution that enables them to shepherd the synthesis of data. In doing so, this research leverages data as an intuitive way for wheelchair users to impact AI training, through which we hope to produce more fair and inclusive AI models.

Through our work, we aim to answer the following research questions and related sub-questions in the context of wheelchair users and pose estimation problems:

(1) **RQ1:** How to ***effectively generate synthetic data***?
   - **RQ1.1:** How can we model wheelchair users?
   - **RQ1.2:** What are the controlling parameters in synthetic data generation?
(2) **RQ2:** What are the ***efficacies of synthetic data***?
   - **RQ2.1:** How do individual parameters of synthetic data generation affect pose estimation performance?
   - **RQ2.2:** What are the benefits and drawbacks of using synthetic data?

To summarize, our contribution is three-fold:

- adoption of data synthesis to improve inclusion of AI.
- a custom data-synthesis pipeline for pose tracking with improved performance for wheelchair users.
- investigations of the efficacy of the overall approach.

(a) (b) (c) (d) (e)

Figure 2: Examples of poor keypoint prediction performance from Detectron2 ImageNet [108]. Figure 2(a) Poor prediction of the legs and torso which are slightly occluded by the wheelchair. Figure 2(b) Both shins are predicted to be on the hand due to the occlusion of the wheelchair. Figure 2(c) Legs are predicted to be on the upper body. Figure 2(d) The right shin is predicted to be on the wheels of the wheelchair. Figure 2(e) The wheelchair dancer is completely undetected by ImageNet.

2 RELATED WORK

2.1 Pose Estimation

Pose estimation plays a key role in fields like the animation and video industry [1, 89]. Developments in deep learning have enabled users to circumvent the need for cumbersome marker suits in traditional motion capture and directly generate human postures from camera outputs [3, 66, 109, 114]. Of particular note is 2D posture recognition, where the recent releases of easily accessible pipelines like MediaPipe [65], OpenPose [20], and BlazePose [12] have enabled widespread access to posture recognition in a wide variety of applications including biomechanics [72], autonomous driving [2], sign language interpretation [71], and more.

2.2 Sensing for People with Limited Mobility

Currently, over 8.5% of the population of the world is age 65 or over. This number is projected to grow to nearly 17% of the world's population by 2050 [44]. Given the direct correlation between age and and mobility limitations and disabilities, this trend implies a growing need for mobility-related technologies [32, 33, 36]. Our research focuses on the community of wheelchair users, where technologies like SpokeSense [22] have established themselves as a part of a broader focus in research related to developing smart wheelchairs [55] which integrate different sensors, including camera, lidar, and EEG, to make wheelchairs more comfortable and safe. Other works focus on developing more accessible control systems [98] or routing systems [53] for users with mobility impairments. Posture estimation techniques for wheelchair users can reveal a user's sitting habits, analyze their mood, and predict health risks such as pressure ulcers or lower back pain [67].

2.3 Synthetic Data for Computer Vision

Computer vision models have traditionally been trained using large-scale human-labeled datasets such as PASCAL VOC [30], Microsoft COCO [64], and ImageNet [24]. While effective, these datasets are costly to produce, requiring large amounts of publicly available images, manpower, and time to create. Furthermore, these datasets are often static and offer little in the form of customizability to allow researchers and engineers to use data specific to their task. One solution to these problems is the use of data simulators. SYNTHIA [82], Synscapes [107], Hypersim [81], and OpenRooms [61] provide synthetic datasets for computer vision tasks related to object detection in different settings. Other robotics simulators including AI-2THOR [54], NVIDIA Isaac Sim [62], Mujoco [93], and iGibson [86] offer a rich set of tools for embodied AI tasks. Other systems, like BlenderProc [25], BlendTorch [45], NVISII [70], and Unity Perception [11] prefer to instead enable the developer to generate their own data through highly configurable simulators. These tools and datasets have already demonstrated considerable success in deep learning-related training tasks [6, 43, 91].

Synthetic humans provide further challenges due to the complexities of human bodies and the variations and limitations of a human's appearance and posture. Various approaches have been taken, using different types of simulators to generate labeled datasets. Examples of different approaches include deriving data from hand-crafted scenes [9], custom 3D scenes with SMPL models [13, 76, 78, 79, 100, 111], existing games like Grand Theft Auto V [18, 19, 31, 48, 49], and game engines [27, 28]. We were inspired by this line of work and extended upon the existing PeopleSansPeople (PSP) data generator with the Unity Perception package [11] using domain randomization principles which help AI models trained in simulated environments to effectively transfer to real-world tasks [92].

2.4 Evaluating the Quality of Synthetic Data

Despite the advantages of data synthesis, an implicit assumption of using synthetic data is that it should be sufficiently high-quality to achieve performance similar to real data. To evaluate the quality of synthetic data, researchers have explored a wide range of metrics. Emam et al. [29] outlined three types of approaches to assess synthetic data utility in their book - workload-aware evaluations, generic assessments, and subjective assessments of data utility. Among them, workload-aware evaluations check if synthetic data replicates the performance of real data, widely used in data synthesis research [35, 77, 101]. Generic assessments measure the utility indicators of real and synthetic data when the indicators are quantifiable and clear [46, 52] (e.g., the distance between their statistical indicators such as mean, average, and distribution). Furthermore, subjective assessments involve real users to evaluate data realism. Some researchers investigate distinguishability, assuming highly-realistic synthetic data leans to be perceived as real [87], similar

to deploying a discriminator in algorithms [96]. Other researchers design Likert-Scale questionnaires for realism, which are broadly adopted in clinical training simulation [11]. Another criterion is to collect user preferences between several synthetic samples to form a high-quality dataset [90, 105]. With a consistent and valid examination of synthetic data, researchers can obtain feedback to improve generation methods and understand how reliable synthetic data is. In our work, we not only conduct workload-aware evaluation but also involve subjective assessments by asking real users to evaluate the data realism. The user-in-the-loop process provides filter handles for more realistic datasets under various contexts and inspires key findings on how data realism affects performance.

3 *WHEELPOSE* DATASET SYNTHESIS

We address **(RQ1.1)** with our system, *WheelPose*, a data synthesis framework where motion data is converted to wheelchair user animations and rigged on human models in a Unity simulation environment to generate synthetic images and annotations. We present a simple simulation environment, with human models randomly placed in front of a background as the most primitive example of synthetic data still capable of generating positive results in pose estimation (Section 4.3). A visualization of the overall pipeline is found in Figure 1. Our pipeline generates a set of datasets, each including 70,000 captured frames (1280×720). Each frame is fully annotated using the COCO 17-keypoint 2D skeletal model, shown in Figure 5(b). Beyond the fact that our dataset is the first fully annotated dataset of wheelchair users, the size of our dataset is comparable with existing datasets like 3DPW with 51,000 captured frames [69] or SMPLy with 24,428 captured frames [57]. Example synthesized data is shown in Figure 3.

3.1 Generating Postures

Our data synthesis pipeline begins with pose generation, where motion data is converted into animations to be used in data synthesis. We choose to use two motion data sources, HumanML3D [39] and Text2Motion [40] in our case study. Other motion sources can be easily adapted and used within our pipeline. Figure 4 demonstrates 14 motion sequences and their resulting postures from our pose generation, documented next. We separate posture generation from the rest of our pipeline to allow developers to iterate upon generated postures through human evaluation, regenerating and filtering data as needed.

3.1.1 Human Skeletal Modeling. In order to enable a wide variety of different postures, we base the implementation of *WheelPose* on a 23-keypoint skeletal model, which is easily converted from commonly available human posture datasets including COCO [64] and MPII [7] and the output of common pose estimation algorithms including BlazePose [12], MediaPipe [65], and OpenPose [20]. Detailed information on these keypoints is shown in Figure 5(a). We note that our current pipeline assumes users have all four limbs and acknowledge that data synthesis for wheelchair users with amputation requires efforts beyond simple ad-hoc removals of key points in our current model. Nonetheless, we interviewed two participants with limb loss, leading to insights for future work which we will discuss later in the paper (Section 4.1).

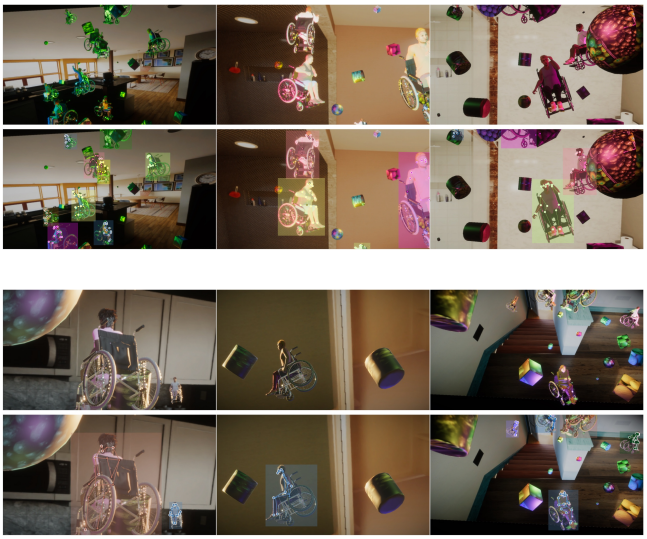


Figure 3: Example output from the full *WheelPose* data generation pipeline. The top row includes the generated RGB images. The bottom row includes the generated RGB images with the keypoint and bounding box annotations superimposed on top. Keypoints outlined in black are labeled as "occluded" while keypoints outlined in white are labeled as "visible". Each image is generated with randomized backgrounds, lighting conditions, humans, postures, and occluders.

Figure 4: Example pose frames in their motion sequences resulting from our pose frame generation. Each column is an individual animation with pose frames selected in chronological order from back to front.

3.1.2 Motion Sequence Generation. We use HumanML3D [39], a 3D human motion dataset collected from real-world motion capture in the form of 3D joint positions as an example of motion sequence generation from existing motion capture datasets. Motions with high translational movement, high lower body movement, and broken animations (e.g., jittering, limb snapping, unrealistic rotations) were filtered out from HumanML3D. Individual motions were then randomly sampled and evaluated for their uniqueness and range of motion compared to previously collected data until 100 unique motions were collected.

(a) *WheelPose* Animation Keypoints (b) COCO Keypoints

Figure 5: Keypoint mappings used in WheelPose. Figure 5(a) WheelPose 23-keypoint animation format. Used as the input format of motion sequences before pose modification. Figure 5(b) COCO 17-keypoint annotation scheme. Used as the final output format of WheelPose annotations.

Additionally, we leverage Text2Motion [40], a motion generation model that uses textual descriptions to generate motions, as a fully generative alternative source of human motion sequences. Text2Motion is only one example of human motion generation through textual descriptions [8, 112, 113] and can be easily substituted in our data generation pipeline. Evaluators from the research team assigned textual descriptions to each of the 100 selected HumanML3D motions before inputting the descriptions into Text2Motion and generating 3 potential motions for each description. The most realistic motion was selected and evaluated on whether the motion would be possible for the evaluator to perform and the absence of any noise or artifacts from the generation process that may lead to unrealistic limb movements or positions. Our Text2Motion generation process results in a new dataset of 100 human motions that directly mirror the actions of the sampled HumanML3D motions and provide a direct comparison between synthesized and real data. Our goal in enabling the use of generative motion models like Text2Motion is to investigate the feasibility of using generative deep learning models to further simplify the data collection process for synthetic data generation and therefore improve the efficiency and overall accessibility, especially in the context of users with disabilities and assistive technologies.

In total, this process yielded 200 motion sequences from both HumanML3D and Text2Motion (i.e., each yielded 100 sequences). On average, HumanML3D motions had 60.26 (SD=42.05) unique frames per animation and Text2Motion has 146.32 (SD=50.84) frames per animation. Since Text2Motion has no set animation length parameter, we choose to take the full animation output for each motion, leading to the discrepancy in average motion lengths, to directly compare data sourced from motion capture and deep learning models. Text2Motion outputs tend to extend and slow down the described action, leading to a longer but not necessarily more diverse animation compared to HumanML3D.

3.1.3 Pose Modification and Conversion. Both HumanML3D and Text2Motion represent motions through the 3D position of joints. All motions are converted into the corresponding 23-keypoint skeletal model used by *WheelPose* through a direct mapping of corresponding joints or the positional average between the surrounding joints. As is common in biomechanical analysis [10, 85], a 5Hz low pass filter is then applied to the data to handle high-frequency noise generated from the motion capture or data generation process. We convert all motions from a 3D position to a joint rotation representation. In the process of this conversion, internal and external rotations – rotations around the axis parallel to the bone not able to be described in joint position notation – of the arms are affixed to the rotation of the parent bone.

We modified the resulting pose sequences from the two generation methods by affixing the model's legs onto the wheelchair model with an additional rotational noise applied independently on the flexion/extension, abduction/adduction, and internal/external rotation on each of the lower body bones (i.e. three joints in total) to simulate regular lower body movements when in a wheelchair. The following steps document the procedure for generating rotational noise on one joint. This procedure was motivated by the need for smooth interpolated noise and inspired by the Poisson process.

Given an array of joint angles F_{orig} expressed in degrees with length n total frames in the animation, the noise is generated by first sampling a set of frames indices S.

$$S = \left\{ x_0, x_1, ..., x_n \;\middle|\; \begin{array}{l} x_0 = 0, \\ x_i = x_{i-1} + k_i, \\ k_i \sim \mathcal{N}(\frac{n}{4}, (\frac{n}{32})^2), \\ x_i < n, \\ i = 1, 2, .., n \end{array} \right\} \tag{1}$$

A new array of joint angle noise values N is then constructed. Let $f(i)$ be a function that generates the i-th joint angle of the animation. Given $U \sim \mathcal{U}(-10, 10)$, representing a random angular noise added to frames in S,

$$f(i) = \begin{cases} U & i \in S \\ \text{NaN} & else \end{cases} \tag{2}$$

$$N = [f(i) \mid 0 \leq i < n] \tag{3}$$

Linear interpolation is then applied to fill in all NaN values in N. The new array of joint angles F_{new} is thus the element-wise sum of F_{orig} and N. Our algorithm is motivated by the need for a simple and efficient noise algorithm that does not jitter as the user iterates through frames of the animation.

$$F_{new} = [f_{orig}[i] + N[i] \mid 0 \leq i < n] \tag{4}$$

All motion sequences are applied to a Blender[1] model before being imported into Unity and converted to Unity Perception-readable *AnimationClips* files. Example outputs from our posture generation step described above can be found in Figure 4.

3.2 Generating Wheelchair User Models

In order to capture synthetic images, we must rig the postures resulting from previous sections onto Unity human models.

[1]A free and open source 3D modeling software. More information is found at https://www.blender.org/

Figure 6: Example human models used in the synthesis of wheelchair user images.

For human models, *WheelPose* enables both the default human models provided by PeopleSansPeople [28] and randomized humans leveraging the Unity SyntheticHumans [4], a Unity Perception package using domain randomization to generate unique human models from a sampling of different clothes, body types, sexes, and more. This utilization allows *WheelPose* to generate unique human models that better capture the wide variety of appearances of real people. We use 8,750 unique human instances using the default SyntheticHumans configuration limited to people over the age of 10 to better reflect the general population of people in wheelchairs [103]. Figure 6 shows 19 examples of these human models.

We also enable the human models to spawn with different objects (e.g., wheelchairs, crutches, canes, walkers, etc.) in user-defined positions when placed into the environment. For the scope of this project, we focused on wheelchair users and used a realistic wheelchair model, sourced from the Unity Asset Store, scaled by the size of the human model. Finally, the posture of each human model is randomly sampled from the *AnimationClips* generated in Section 3.1.

3.3 Generating the Simulation Environment

We address **(RQ1.2)** by developing a highly configurable simulation environment. We use the PeopleSansPeople (PSP) [28] base template and its related extension, PSP-HDRI [27], as our baseline data generator built in the Unity[2] game engine through the High Definition Render Pipeline (HDRP). PSP is a parametric human image generator that contains a fully developed simulation environment including rigged human models, parameterized lighting and camera systems, occluders, synthetic RGB image outputs, and ground truth annotations. PSP is built on the idea of *domain randomization* [92] where different aspects of a simulation environment are independently randomized to diversify the generated synthetic data, exposing models to a wider array of different environments during training and improving testing accuracy [94, 99]. All domain randomization is implemented through the "randomizer" paradigm designed in the Unity Perception package [15], a Unity toolkit to generate synthetic data for computer vision training. Within each scene, individual randomizers are assigned to a specific parameter (i.e. lighting, occluder positions, human poses, etc.) and independently sample parameter values from a uniform distribution. PSP was then updated to Unity 2021.3 and Unity Perception 1.0.0[3] which enabled more annotations, more extendable randomizers, and flexibility with other Unity packages. We then added a new background image parameter and its related randomizer for the sampling of user-defined images to be used as a backdrop in each scene. We use this parameter to enable three different sets of background images: PSP default textures, 100 background images randomly sampled from the BG-20k background image dataset [58–60], and 100 generative images from Unity SynthHomes [97], a dataset generator for photorealistic home interiors. Table 1 outlines the statistical distributions of the environment parameters used.

3.3.1 Camera Configuration and Keypoint Annotations. A main camera in the Unity scene is used as the primary capture source of all images and annotations. The Unity Perception *Perception Camera* is used to simulate realistic camera features including focal length and field of view (FOV) and to capture all annotations. The position, rotation, focal length, and FOV of the main camera are set through a series of randomizers with default parameters found in Table 1. The main camera captures RGB, depth, surface normal, and instance segmentation images in 1280×720 for each frame of captured data. Default Unity Perception annotation labelers are placed on the main camera to capture 2D/3D bounding boxes for each human model, object counts, rendered object metadata, semantic segmentation, 2D/3D keypoint locations in COCO format, and percent of human model occluded. Out-of-view and fully occluded human instances are automatically ignored in annotation capture, recording only data on human instances within direct view of the main camera.

3.4 Assembling *WheelPose* Datasets

Overall, our pipeline yields 70,000 images for each generated dataset. All data was generated in a Unity 2021.3 project configured with parameters preset to the values listed in Table 1. We ran our data synthesis pipeline on a PC with a 4.2GHz 6-Core/12-Thread AMD Ryzen R5 3600, NVIDIA GTX 1070 8GB VRAM, and 32 GB 3600MHz DDR4 memory for an average generation time of $\approx$ 1 hour and 45 minutes for 10,000 images – which translates to 12 hours and 15 minutes for each dataset. This time includes all steps of the generation process including motion generation, parameter randomization, data capture, label creation, and writing to disk. Examples of generated synthetic images are shown in Figure 7. We open-source our data synthesis pipeline including pose modification and the full configurable Unity 2021.3 project for data generation in https://github.com/hilab-open-source/wheelpose.

4 EVALUATION OF *WHEELPOSE*

To answer **(RQ2)** on the benefits and drawbacks of synthetic data, we evaluate the *WheelPose* pipeline and generated data through three specific methods: 1) human evaluation on realism, 2) statistical analysis of innate dataset characteristics, and 3) evaluation of our generated datasets' effects on AI model performance. We document the results of these methods in the following sections.

4.1 Human Evaluation

We involved real wheelchair users in the loop to evaluate the realism of our synthetic data. In our study, "realism" manifests as ***ease*** and ***frequency***. We sent out online surveys and strictly verified users' eligibility and authenticity manually to prevent scammers. We recruited 13 daily wheelchair users (5 F, 8 M), with ages ranging from 26 to 56 (M=32, SD=9.6), as shown in Table 1. A key limitation of

[2]More information found at https://unity.com/
[3]First official release of Unity Perception

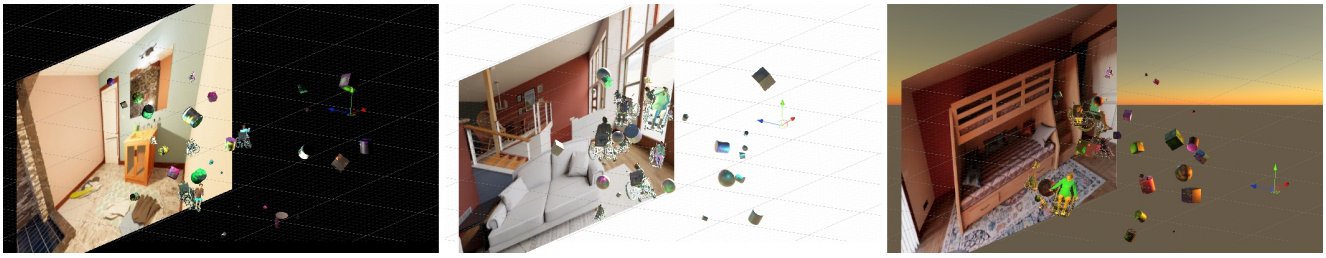


Figure 7: Examples of a scene being generated. Notice the random placement of wheelchair users, different occluder objects, different lighting conditions, and various SynthHome backgrounds. The red, green, and blue arrows represent the camera coordinates which we used to insert randomization in forms of Cartesian translations and Euler rotations.

this research is the limited diversity among the wheelchair user participants. All participants were individuals with spinal cord injuries (SCI), a specific condition that has distinct movement patterns. This represents only a small subsection of the broader wheelchair user population. Despite our efforts to diversify the participant pool by including participants with different levels of SCI, we acknowledge that our participant population is not sufficiently representative to draw statistical insights for wheelchair users with different conditions or bodies than those in this study (e.g., muscular dystrophy, amputations, dwarfism, spinal deformities).

4.1.1 Procedure. Participation was conducted entirely online, allowing users to contribute at their convenience. Users first answered a questionnaire consisting of demographic information and mobility capability before then evaluating two groups of synthetic motions - HumanML3D motions and Text2Motion motions, using a browser-based user interface (Figure 8). The motions were presented as animation GIFs of a human skeleton performing a certain movement. We chose to use skeletons rather than a more photorealistic model as an embodiment technique facilitating users to think of these skeletons as tracked motion of their bodies. Related work [34] has shown that a dummy avatar provides stronger senses of embodiment comparable to non-personalized realistic avatars. Our uses of the skeletal model aim to shift the focus towards motion, steering attention away from superficial cosmetic details. Users observed one animation clip at a time, simultaneously from four perspectives (45-degree oblique top view, top view, side view, and front view). To navigate through motion clips, users click the previous/next buttons or press the arrow keys. Buttons were designed for participants to select scores for two Likert-Scale questions.

For each motion, users answered three questions:

- **Q1:** "How difficult is it for you to do this motion?" - by rating from 1 (Cannot perform the sequence at all) to 7 (Without any difficulty).
- **Q2:** "How often do you do this motion?" - by rating from 1 (Never) to 7 (Everyday).
- **Q3:** "Have you seen or do you know of other wheelchair users who perform this motion?" (Yes/No).

We used all 100 motions in both HumanML3D and Text2Motion converted motion sets. After finishing the last animation GIF, users proceeded to the other motion group. The question set is consistent for both groups. The order of groups was random. Three users evaluated HumanML3D first, while the others saw Text2Motion first.

After all animations were scored, we followed up with participants via email to better understand the following:

(1) What are the criteria used in your evaluation of a motion's difficulty?
(2) What determines the "frequency" of performing a motion?
(3) What motion do most wheelchair users often perform and what motion do you frequently perform, that did not show up in our dataset?

Participants were paid $20 per hour as compensation for their time. As the study did not enforce a time limit and was purely online, users could take a brief break whenever they wanted as long as the questionnaire remained open. Excluding breaks, the study took 1.78 hours on average. The study was evaluated and approved by the Institutional Review Board (IRB) at UCLA.

4.1.2 Data Analysis. User responses were Likert-Scale scores (1-7 for Q1 and Q2) and binary responses (yes/no for Q3). We first visualized the distribution of data across users (Figure 9). Afterward, we separately ranked the motions based on average ease and frequency scores. We also calculated the correlation between difficulty and frequency. From follow-up emails, we collected their comments and performed a thematic analysis of their perception of metrics, and the validity of our dataset. The initial codes were the summary of their rating reasons, which were later merged and discussed. In the end, we ranked Text2Motion motions by the total frequency and difficulty score and identified the bottom 10% for later use in the model performance evaluation.

4.1.3 Results and Findings. User perception of the generated dataset (i.e., perceived realism) is a strong indicator of the efficacy of our data generation pipeline in that efficient data generation should only yield data that wheelchair users perceive as being realistic. Investigating deeper causes for the perceived realism of our generated data also inevitably led to insights about wheelchair users. We summarize the findings of our user evaluation in the rest of this section.

Participants found our synthetic motion sequences realistic. HumanML3D motions received an average ease score of 4.742 (SD=0.943) and an average frequency score of 4.079 (SD=1.085) across the dataset. Text2Motion motions performed comparably,

Table 1: Demographics of participants (P1-P13) in human evaluation.

ID	Age	Gender	Occupation	SCI Level	Exercise Routines	Full Mobility of Arms, Shoulders, and Hands
P1	56	M	Professor	T-12/L-1	No	Yes
P2	29	F	Home-maker	T-3	Weight lifting, Strolling, and Stretches	Yes
P3	40	M	Entrepreneur	T-12	Swimming, Cycling, and Gym	Yes
P4	40	F	Self-employed	C-5	No	No (DASH Score 79.2/100 [50])
P5	36	M	Customer Relations	C-5	Stretching and Strength training	No
P6	26	M	Software Developer	Lumbar spinal stenosis	Wheelchair walking	Yes
P7	51	M	Proctor/Graphic Designer	T-12; L1	Gym workouts periodically, Wheelchair Basketball, Wheelchair Tennis, and Pushing long distances	Yes
P8	26	M	Librarian	C7	No	No (DASH Score 40.0/100 [50])
P9	27	F	Remote Computer Programmer	L5	Arm training using a band	Yes
P10	27	F	Teacher	Lumbar SCI	Aquatic therapy	Yes
P11	32	F	Receptionist	Thoracic SCI	Aerobic exercise	Yes
P12	34	M	Marketing Manager	Sacral SCI	No	Yes
P13	29	M	Freelancer	Lumbar SCI	Water exercise	Yes

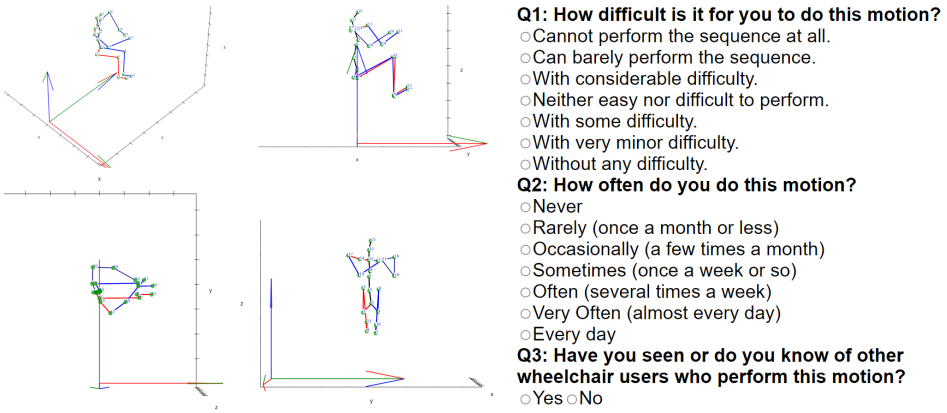


Figure 8: Screenshot of the human evaluation interface. Note that the four subplots on the left are supposed to show an animation of a human skeleton performing a motion sequence in loops from different perspectives (45-degree oblique top view, top view, side view, and front view). Participants were asked to observe an animation and give Likert-Scale scores and a binary response before moving on to evaluation of the next motion sequence.

receiving an average ease score of 4.977 (SD=0.915) and an average frequency score of 4.507 (SD=0.889). Regarding Q3, for each motion we presented, our participants had seen or knew of other wheelchair users who performed that motion. Specifically, as Figure 9(g)(h) shows, most participants have seen most motions performed by other wheelchair users.

Regarding the representativeness of datasets, most participants said they had seen all of the common motions (e.g., "rolling forward", "driving", "writing", "drinking", "cooking") they knew about in our datasets. We suspect two factors that account for the outcomes observed in these participants: 1) lack of contextual cues made it challenging for them to recall specific motions, and 2) our embodiment technique facilitated their use of imagination that bridged the gaps between the motions we demonstrated and those they executed in daily tasks. Nevertheless, some other participants commented on popular but missing motions, including "wearing some lower body clothes" (P10), "clapping hands" (P11), "typing on a keyboard" (P13). The rest of the users were unsure about their recollection of the datasets and used general phrasings e.g., "seen almost all". Our datasets were not comprehensive enough to cover all common motions. We believe user feedback is the key to improving data completeness and representativeness.

Ease (Q1) and frequency (Q2) of performing a motion vary across participants. From Figure 9(a) to Figure 9(d), we observe diversity of perception on ease and frequency across individuals. The horizontal axis is participant ID, the vertical axis is the percentage of motions in the whole group. The sequential color palette depicts the ease of doing a motion from 1 to 7, with 7 being very easy. For example, P4 and P5, who had a higher injury position, rated more motions into the difficult pool. The takeaway is that the ease and frequency of performing a motion are highly personal. A realistic motion dataset should include motions across the entire spectrum while eliminating motions that no wheelchair users will ever perceive as being easy/frequent to perform. However, the result of this user evaluation would change as the size of the participant group

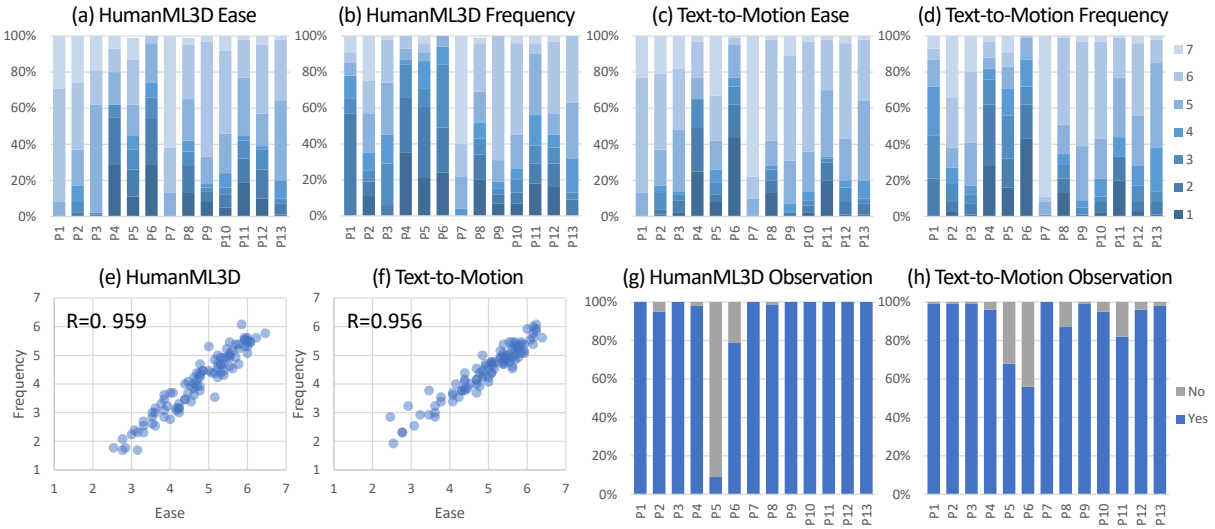


Figure 9: From left to right, Figure (a)(b)(c)(d) are 100% stacked bar charts showing the motion distribution in HumanML3D or Text2Motion, with (a) and (b) depicting the perceived ease and frequency of HumanML3D, while (c) and (d) depicting Text2Motion. In (a) and (c), a score of 1 denotes *Cannot perform the sequence at all* while a score of 7 denotes *Can perform without difficulty*. In (b) and (d), 1 means *Never*, and 7 means *Everyday*. The two scatter plots, Figure (e) and (f), demonstrate the strong correlation between ease and frequency in both motion groups. Figures (g) and (h) depict stacked bar graphs showing whether participants have seen or known of a wheelchair user who has performed this motion.

grows, which lowers the likelihood of unrealistic motions. Nonetheless, we argue that a realistic dataset should include motions that most wheelchair users would perceive as easy and frequent, to ensure the data quality and avoid introducing new biases, while addressing existing issues in the inclusiveness of data collection. Our human evaluation serves as a reference for conducting user assessments of data quality in motion synthesis.

There was a strong correlation between ease and frequency. The last two scatter plots (Figure 9(e)(f)) showcase a strong positive correlation between ease and frequency. In other words, less difficulty was associated with a higher frequency of usage. The Pearson correlation coefficients [84] are respectively 0.956 and 0.959 for Text2Motion and HumanML3D. This result was expected for motions that are difficult to perform for wheelchair users being less frequently performed because they are often circumvented by alternative motions, which was confirmed by comments from participants later on.

Factors on ease: range of motion, pain, balance, and tiredness. Participants evaluated how easy a motion was with several factors. 7 out of 13 people highlighted their range of motion using keywords "*range of motion*", "*being paralyzed*", "*based on my abilities*", "*considering the angles*." Six participants mentioned that pain affected their decisions. Regarding tiredness, P2 explained "*Since I have no core muscles, being paralyzed from the nipple line down, a lot I can do, but can't do for long*". Besides, P2, P4, and P5 emphasized balance. For example, P2 commented "*My left side is a little higher on my injury, so I struggle a lot with that side, or even staying upright when both hands are in use.*" Their perception of ease was reflected in Likert-scale scores in Figure 9 (a)(c).

Factors on frequency: utility and ease. How frequently one would make a motion depended on both utility and ease. This explained why the correlation coefficient was strong but not definite (e.g., R=0.99). An easy motion did not necessarily lead to frequent usage. Participants determined frequency mainly based on their routine. P1 commented, "*Cooking, cleaning, driving, shopping... based on where/when/how I might use the motion, I decided how often I might actually use the motion.*" Along with P1, six more users recalled their daily routine when rating for the frequency. P3 and P8 also referred to exercises/therapies to define the frequency of each motion.

On the other hand, when difficulty and utility had conflicts, participants were experts in circumventing them with alternative motions if possible. P4 explained, "*I'm limited in my arm functions with putting hands or arms over my head and behind my back... so I use straps on shoulders and high back on my wheelchair.*" P5 also commented on how he used elbows and hands to support himself without abdominal muscles when needed, "*As a tetraplegia, I don't really have abs. So, I'm kinda 'crawling' to get back up. Propping myself up with elbows on knees and such.*"

Sense of embodiment. Results show that our uses of the skeletal model successfully facilitated the sense of embodiment – participants could think of the human model as their own body, and imagine themselves performing those motions when evaluating the ease and frequency. P1 commented, "*I mimicked the motions and tried to decide when/how I would use the motion*". Similarly, another comment said, "*I picture myself doing certain tasks throughout the day, or motions I use for exercise.*" On this note of sense of embodiment facilitation, we also chose not to attach text descriptions to

animation clips, but encouraged users' imagination by allowing them to interpret the visuals on their own.

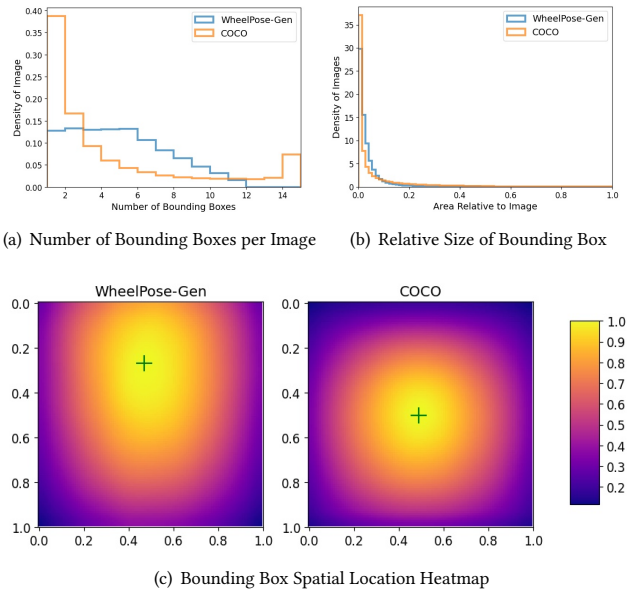
We expanded our study and recruited two new amputee participants (U1 and U2, not from P1-P13). They (1F, 1M) both had an artificial leg and used wheelchairs every day. We asked open-ended questions about their expectation of pose estimation, including but not limited to an amputated skeleton, the integration of artificial limbs, and the desired features/interfaces for pose estimation. To increase the sense of embodiment, U1 and U2 both proposed using an amputated model that accurately represented their body. For example, U1 commented, "To be accurate I think the skeleton should be amputated to maintain accurate pose estimations." Similarly, U2 said the model "should be able to be customized to meet different users with different amputations." However, expectations changed when artificial limbs came in. U2 talked about our research's reflection of his artificial limb, and said "The skeleton should adapt to work together with the artificial part that is added." Meanwhile, U1 insisted the integration of artificial limbs depended on users, saying "I think should allow users to access settings and have a choice of their own." As for the feature/interface, both U1 and U2 mentioned a settings panel to annotate where a person was amputated, and U1 further suggested an option to turn on/off the integration of artificial limbs.

4.2 Statistical Analyses

We performed a set of statistical analyses to examine how the diversity and size of our dataset compare to the full COCO 2017 persons dataset (training and validation) [64], selected as our benchmark for its ubiquity in other 2D human pose estimation related research [21, 88, 110]. Greater diversity and size in training datasets has been shown to improve machine learning performance [37] and is a common solution to prevent overfitting. We consider the following categories in our analyses: high-level dataset features, bounding box location, size, and number in generated images, keypoint number and occlusion per image and instance, diversity of human poses, and camera placement. For all subsequent dataset statistics, we used the *WheelPose*-Gen dataset, which was generated with no real-world data using SyntheticHumans models, Text2Motion animations, and SynthHome background images. The other datasets from *WheelPose* shared similar statistics and were skipped in this validation to avoid redundancy.

4.2.1 High-Level Dataset Features. In total, our dataset has 70,000 images with bounding boxes and keypoint annotations. We chose to generate 70,000 images to mimic the size of the COCO dataset. For reference, a recent effort in synthetic data has proven that a PSP dataset of as little as 49,000 images was found to have meaningful pose estimation improvements in Detectron2 fine-tuning [28].

Our dataset contains 296,508 human instances, of which 271,803 instances have annotated keypoint labeling. Note that not all human instances have annotated keypoints because some human instances had no joints within the camera's view despite still being visible. In comparison, COCO has 66,808 images with 273,469 human instances, of which 156,165 have annotated keypoints. This difference in human instances and annotated keypoints are both due to out-of-view keypoints like in our dataset, and a lack of keypoint labels due to human labeling errors.

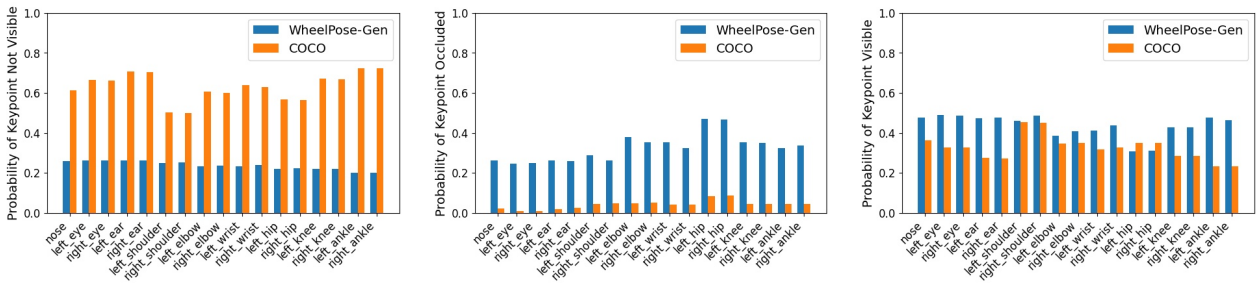


(a) Number of Bounding Boxes per Image (b) Relative Size of Bounding Box

(c) Bounding Box Spatial Location Heatmap

Figure 10: Bounding Box Statistics. All COCO statistics are computed for images that contain at least one person instance in COCO. Figure 10(a) Number of bounding boxes per image. Images with no human bounding boxes are not counted as COCO is not a purely human dataset. Figure 10(b) Relative area of each bounding box compared to the image area. Relative area is computed through $\sqrt{\text{bounding box area}/\text{image area}}$. Figure 10(c) Heatmap of bounding box location scaled by image size. The color of each pixel maps to the likelihood of that corresponding coordinate in an image being bounded by bounding boxes. The peak location of the heatmap is marked with a green +.

4.2.2 Bounding Boxes. We analyze the bounding box annotations by generating a set of statistics comparing *WheelPose*-Gen against COCO (Figure 10) to evaluate the diversity of the frequency, placement, and size of human instances. We find that *WheelPose*-Gen contains a more even distribution of the number of bounding boxes, or human instances, per image compared to COCO, indicating a greater diversity of images featuring wheelchair users in differently sized groups (Figure 10(a)). The COCO dataset does however have a higher concentration of images with large amounts of human instances within them, most likely due to the number of images depicting crowds of people. We also calculated the area of the bounding box relative to the overall image size to analyze the size of individual human instances (Figure 10(b)). Here, we observe that our bounding boxes have slightly more evenly distributed sizes in relation to the image size when compared to COCO. Note that our dataset consists of images of a uniform size (1280×720), and COCO contains a wide variety of different image sizes up to a maximum size of 640×640. Thus, images with the same relative size in relation to the image dimensions are still a higher definition in *WheelPose*-Gen comparatively.



(a) Not Visible (b) Occluded (c) Visible

Figure 11: Probability of Occlusion Labelings per Keypoint. Figure 11(a) Probability of a keypoint being labeled as "not visible." Figure 11(b) Probability of a keypoint being labeled as "occluded." Figure 11(c) Probability of a keypoint being labeled as "visible." Labeling definitions are taken directly from the COCO occlusion and keypoint labeling standard. [64].

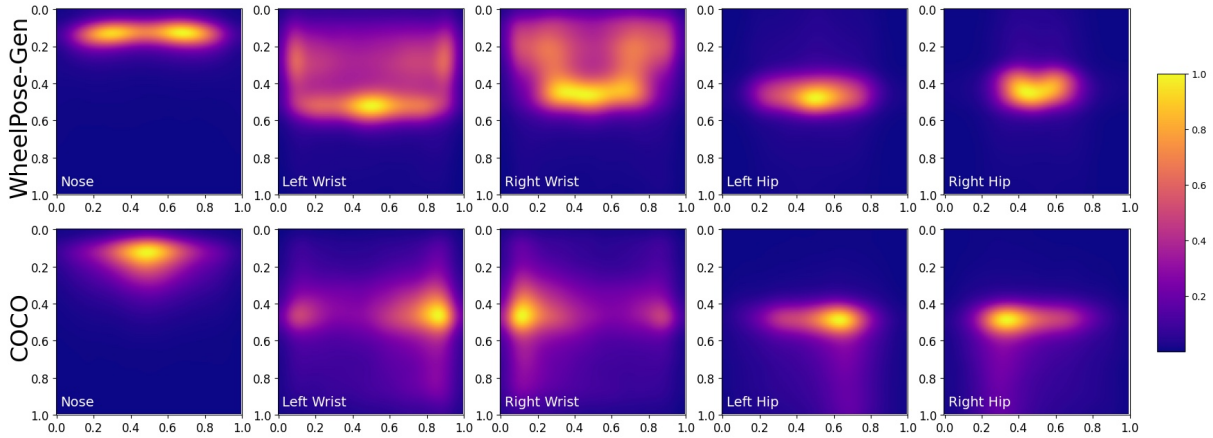


Figure 12: Heatmap of Five Keypoints Location. Top row: *WheelPose*-Gen. Bottom row: COCO. All keypoints locations in the heatmap are computed by $(\frac{\text{x−bounding box top corner x}}{\text{bounding box width}}, \frac{\text{y−bounding box top corner y}}{\text{bounding box height}})$. The heatmap is normalized according to the size of the dataset.

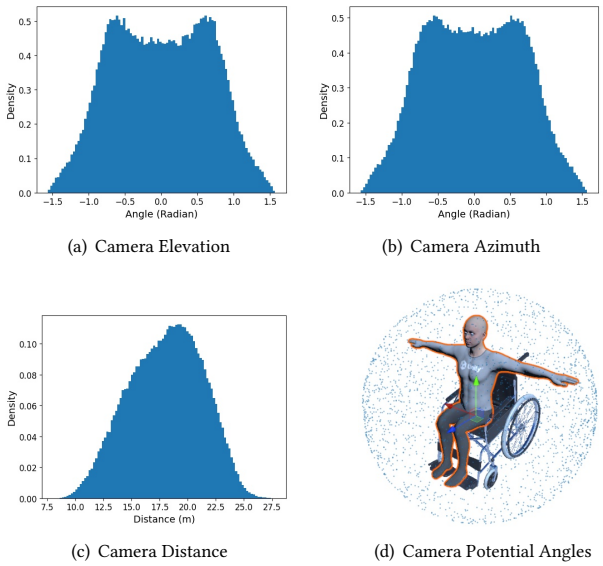
We then analyze the spatial distribution of bounding boxes by plotting a heatmap of bounding box locations for both *WheelPose*-Gen and COCO (Figure 10(c)). We note that the location of bounding boxes is a direct product of the human generation, occluder, and camera randomizers and acts as a quantification of their effects. For both datasets, we overlay the bounding boxes with their location scaled by the overall size of the image. For the COCO dataset, we observe a majority of bounding boxes are centered in the middle of the image. We also observe that *WheelPose*-Gen has a wider bounding box distribution with more spreading into the top edge of the image compared to COCO, indicating our randomization parameters create a more even spread of human instances across the image with more examples of humans at the edge of the camera. The center of the distribution of bounding boxes in *WheelPose*-Gen is also slightly higher in the image than that of COCO.

4.2.3 Keypoints. We first measure the probability of a keypoint to be one of the three predefined COCO occlusion levels (not visible, occluded, visible) in *WheelPose*-Gen and COCO as further quantification of the effects of the randomizers listed previously. In the context of *WheelPose*, not visible is when a keypoint is not in the image and has no prediction, occluded is when a keypoint is in the image but not visible (e.g., behind an object), and visible is when a keypoint is seen in an image. Here we see that *WheelPose*-Gen displays a significantly smaller probability of having nonvisible keypoints and a more uniform distribution of keypoints compared to COCO (Figure 11(a)). We also notice that *WheelPose*-Gen has a far higher probability for a keypoint to be labeled as occluded compared to COCO, especially in the hips (Figure 11(b)). This can be explained by the different methods both datasets used in keypoint annotation. While *WheelPose*-Gen uses a self-occlusion labeler defined in PSP which computes the distance between each keypoint and the closest visible part of the object within a threshold to determine occlusion labeling [28], COCO is a fully human-labeled dataset. Within human annotators, there are variations between how an annotator might define and classify as occluded and not visible. *WheelPose* does not suffer from the same issue as the labeler has information on the full 3D scene and can precisely identify every keypoint location over a human annotator who only has access

to a single 2D view with no additional context. This phenomenon can be further seen in the probability of keypoint visibility, where *WheelPose*-Gen displayed a higher probability for all keypoints except the two keypoints on hips (Figure 11(c)) where many hip keypoints are labeled as occluded due to the self-occlusion of the wheelchair.

We evaluate the diversity of our poses by creating a heatmap of keypoint annotations locations scaled by the corresponding bounding box in *WheelPose*-Gen and COCO (Figure 12). *WheelPose*-Gen displays a wider distribution of potential keypoint locations in upper body keypoints compared to COCO. Lower body keypoints display a smaller distribution due to the limitations of postures in a wheelchair. We see for asymmetrical keypoints (left/right wrist), *WheelPose*-Gen is far more evenly distributed across the X axis compared to COCO, a clear indication of a more even distribution of front, side, and back-facing human instances. We also notice a smaller Y-axis distribution in *WheelPose*-Gen due to the presence of the wheelchair limiting potential movements up and down.

4.2.4 Camera. Finally, we quantify the variations in our camera placement and rotation. Recent studies have shown the critical impact of diverse camera angles on model performance in 3D human recovery problems [18, 68, 75]. We visualize our diversity of camera angles and distances and observe a wide distribution of potential elevations, azimuths, and distances (Figure 13). We then sample a set of camera locations relative to individual human instances and visualize their angle around the instance to observe full coverage of camera angles all around instances (Figure 13(d)). All visualizations indicate a loosely followed Gaussian distribution with a wide variety of different camera angles. We do not compare these statistics with COCO since it does not include camera configuration information. Our camera configuration parameters can be found in Table 1.



(a) Camera Elevation (b) Camera Azimuth

(c) Camera Distance (d) Camera Potential Angles

Figure 13: Distribution of Potential Camera Angles and Distances. Figure 13(a) Distribution of elevation angle (up-down, positive indicating a camera above the nose and looking down). Figure 13(b) Distribution of azimuth angle (left-right, positive indicating a camera is to the right looking left) distribution. Figure 13(c) Distribution of camera distance to a human instance. Figure 13(d) Visualization of potential camera angles. Computed by sampling camera location relative to human instance every 100 human instances and visualizing the corresponding unit vector.

4.3 Model Performance Evaluation

4.3.1 Testing Dataset. There currently does not exist an image dataset focused primarily on wheelchair users. For the testing of our system, we create a new dataset of 2,464 images of wheelchair users collected from 84 *YouTube* videos in a similar data collection process as other computer vision works [7, 104]. A set of 16 action classes consisting of common daily tasks wheelchair and able-bodied users both perform and unique wheelchair sports were selected (e.g., talking, basketball, rugby, dancing, etc.). Action classes are listed in Table 2. Videos are collected through keyword searches revolving around each of the action classes. Annotators iterate through 500 equally spaced frames (minimum one-second intervals) from each video and identify frames with poses and settings sufficiently different from the previously collected images. Annotators ensure that there is a wheelchair user within view for every collected frame. Crowd worker involvement is then utilized to annotate bounding boxes and keypoint locations on collected images. Researchers manually validated results from this process for accuracy. Examples of this dataset can be found in our open-source repository at https://github.com/hilab-open-source/wheelpose.

4.3.2 Training Strategy. Similar to training outlined in PSP-HDRI [27], all of our models in this evaluation utilized ResNet-50 [42] plus Feature Pyramid Network (FPN) [63] backbones. Additionally, these models were fine-tuned using the starting weights and framework of the Detectron2 ImageNet Keypoint R-CNN R50-FPN variant [41].

We create each model in the same way: fine-tuning with the relevant dataset on the backbone described previously. We opted for a simple training approach, setting the initial learning rate to 0.000025 for 20 epochs, and then lowering the learning rate by a factor of 10 for an additional 10 epochs. For the first 1000 iterations we also conduct a linear warm-up of the learning rate to its starting value, slowly increasing the learning rate to its starting value. The momentum was set to 0.9 and the weight decay was set to 0.0001. All training runs were completed using a 4.2GHz 16-core/32-thread AMD Ryzen Threadripper PRO 3955WX CPU, 2 NVIDIA RTX A5500 24GB VRAM, and 256 GB 3200MHz DDR4 memory with a mini-batch size of 13 images per GPU, where each image was normalized using the mean pixel value and standard deviation of the ImageNet base model. For each model, we checkpointed the model weights during every epoch and selected the epoch with the best-performing keypoint AP to report in evaluation. This evaluation scheme was consistent across baseline datasets and our synthetic datasets.

We note that *WheelPose* fine-tuned models and individual human evaluators may not agree with each other on where a keypoint is due to the innate difference between a modeled person and a real person. Changes in the way *WheelPose* defines keypoints can

Table 2: Ablation trials with configurations of each dataset and results in terms of the mean Average Precision (mAP) in the bounding box (BB) and keypoint (KP) detection tasks.

Dataset	Human Model	Background	Animation	BB mAP	KP mAP
WheelPose-base	PSP default	PSP default	HumanML3D	68.68	65.18
WheelPose-SH	SyntheticHumans	PSP default	HumanML3D	68.53	**68.04**
WheelPose-t2m	PSP default	PSP default	Text2Motion	68.95	64.83
WheelPose-t2mR10	PSP default	PSP default	Text2Motion random 10% removed	68.97	65.19
WheelPose-t2mHE10	PSP default	PSP default	Text2Motion HE 10% removed	69.36	65.53
WheelPose-SB	PSP default	SynthHomes	HumanML3D	**69.75**	66.60
WheelPose-RB	PSP default	BG-20K	HumanML3D	64.44	63.89
WheelPose-Gen	SyntheticHumans	SynthHomes	Text2Motion	**69.71**	67.53
WheelPose-Opt	SyntheticHumans	SynthHomes	Text2Motion HE 10% removed	69.58	**67.96**
ImageNet (baseline)	N.A.	N.A.	N.A.	**35.19**	**63.11**
PSP (baseline)	PSP Default	PSP Default	PSP Default Able-Bodied	31.03	53.26

alter where a keypoint is predicted and the metrics computed in the coming sections as seen in Appendix A.5. We attempted to minimize these differences as much as possible through realistic keypoint definitions in Unity.

4.3.3 Ablation Testing Strategy. We address **(RQ2.1)** through ablation testing[4] on a set of selected domain randomization parameters, including animations, backgrounds, and human models, to better understand the performance impacts of select data generation parameters.

Configuration. Regarding the *human model*, we analyze the effects on the model performance of using: 1) PSP default human models (PSP Default), and 2) SyntheticHuman human models (SyntheticHumans). As to the *background*, we compare between 1) PSP default texture backgrounds (PSP Default), 2) SynthHomes background images (SynthHomes), and 3) BG-20K real background images (BG-20K). Finally, regarding the *motion sequence*, we compare between 1) HumanML3D animations (HumanML3D), 2) Text2Motion animations (Text2Motion), 3) Text2Motion with 10% of animations randomly removed (Text2Motion random 10% removed), and 4) Text2Motion animations with the bottom 10% of animations in total ease and frequency from human evaluations (HE) (Text2Motion HE 10% removed). Table 2 shows this list of datasets and their configurations.

We use combinations of these parameters to assemble a set of datasets of 70,000 images each. Each dataset consisted of $\approx 65,000$ images with at least one human instance and was used to fine-tune the base ImageNet model using the strategy listed in Section 4.3.2. We also include the original PSP dataset with $\approx 65,000$ images with at least one human instance generated from the provided Unity environment to test the efficacy of fine-tuning with synthetic able-bodied user data [28]. For these tests, all models and our baseline tests, ImageNet and PSP, were tested on our real wheelchair data testing set using the industry standard metric for detection and estimation accuracy, COCO mean Average Precision [64].

Results. Results in terms of bounding box (BB) and keypoint (KP) mean APs (mAP) across ablation trials are shown in Table 2.

[4]Ablation testing involves the removal of certain components to understand the contribution of the component to the overall performance of an AI system

Regarding person detection (BB) performance, all datasets demonstrated significant performance boosts of varying degrees when compared to ImageNet and PSP. Notably, our best-performing ablation test led to a 98.21% improvement in BB mAP (*WheelPose*-SB) and a 7.81% improvement in keypoint mAP (*WheelPose*-SH) over the best baseline dataset (ImageNet). This indicates the efficacy of our synthetic data and the large headroom for improvement in detecting poses of wheelchair users with industry-standard deep learning models.

Poor performance from BG-20K may be explained by the detail of background images. PSP default and SynthHomes data tend to feature a set of simple or smooth textures while real-world images are often more detailed and consist of more texture. These results may signal a preference for other background characteristics over pure realism.

When examining keypoint performance, the most significant improvement was the inclusion of Unity SyntheticHumans models. This makes intuitive sense, as the diverse and more representative human models more closely match up with the humans found in the real world rather than the generalized Unity humanoid models. The variations between models also help combat overfitting issues by introducing many different definitions of what is a "human" to the model.

Finally, we examine the performance tradeoffs in our different animation sets. We found motion generation outputs with random filtering (*WheelPose*-t2mR10) performed comparably to motion capture data (*WheelPose*), indicating similar motion quality between the two data sources. We also found that randomly filtered Text2Motion animations (*WheelPose*-t2mR10) performed better than the full motion set (*WheelPose*-t2m). This may imply that the number of animations and poses is not directly correlated with model performance. It is important to note these results may change depending on what animations have been filtered. This idea is further shown in the HE model (*WheelPose*-t2mHE10), which showed improvement in both BB and KP mAP over the randomly removed 10% model. We see that the removal of specific animations that are not perceived as "realistic" can improve the model performance of generated data.

Table 3: Bounding box AP performance comparison between base models and *WheelPose*-Opt. We list the mean of our seeded testing ± the maximum absolute deviation from the mean.

Dataset	BB mAP	BB $AP^{IoU=.50}$	BB $AP^{IoU=.75}$	BB AP^{small}	BB AP^{medium}	BB AP^{large}
WheelPose-Opt	69.46 ± 0.22	90.77 ± 0.20	82.75 ± 0.61	3.24 ± 0.76	59.82 ± 0.85	69.91 ± 0.21
ImageNet (baseline)	35.19	71.49	28.50	0.00	3.15	36.12
PSP (baseline)	31.03	63.64	25.08	0.00	6.27	32.35

Table 4: Keypoint AP performance comparison between baseline and *WheelPose*-Opt. We list the mean of our seeded testing ± the maximum absolute deviation from the mean. We do not include AP^{small} since the human is too small to accurately assign keypoints.

Dataset	KP mAP	KP $AP^{OKS=.50}$	KP $AP^{OKS=.75}$	KP AP^{medium}	KP AP^{large}
WheelPose-Opt	67.93 ± 0.02	87.61 ± 0.19	74.48 ± 0.25	35.48 ± 0.40	68.99 ± 0.06
ImageNet (baseline)	63.11	77.43	67.20	6.22	64.96
PSP (baseline)	53.26	68.07	57.36	10.15	56.12

From our ablation testing results, we assembled two new datasets: *WheelPose*-Gen, a dataset created from fully generative parameters using SyntheticHuman models, SynthHomes backgrounds, and all 100 Text2Motion animations, and *WheelPose*-Opt, a dataset created from the best performing parameters from ablation testing which include SyntheticHuman models, SynthHomes backgrounds, and Text2Motion HE 10% removed animations. Both datasets performed comparably to the best performing ablation test in BB and KP mAP (*WheelPose*-SH) with *WheelPose*-Opt. We note that *WheelPose*-Opt outperforms *WheelPose*-Gen in KP mAP which follows our findings in the initial ablation test. Our findings indicate that different combinations of domain randomization parameters can produce better AI models than the best perform parameters individually.

4.3.4 WheelPose-Opt Model Performance Deep Dive. We further evaluate **(RQ2.2)** by conducting an in-depth quantitative and qualitative analysis of the changes in performance in Detectron2 when fine-tuned with *WheelPose*-Opt, the best performing dataset from ablation testing using synthetic data and simple human evaluations.
Configuration. We trained and evaluated the results for *WheelPose*-Opt with the same strategy described in Section 4.3.2 three separate times using different model seeds (42, 4242, 424242). We then computed the mean and maximum absolute deviation of a set of evaluation metrics, including BB AP, KP AP, and individual keypoint metrics, across the three trials. We compute the same metrics on ImageNet and PSP for use as our baseline.
Results. We first quantify our overall bounding box and keypoint performance with AP and its related breakdowns to build an overarching view of our dataset's performance across different scenarios. AP at different IoU and OKS[5] thresholds measure the prediction accuracy at varying degrees of recall (Higher indicates a stricter ground truth definition). Additionally, the overall AP score is split into small, medium, and large based on the detection segment area to quantify the performance at different camera distances and human instance sizes. More information is found in the COCO documentation [64].

Table 3 lists the BB AP performance for *WheelPose*-Opt and the baselines. Our dataset displays over a 98% improvement over both baseline scores in all subcategories of AP. Furthermore, we notice a major drop in performance in the baseline models as the IoU threshold becomes stricter. In contrast, our *WheelPose* models maintain a high level of accuracy across a much wider range of IoU thresholds. These results indicate *WheelPose*-Opt can not only identify a wheelchair user but is capable of drawing an accurate bounding box around the entire human instance compared to the baseline models which are only capable of low-fidelity wheelchair user detections.

As shown in Table 4, pose estimation improved by up to 7.64% in KP mAP in *WheelPose*-Opt when compared to ImageNet and up to 27.54% when compared to PSP. Furthermore, we see a similar or greater magnitude of improvement across all AP metrics, indicating the use of *WheelPose*-Opt improves pose estimation in all scenarios. We note drastic improvements in AP on medium-sized human instances, where ImageNet had noticeably poor performance (6.22). Our system, thus, not only improves but even enables existing models to estimate the postures of wheelchair chairs at further distances.

Finally, we compute a set of per-keypoint metrics shown in Table 5 to analyze the differences in specific keypoint predictions between *WheelPose*-Opt and the baseline. We only showcase percent change with respect to ImageNet as the PSP fine-tuned model performs drastically worse in nearly all BB and KP metrics. Upon examining the percentage of detected joint (PDJ) [83] at a 5% threshold, a measure of a model's ability to identify a joint, we notice a 76.14% improvement in ankle detection, attributed to more information on foot placement in wheelchairs, and a 18.96% decrease in detected hips, attributed to the wheelchair obstructing most of the lower torso and hip area. We then compute the per joint position error (PJPE), a simple accuracy metric measuring the Euclidean distance error in detected joints, and found that as long as a joint is

[5]IoU and OKS perform the same fundamental purpose for bounding boxes and keypoints respectively

Table 5: Keypoint performance breakdown of our primary datasets. We list the mean of our seeded testing ± the maximum absolute deviation from the mean. Within the parentheses on the right is the percent difference from the base model, ImageNet. Percentage of Detected Joints (PDJ) @ 5 describes the percentage of correctly predicted joints within a 5% bounding box diagonal radius [83]. Per Joint Position Error (PJPE) describes the mean Euclidean distance for each keypoint from the ground truth [115]. Object Keypoint Similarity Score (OKS) as described by COCO [64] and used in Chen *et al.* [23] is the mean precision per keypoint evaluated at both standard loose and strict similarity thresholds of 0.5 and 0.75 respectively. Superiority directions are noted as + and - next to each metric.

Keypoint	PDJ@5 (+)	PJPE (-)	OKS50 (+)	OKS75 (+)
nose	0.91 ± 0.02(−2.15%)	8.73 ± 4.06(−77.38%)	0.90 ± 0.02(−3.23%)	0.89 ± 0.03(−1.83%)
eyes	0.99 ± 0.00(+5.70%)	6.30 ± 0.64(−77.6%)	0.96 ± 0.01(+3.76%)	0.95 ± 0.01(+4.19%)
ears	0.92 ± 0.01(+5.56%)	10.90 ± 0.03(−61.60%)	0.93 ± 0.00(+5.90%)	0.83 ± 0.01(+0.81%)
shoulders	0.92 ± 0.02(+6.74%)	10.98 ± 0.26(−63.89%)	1.00 ± 0.01(+4.74%)	0.97 ± 0.01(+6.38%)
elbows	0.86 ± 0.01(+2.78%)	14.54 ± 0.23(−69.38%)	0.97 ± 0.00(+5.80%)	0.92 ± 0.00(+6.55%)
wrists	0.85 ± 0.01(−0.19%)	16.22 ± 1.65(−76.22%)	0.92 ± 0.02(+6.13%)	0.88 ± 0.02(+1.73%)
hips	0.65 ± 0.01(−18.96%)	22.96 ± 0.61(−51.24%)	0.97 ± 0.00(−2.02%)	0.89 ± 0.01(−6.63%)
knees	0.89 ± 0.02(+6.63%)	14.99 ± 0.97(−78.85%)	0.97 ± 0.01(+5.25%)	0.94 ± 0.01(+7.01%)
ankles	0.78 ± 0.01(+76.14%)	18.16 ± 0.95(−69.04%)	0.94 ± 0.00(+87.67%)	0.90 ± 0.01(+91.49%)

detected, *WheelPose*-Opt predicts keypoints over 51.24% (hips) more accurately. Finally, we compute the meaned per-keypoint precision values at OKS thresholds of 0.5 and 0.75 as another measure of individual keypoint prediction accuracy. We see slight improvements across most joints. Similar to PDJ, we can see similar trends in the ankles and hips, improving a significant amount or slightly worse respectively.

4.3.5 Key Prediction Changes. We conduct a visual analysis of the changes in predicted keypoints between ImageNet and *WheelPose*-Opt to analyze the information transfer between our synthetic data and the real world across different scenarios to identify specific situations where we perform better or worse. We ignore the PSP dataset as it demonstrated noticeably worse performance compared to ImageNet in both bounding box and pose estimation (Tables 3 and 4). Examples of different trends were plotted with the predictions from both *WheelPose*-Opt, in green, and ImageNet, in red, overlaid on top.

Improvements in wheelchair user detection. As shown in the BB AP improvements between ImageNet and *WheelPose*-Opt in Table 3, we notice major improvements in wheelchair user detection. Figure 14 shows some examples of these improvements in a variety of different environments. Figure 14(a), 14(d), and 14(e) shows two examples of proper wheelchair user detection through *WheelPose*-Opt. Figure 14(b), and 14(c) shows two examples of wheelchair users being detected in low visibility settings due to both extremely bright and dark lighting conditions. Of particular note is Figure 14(d), where even the background poster of a wheelchair image has been detected, which human annotators had missed. We also notice that even if ImageNet had detected a wheelchair user, its bounding box prediction still was not accurate. Figure 14(f) to14(j) shows some examples of this, where ImageNet tends to cut off portions of the wheelchair user's full body while *WheelPose*-Opt captures a more accurate and representative bounding box.

Similar performance in front-facing scenarios. In front-facing scenarios, ImageNet and other pose estimation models often perform very well on wheelchair users. This is because, at this angle, the user can simply be interpreted to be sitting, with all limbs in full view of the camera. Thus in practice, the front-facing wheelchair user is very similar to a front-facing able-bodied user that is sitting. We find that fine-tuning with *WheelPose*-Opt performs comparably with the base ImageNet models in front-facing scenarios. Thus, our system maintains crucial information learned from the initial training of ImageNet that has proven to work well on wheelchair users already. Examples of this phenomenon are shown in Figure 15.

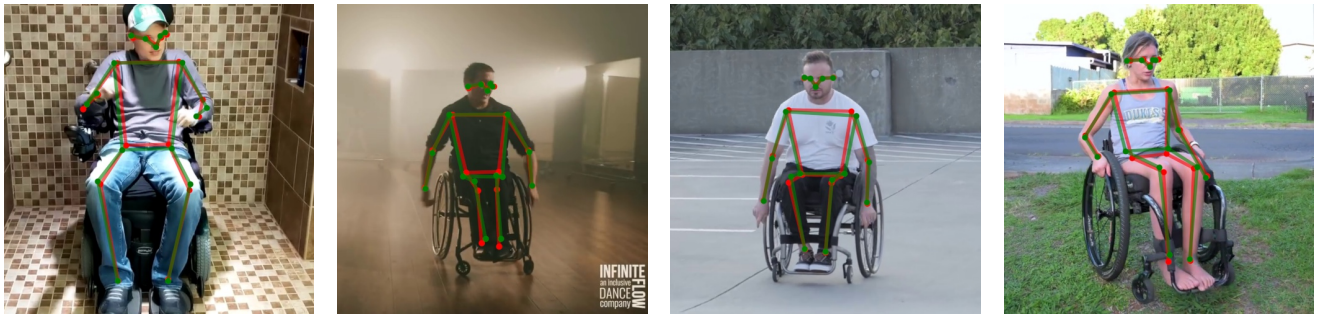
Improvements in keypoint estimation in wheelchair self-occlusion scenarios. While existing pose estimation models may work well when the wheelchair user is facing directly forward, they often break down when the user is turned away and the wheelchair begins obstructing the view of the full human body. We find that the additional synthetic data from *WheelPose*-Opt helps Detectron2 discern between what is a part of the wheelchair and what is a part of the user's body for a more accurate prediction. Figure 16 illustrates a few examples of such poor performance in ImageNet and improved predictions enabled through *WheelPose*-Opt. In situations where the legs are fully occluded by the wheelchair like in Figure 16(a), and 16(d), our system generates more reasonable predictions compared to those of ImageNet, which placed the legs onto the wheels of the wheelchair or even the elbow. Figure 16(b) shows an example of how *WheelPose*-Opt can improve the detection of self occluded keypoints like the user's left knee and ankle. Figure 16(b), and 16(c) shows examples of where ImageNet has mistakenly classified parts of the wheelchair as a keypoint.

Overfitting on wheelchairs. Upon examining the predictions made by ImageNet and ImageNet fine-tuned with *WheelPose*-Opt, we notice that both systems perform poorly on users with no lower limbs. As shown in Figure 17(a), and 17(b) While ImageNet tends to classify the wheelchair as the missing legs, we notice that our system instead "fills in the blanks" and predicts legs in reference to the wheelchair where they might usually be for a wheelchair user. We further notice that our system tends to have more false positives in detecting what can be defined as a wheelchair. As seen in Figure

(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

Figure 14: Examples of wheelchair user detection improvements with *WheelPose*-Opt over ImageNet. Red represents ImageNet predictions while green represents *WheelPose*-Opt predictions. Figure 14(a) to 14(e) all show wheelchair users in different scenarios who were completely undetected by ImageNet but detected with *WheelPose*-Opt fine-tuning. Figure 14(f) to 14(j) all show wheelchair users in different scenarios who were detected by ImageNet, but had poor bounding box predictions which were improved in *WheelPose*-Opt fine-tuning.



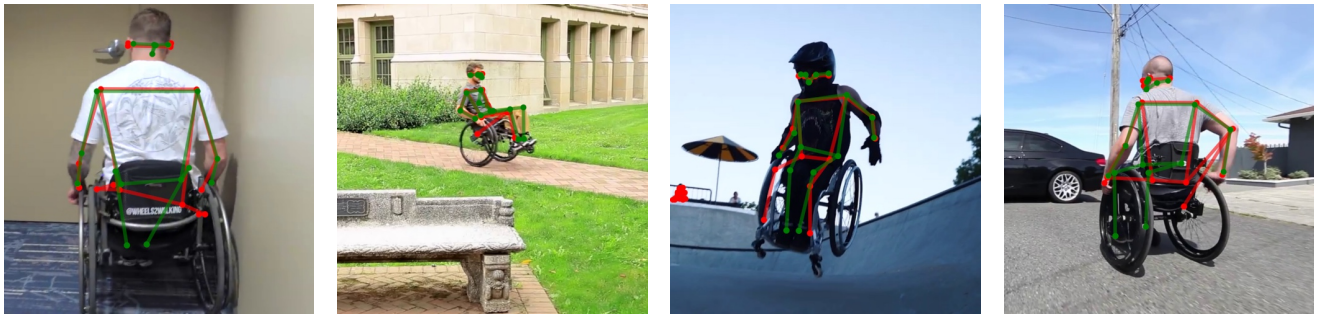
(a) (b) (c) (d)

Figure 15: Examples of similar performance between *WheelPose*-Opt and ImageNet in front-facing wheelchair users. Red represents ImageNet predictions while green represents *WheelPose*-Opt predictions. Figure 15(a) to 15(d) all depict examples of front-facing wheelchair users in a variety of different settings. The *WheelPose*-Opt fine-tuned model and ImageNet both performed similarly, generating relatively matching keypoint predictions.

17(c), objects that resemble a wheelchair, like a grocery cart, may affect the keypoint estimation of a wheelchair user. In other cases like the one shown in Figure 17(d), we see that our system can even detect empty wheelchairs and fill them in with humans when there are not any. While this shows great promise in the information transfer between digitally modeled mobility assistive devices and real-world data, we find these tendencies can obfuscate the real postures of wheelchair users.

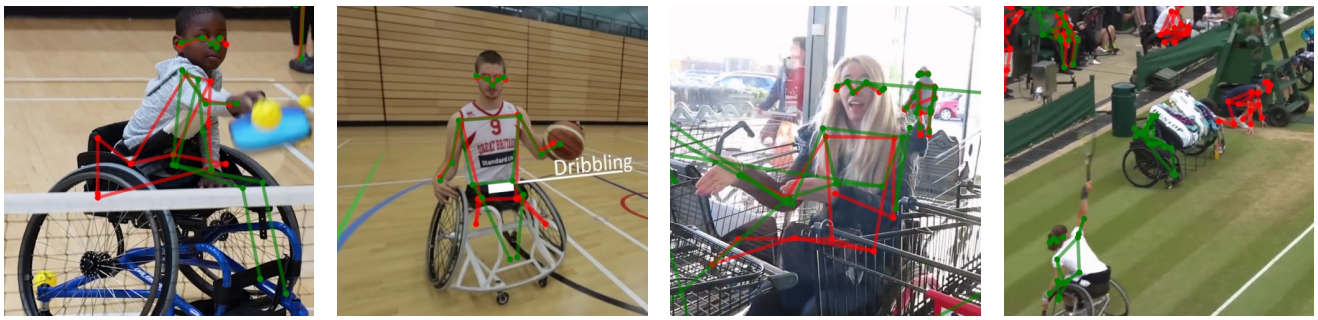
5 DISCUSSION AND FUTURE WORK

Generalizable knowledge and transferability to HCI research. Data generation approaches have been widely adopted in projects in HCI for problems involving thermal imaging [47], IMU [51, 80], stroke gesture [56], and RF data [5, 17]. Given the popularity of data generation in HCI, we believe our techniques could be easily transferable to related and future works in accessibility, motion generation, and pose estimation. Furthermore, the modularization in our pipeline could improve transferability by facilitating segmented changes – a flexible way for data synthesis to experiment with different components. Finally, methods shown in our statistical analysis and model performance evaluation could be highly reusable in future work that adopts our technique. That being said, our pipeline provides the baseline framework for futures efforts in research that require different humanoid models, motion synthesis techniques for upper body and lower body, environmental factors, and VR toolchains.



(a) (b) (c) (d)

Figure 16: Examples of improvements in keypoint estimation in *WheelPose*-Opt over ImageNet in scenarios where the wheelchair occludes part of the user's body. Red represents ImageNet predictions while green represents *WheelPose*-Opt predictions. Figure 16(a) to 16(d) all display improvements from *WheelPose*-Opt on keypoint predictions, specifically in the lower body.



(a) (b) (c) (d)

Figure 17: Examples of overfitting on wheelchairs from *WheelPose*-Opt. Red represents ImageNet predictions while green represents *WheelPose*-Opt predictions. Figure 17(a) and 17(b) Examples of poor predictions on wheelchair users without lower limbs. Figure 17(c) An example of overfitting onto any wheelchair resembling object, like a grocery cart. Figure 17(d) An example of overfitting where even an empty wheelchair is detected to have a user.

Configurability to cater needs of developers and end users. We believe the results shown in Table 2, Appendix A.1, and Appendix A.5 demonstrate a clear need for developers to have an interactive tool to modify training data, where slight changes in the data modeling can have major effects on AI performance. Currently, there are few tools to do so, with many developers choosing to leverage static motion datasets which may not perfectly fit their needs. This has been one of the dominant reasons for the rise of inequitable AI models. For this reason, *WheelPose* is a pipeline specifically designed to enable a high degree of configurability. This allows for the creation of personalized synthetic datasets that cater to developers' needs, thereby increasing the likelihood of catering to the needs of end users. This would lead to more effective and inclusive AI models, which can be tuned to the individual needs of the user instead of a one-size-fits-all solution for better performance in real-world applications.

Improvements on the realism of generated data. We hope the use of Unity enables future research into this idea and enables developers to build a synthetic data generator that extends beyond our simple simulation environment using 3D modeled scenes and rooms instead of flat background images. Recent achievements in Neural Radiance Fields (NeRF) could be leveraged to synthesize photorealistic background images that adapt in response to changes in the virtual camera's perspective. Future work in realistic 3D environment modeling will enable research in surface semantics for more realistic configurations – having wheelchair users positioned at ground surfaces. Moreover, physics could be incorporated to simulate the locomotion of wheelchair users which would enable the modeling of more realistic motions and collisions with other models compared to our simple animations.

Efficacy of human evaluation. We found that motions from motion generation models and motion capture were largely perceived similarly by participants in the human evaluation (Section 4.1). We further found that these two types of motions after being filtered for human-perceived realism performed comparably in model performance evaluation (Table 2). However, whether due to current generative AI performance or the lack of training data for disability-related movement, our human evaluations indicate that generative AI models are still not able to represent users with disabilities accurately without external human feedback. This is shown in the improved performance with the addition of human evaluations (Table 2). Due to this, we believe that human evaluators are still vital to ensure that generated data is representative. We recognize that our method for assessing motions, although not exhaustive or entirely free of bias, contributes as an initial

stride towards creating datasets that are more inclusive. Future research should extend upon this work to integrate a more involved human-centered system which will, instead, allow evaluators to meaningfully guide the generation of data, including motions, simulation scenarios, and user modeling, through iterative feedback (e.g., guided prompting) to create a more representative dataset. We also found little literature on motion synthesis for people with disabilities and recognized this vacuum as both a challenge and an opportunity for generative AI.

Comprehensive model performance evaluation. Our model testing did not involve a serious grid search for data generation hyperparameters, model seeds, initializations, or model configurations. We also used the same training strategy for all tested datasets. We held all these values constant to focus on the quality and impact of the synthetic dataset on model performance. Even with this naïve approach to training, *WheelPose* still yielded promising results in the improvement of pose estimation models through synthetic data on wheelchair users. We found noticeable improvements in both person detection and pose estimation problems. This indicates that the specific synthetic modeling of users in wheelchairs in *WheelPose* can make existing computer vision models more equitable by improving performance on wheelchairs. We believe our findings pave the way for future works in synthetic data on humans with other mobility-assistive technologies to improve pose estimation equitability.

Diverse participant groups. A key limitation in this work is that we only analyze the digitalized representation of users with all four limbs and feet fixed on the foot rest, which does not express the full range of wheelchair users with different bodies such as those with amputations, dwarfism, spinal deformities, and other conditions. Thus the findings may not be reflective of the wheelchair population at large. We hope our data generation pipeline provides a framework and will enable future developers to leverage 3D modeling and Unity tools to create a more diverse body of wheelchair user models. For instance, developers can easily add new bones to existing models to more realistically represent spinal deformities. We hope that these tools will also enable future works analyzing different disabilities and mobility assistive devices which were not addressed in our current research.

Improve inclusiveness of AI for more recognition modalities. Furthermore, *WheelPose* enables more work beyond 2D bounding boxes and pose estimation. Annotations on depth, surface normals, object segmentation, occlusion, 3D bounding boxes, and 3D keypoints are fully implemented in our current data synthesis pipeline but still unexplored. These annotations can be even more difficult and costly to collect compared to RGB images and 2D pose annotations, often requiring the use of specialized equipment and data collection processes. Synthetic data has no such problem, where any desired annotation and labeling are all equally accessible to collect. Thus, we believe that *WheelPose* can be adapted to potentially address problems in wheelchair pedestrian detection with object segmentation and occlusion annotations [26], 3D pose estimation using 3D bounding box and pose annotations, and robotics detection of people and mobility aids through depth data [102] among other accessibility-related problems cheaply and efficiently.

Pitfalls of exclusion in data generation vs. data collection. *WheelPose* is a pipeline for both AI developers and wheelchair users to circumvent existing inaccessible data collection methods and meaningfully improve the training process of AI models by generating data for wheelchair users. However, we are cautious that by circumventing existing inaccessible data collection methods with our tool, we could run the potential risk of furthering exclusion, which echoes long-standing debates within the accessibility community. Our paper is based on the assumption that making AI equitable requires pursuing multiple approaches together – effective approaches to improving representations of training data from people with disabilities leveraging both data collection and data generation. We advocate that new tools for data generation and the existing data collection methods are not mutually exclusive. Their synergy could lead to a more practical approach to resolving accessibility challenges than what could be offered by either of the two approaches alone. We believe that the following characteristics are vital in future works to avoid pitfalls of exclusion: 1) representative and diverse participant groups, for which we have conducted studies around spinal injuries of various levels and recommend future work to consider participants from wider backgrounds; 2) realistic generated data, for which we invented several data generation techniques optimizing data realism; and 3) effective tools for human evaluation, for which we adopted embodiment in our human evaluation interface allowing participants to seamlessly transfer the presented motion sequences to their own bodies for a more intuitive evaluation.

6 CONCLUSION

We introduce *WheelPose*, an extension of the highly-parameterized synthetic data generator PeopleSansPeople, for wheelchair users with the possibility of use in other mobility-assistive technologies to improve the performance of common pose estimation algorithms in the traditionally underrepresented group of wheelchair users. *WheelPose* includes a full end-to-end pipeline to convert existing motion capture and motion generation model outputs into wheelchair user animations for use in a complete Unity Simulation scene **(RQ1.1)** containing a range of 3D human models from Unity SyntheticHumans in wheelchairs, backgrounds, occluders, and unique lighting conditions. We provide full control over all related parameters, including keypoint labeling schema, for computer vision tasks **(RQ1.2)**. We tested our pipeline using two different motion sequence sources: motion capture data from HumanML3D and motion generation outputs from Text2Motion. These motions underwent a set of human evaluations. We then analyzed the impacts of different domain randomization parameters and motions on model performance, finding an "optimal" combination of parameters and comparable performance between our motion sources **(RQ2.1)**. Finally, we tested the model performance of the dataset generated through *WheelPose* with no real-world data using the optimal parameters found previously on a dataset of real wheelchair users to find noticeable improvements in model performance when compared against existing pose estimation models **(RQ2.2)**. We expect *WheelPose* to enable a new range of research in using synthetic data to model users with disabilities in improving the equity of AI.

REFERENCES

[1] 2019. The Kinesthetic Index: Video Games and the Body of Motion Capture – InVisible Culture. https://ivc.lib.rochester.edu/the-kinesthetic-index-video-games-and-the-body-of-motion-capture/.

[2] 2022. Waypoint - The Official Waymo Blog: Utilizing Key Point and Pose Estimation for the Task of Autonomous Driving. https://waymo.com/blog/2022/02/utilizing-key-point-and-pose-estimation.html.

[3] 2023. DeepMotion - AI Motion Capture & Body Tracking. https://www.deepmotion.com/.

[4] 2023. SyntheticHumans Package (Unity Computer Vision). Unity Technologies.

[5] Karan Ahuja, Yue Jiang, Mayank Goel, and Chris Harrison. 2021. Vid2Doppler: Synthesizing Doppler Radar Data from Videos for Training Privacy-Preserving Activity Recognition. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (<conf-loc>, <city>Yokohama</city>, <country>Japan</country>, </conf-loc>) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 292, 10 pages. https://doi.org/10.1145/3411764.3445138

[6] Jason W. Anderson, Marcin Ziolkowski, Ken Kennedy, and Amy W. Apon. 2022. Synthetic Image Data for Deep Learning. arXiv:2212.06232 [cs.CV]

[7] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2014. 2D Human Pose Estimation: New Benchmark and State of the Art Analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[8] Samaneh Azadi, Akbar Shah, Thomas Hayes, Devi Parikh, and Sonal Gupta. 2023. Make-An-Animation: Large-Scale Text-conditional 3D Human Motion Generation. arXiv:2305.09662 [cs.CV]

[9] Slawomir Bak, Peter Carr, and Jean-Francois Lalonde. 2018. Domain Adaptation through Synthesis for Unsupervised Person Re-identification. arXiv:1804.10094 [cs.CV]

[10] Roger Bartlett. 2014. *Introduction to Sports Biomechanics: Analysing Human Movement Patterns.* Routledge.

[11] Begoña Bartolomé Villar, Irene Real Benlloch, Ana De la Hoz Calvo, and Gleyvis Coro-Montanet. 2022. Perception of Realism and Acquisition of Clinical Skills in Simulated Pediatric Dentistry Scenarios. *International Journal of Environmental Research and Public Health* 19, 18 (2022), 11387.

[12] Valentin Bazarevsky, Ivan Grishchenko, Karthik Raveendran, Tyler Zhu, Fan Zhang, and Matthias Grundmann. 2020. BlazePose: On-device Real-time Body Pose tracking. arXiv:2006.10204 [cs.CV]

[13] Eduard Gabriel Bazavan, Andrei Zanfir, Mihai Zanfir, William T. Freeman, Rahul Sukthankar, and Cristian Sminchisescu. 2022. HSPACE: Synthetic Parametric Humans Animated in Complex Environments. arXiv:2112.12867 [cs.CV]

[14] Cynthia L Bennett and Daniela K Rosner. 2019. The promise of empathy: Design, disability, and knowing the" other". In *Proceedings of the 2019 CHI conference on human factors in computing systems*. 1–13.

[15] Steve Borkman, Adam Crespi, Saurav Dhakad, Sujoy Ganguly, Jonathan Hogins, You-Cyuan Jhang, Mohsen Kamalzadeh, Bowen Li, Steven Leal, Pete Parisi, Cesar Romero, Wesley Smith, Alex Thaman, Samuel Warren, and Nupur Yadav. 2021. Unity Perception: Generate Synthetic Data for Computer Vision. arXiv:2107.04259 [cs.CV]

[16] Robert Bridson. 2007. Fast Poisson Disk Sampling in Arbitrary Dimensions. In *ACM SIGGRAPH 2007 Sketches* (San Diego, California) *(SIGGRAPH '07)*. Association for Computing Machinery, New York, NY, USA, 22–es. https://doi.org/10.1145/1278780.1278807

[17] Hong Cai, Belal Korany, Chitra R. Karanam, and Yasamin Mostofi. 2020. Teaching RF to Sense without RF Training Measurements. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 4, Article 120 (dec 2020), 22 pages. https://doi.org/10.1145/3432224

[18] Zhongang Cai, Mingyuan Zhang, Jiawei Ren, Chen Wei, Daxuan Ren, Zhengyu Lin, Haiyu Zhao, Lei Yang, Chen Change Loy, and Ziwei Liu. 2022. Playing for 3D Human Recovery. arXiv:2110.07588 [cs.CV]

[19] Zhe Cao, Hang Gao, Karttikeya Mangalam, Qi-Zhi Cai, Minh Vo, and Jitendra Malik. 2020. Long-term Human Motion Prediction with Scene Context. arXiv:2007.03672 [cs.CV]

[20] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. 2019. OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. arXiv:1812.08008 [cs.CV]

[21] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. 2017. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. arXiv:1611.08050 [cs.CV]

[22] Patrick Carrington, Gierad Laput, and Jeffrey P. Bigham. 2020. SpokeSense: Developing a Real-Time Sensing Platform for Wheelchair Sports. *SIGACCESS Access. Comput.* 124, Article 2 (mar 2020), 1 pages. https://doi.org/10.1145/3386308.3386310

[23] Shuhong Chen and Matthias Zwicker. 2021. Transfer Learning for Pose Estimation of Illustrated Characters. arXiv:2108.01819 [cs.CV]

[24] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 248–255. https://doi.org/10.1109/CVPR.2009.5206848

[25] Maximilian Denninger, Martin Sundermeyer, Dominik Winkelbauer, Youssef Zidan, Dmitry Olefir, Mohamad Elbadrawy, Ahsan Lodhi, and Harinandan Katam. 2019. BlenderProc. arXiv:1911.01911 [cs.CV]

[26] Piotr Dollár, Christian Wojek, Bernt Schiele, and Pietro Perona. 2011. Pedestrian Detection: An Evaluation of the State of the Art. *IEEE transactions on pattern analysis and machine intelligence* 34 (07 2011), 743–61. https://doi.org/10.1109/TPAMI.2011.155

[27] Salehe Erfanian Ebadi, Saurav Dhakad, Sanjay Vishwakarma, Chunpu Wang, You-Cyuan Jhang, Maciek Chociej, Adam Crespi, Alex Thaman, and Sujoy Ganguly. 2022. PSP-HDRI$+$: A Synthetic Dataset Generator for Pre-Training of Human-Centric Computer Vision Models. arXiv:2207.05025 [cs]

[28] Salehe Erfanian Ebadi, You-Cyuan Jhang, Alex Zook, Saurav Dhakad, Adam Crespi, Pete Parisi, Steven Borkman, Jonathan Hogins, and Sujoy Ganguly. 2022. PeopleSansPeople: A Synthetic Data Generator for Human-Centric Computer Vision. arXiv:2112.09290 [cs]

[29] Khaled El Emam, Lucy Mosquera, and Richard Hoptroff. 2020. *Practical synthetic data generation: balancing privacy and the broad availability of data.* O'Reilly Media.

[30] Mark Everingham, Luc Gool, Christopher K. Williams, John Winn, and Andrew Zisserman. 2010. The Pascal Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vision* 88, 2 (jun 2010), 303–338. https://doi.org/10.1007/s11263-009-0275-4

[31] Matteo Fabbri, Fabio Lanzi, Simone Calderara, Andrea Palazzi, Roberto Vezzani, and Rita Cucchiara. 2018. Learning to Detect and Track Visible and Occluded Body Joints in a Virtual World. arXiv:1803.08319 [cs.CV]

[32] Luigi Ferrucci, Rachel Cooper, Michelle Shardell, Eleanor M. Simonsick, Jennifer A. Schrack, and Diana Kuh. 2016. Age-Related Change in Mobility: Perspectives From Life Course Epidemiology and Geroscience. *The Journals of Gerontology Series A: Biological Sciences and Medical Sciences* 71, 9 (Sept. 2016), 1184–1194. https://doi.org/10.1093/gerona/glw043

[33] Ellen Freiberger, Cornel Christian Sieber, and Robert Kob. 2020. Mobility in Older Community-Dwelling Persons: A Narrative Review. *Frontiers in Physiology* 11 (2020).

[34] Rebecca Fribourg, Ferran Argelaguet, Anatole Lécuyer, and Ludovic Hoyet. 2020. Avatar and sense of embodiment: Studying the relative preference between appearance, control and point of view. *IEEE transactions on visualization and computer graphics* 26, 5 (2020), 2062–2072.

[35] Adrien Gaidon, Qiao Wang, Yohann Cabon, and Eleonora Vig. 2016. Virtual worlds as proxy for multi-object tracking analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4340–4349.

[36] Elizabeth A Gardener, Felicia A Huppert, Jack M Guralnik, and David Melzer. 2006. Middle-Aged and Mobility-Limited: Prevalence of Disability and Symptom Attributions in a National Survey. *Journal of General Internal Medicine* 21, 10 (Oct. 2006), 1091–1096. https://doi.org/10.1111/j.1525-1497.2006.00564.x

[37] Zhiqiang Gong, Ping Zhong, and Weidong Hu. 2019. Diversity in Machine Learning. *IEEE Access* 7 (2019), 64323–64350. https://doi.org/10.1109/access.2019.2917620

[38] Anhong Guo, Ece Kamar, Jennifer Wortman Vaughan, Hanna Wallach, and Meredith Ringel Morris. 2020. Toward fairness in AI for people with disabilities SBG@ a research roadmap. *ACM SIGACCESS Accessibility and Computing* 125 (2020), 1–1.

[39] Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. 2022. Generating Diverse and Natural 3D Human Motions From Text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5152–5161.

[40] Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. 2022. Generating Diverse and Natural 3D Human Motions From Text. (2022).

[41] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2018. Mask R-CNN. arXiv:1703.06870 [cs.CV]

[42] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 770–778. https://doi.org/10.1109/CVPR.2016.90

[43] Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip Torr, Song Bai, and Xiaojuan Qi. 2023. Is synthetic data from generative models ready for image recognition? arXiv:2210.07574 [cs.CV]

[44] Wan He, Daniel Goodkind, and Paul Kowal. 2016. International Population Reports. (2016).

[45] Christoph Heindl, Lukas Brunner, Sebastian Zambal, and Josef Scharinger. 2020. BlendTorch: A Real-Time, Adaptive Domain Randomization Library. arXiv:2010.11696 [cs.CV]

[46] Bill Howe, Julia Stoyanovich, Haoyue Ping, Bernease Herman, and Matt Gee. 2017. Synthetic data for social good. *arXiv preprint arXiv:1710.08874* (2017).

[47] Fang Hu, Peng He, Songlin Xu, Yin Li, and Cheng Zhang. 2020. FingerTrak: Continuous 3D hand pose tracking by deep learning hand silhouettes captured by miniature thermal cameras on wrist. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 2 (2020), 1–24.

[48] Yuan-Ting Hu, Hong-Shuo Chen, Kexin Hui, Jia-Bin Huang, and Alexander G. Schwing. 2019. SAIL-VOS: Semantic Amodal Instance Level Video

Object Segmentation – A Synthetic Dataset and Baselines. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 3100–3110. https://doi.org/10.1109/CVPR.2019.00322

[49] Yuan-Ting Hu, Jiahong Wang, Raymond A. Yeh, and Alexander G. Schwing. 2021. SAIL-VOS 3D: A Synthetic Dataset and Baselines for Object Detection and 3D Mesh Reconstruction from Video Data. arXiv:2105.08612 [cs.CV]

[50] Pamela L Hudak, Peter C Amadio, Claire Bombardier, Dorcas Beaton, Donald Cole, Aileen Davis, Gillian Hawker, Jeffrey N Katz, Matti Makela, Robert G Marx, et al. 1996. Development of an upper extremity outcome measure: the DASH (disabilities of the arm, shoulder, and head). *American journal of industrial medicine* 29, 6 (1996), 602–608.

[51] Yash Jain, Hyeokhyen Kwon, and Thomas Ploetz. 2023. On the Effectiveness of Virtual IMU Data for Eating Detection with Wrist Sensors. In *Adjunct Proceedings of the 2022 ACM International Joint Conference on Pervasive and Ubiquitous Computing and the 2022 ACM International Symposium on Wearable Computers* (Cambridge, United Kingdom) *(UbiComp/ISWC '22 Adjunct)*. Association for Computing Machinery, New York, NY, USA, 50–52. https://doi.org/10.1145/3544793.3560337

[52] Ioannis Kaloskampis, David Pugh, Chaitanya Joshi, and Louisa Nolan. 2019. Data science for the public good. (2019).

[53] Reuben Kirkham and Benjamin Tannert. 2021. Using Computer Simulations to Investigate the Potential Performance of 'A to B' Routing Systems for People with Mobility Impairments. arXiv:2107.01570 [cs.HC]

[54] Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Matt Deitke, Kiana Ehsani, Daniel Gordon, Yuke Zhu, Aniruddha Kembhavi, Abhinav Gupta, and Ali Farhadi. 2022. AI2-THOR: An Interactive 3D Environment for Visual AI. arXiv:1712.05474 [cs.CV]

[55] Jesse Leaman and Hung M. La. 2017. A Comprehensive Review of Smart Wheelchairs: Past, Present and Future. arXiv:1704.04697 [cs.RO]

[56] Luis A. Leiva, Daniel Martín-Albo, and Radu-Daniel Vatavu. 2017. Synthesizing Stroke Gestures Across User Populations: A Case for Users with Visual Impairments. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 4182–4193. https://doi.org/10.1145/3025453.3025906

[57] Vincent Leroy, Philippe Weinzaepfel, Romain Brégier, Hadrien Combaluzier, and Grégory Rogez. 2020. SMPLy Benchmarking 3D Human Pose Estimation in the Wild. arXiv:2012.02743 [cs.CV]

[58] Jizhizi Li, Sihan Ma, Jing Zhang, and Dacheng Tao. 2021. Privacy-Preserving Portrait Matting. arXiv:2104.14222 [cs.CV]

[59] Jizhizi Li, Jing Zhang, Stephen J Maybank, and Dacheng Tao. 2022. Bridging composite and real: towards end-to-end deep image matting. *International Journal of Computer Vision* 130, 2 (2022), 246–266.

[60] Jizhizi Li, Jing Zhang, and Dacheng Tao. 2021. Deep Automatic Natural Image Matting. arXiv:2107.07235 [cs.CV]

[61] Zhengqin Li, Ting-Wei Yu, Shen Sang, Sarah Wang, Meng Song, Yuhan Liu, Yu-Ying Yeh, Rui Zhu, Nitesh Gundavarapu, Jia Shi, Sai Bi, Zexiang Xu, Hong-Xing Yu, Kalyan Sunkavalli, Miloš Hašan, Ravi Ramamoorthi, and Manmohan Chandraker. 2021. OpenRooms: An End-to-End Open Framework for Photorealistic Indoor Scene Datasets. arXiv:2007.12868 [cs.CV]

[62] Jacky Liang, Viktor Makoviychuk, Ankur Handa, Nuttapong Chentanez, Miles Macklin, and Dieter Fox. 2018. GPU-Accelerated Robotic Simulation for Distributed Reinforcement Learning. arXiv:1810.05762 [cs.RO]

[63] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. 2017. Feature Pyramid Networks for Object Detection. arXiv:1612.03144 [cs.CV]

[64] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. 2015. Microsoft COCO: Common Objects in Context. arXiv:1405.0312 [cs.CV]

[65] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. 2019. MediaPipe: A Framework for Building Perception Pipelines. arXiv:1906.08172 [cs.DC]

[66] Diogo Luvizon, Marc Habermann, Vladislav Golyanik, Adam Kortylewski, and Christian Theobalt. 2023. Scene-Aware 3D Multi-Human Motion Capture from a Single Camera. arXiv:2301.05175 [cs.CV]

[67] Congcong Ma, Wenfeng Li, Raffaele Gravina, and Giancarlo Fortino. 2017. Posture Detection Based on Smart Cushion for Wheelchair Users. *Sensors (Basel, Switzerland)* 17, 4 (March 2017), 719. https://doi.org/10.3390/s17040719

[68] Spandan Madan, Timothy Henry, Jamell Dozier, Helen Ho, Nishchal Bhandari, Tomotake Sasaki, Frédo Durand, Hanspeter Pfister, and Xavier Boix. 2021. When and how CNNs generalize to out-of-distribution category-viewpoint combinations. arXiv:2007.08032 [cs.CV]

[69] Timo Marcard, Roberto Henschel, Michael Black, Bodo Rosenhahn, and Gerard Pons-Moll. 2018. *Recovering Accurate 3D Human Pose in the Wild Using IMUs and a Moving Camera: 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part X*. 614–631. https://doi.org/10.1007/978-3-030-01249-6_37

[70] Nathan Morrical, Jonathan Tremblay, Yunzhi Lin, Stephen Tyree, Stan Birchfield, Valerio Pascucci, and Ingo Wald. 2021. NViSII: A Scriptable Tool for Photorealistic Image Generation. arXiv:2105.13962 [cs.CV]

[71] Amit Moryossef, Ioannis Tsochantaridis, Roee Aharoni, Sarah Ebling, and Srini Narayanan. 2020. Real-Time Sign Language Detection using Human Pose Estimation. arXiv:2008.04637 [cs.CV]

[72] Marion Mundt, Zachery Born, Molly Goldacre, and Jacqueline Alderson. 2022. Estimating Ground Reaction Forces from Two-Dimensional Pose Data: A Biomechanics-Based Comparison of AlphaPose, BlazePose, and OpenPose. *Sensors (Basel, Switzerland)* 23, 1 (Dec. 2022), 78. https://doi.org/10.3390/s23010078

[73] Temitayo Olugbade, Marta Bieńkiewicz, Giulia Barbareschi, Vincenzo D'Amato, Luca Oneto, Antonio Camurri, Catherine Holloway, Mårten Björkman, Peter Keller, Martin Clayton, Amanda Williams, Nicolas Gold, Cristina Becchio, Benoît Bardy, and Nadia Bianchi-Berthouze. 2022. Human Movement Datasets: An Interdisciplinary Scoping Review. *Comput. Surveys* 55 (05 2022). https://doi.org/10.1145/3534970

[74] Joon Sung Park, Danielle Bragg, Ece Kamar, and Meredith Ringel Morris. 2021. Designing an online infrastructure for collecting AI data from people with disabilities. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*. 52–63.

[75] Sola Park, Seungjin Yang, and Hyuk-Jae Lee. 2023. MVDet: multi-view multi-class object detection without ground plane assumption. *Pattern Analysis and Applications* 26 (06 2023). https://doi.org/10.1007/s10044-023-01168-6

[76] Priyanka Patel, Chun-Hao P. Huang, Joachim Tesch, David T. Hoffmann, Shashank Tripathi, and Michael J. Black. 2021. AGORA: Avatars in Geography Optimized for Regression Analysis. arXiv:2104.14643 [cs.CV]

[77] Leonid Pishchulin, Arjun Jain, Mykhaylo Andriluka, Thorsten Thormählen, and Bernt Schiele. 2012. Articulated people detection and pose estimation: Reshaping the future. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 3178–3185.

[78] Albert Pumarola, Jordi Sanchez, Gary Choi, Alberto Sanfeliu, and Francesc Moreno-Noguer. 2019. 3DPeople: Modeling the Geometry of Dressed Humans. In *International Conference in Computer Vision (ICCV)*.

[79] Miroslav Purkrábek and Jiří Matas. 2023. Improving 2D Human Pose Estimation across Unseen Camera Views with Synthetic Data. arXiv:2307.06737 [cs.CV]

[80] Vitor Fortes Rey, Peter Hevesi, Onorina Kovalenko, and Paul Lukowicz. 2019. Let There Be IMU Data: Generating Training Data for Wearable, Motion Sensor Based Activity Recognition from Monocular RGB Videos. In *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers* (London, United Kingdom) *(UbiComp/ISWC '19 Adjunct)*. Association for Computing Machinery, New York, NY, USA, 699–708. https://doi.org/10.1145/3341162.3345590

[81] Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M. Susskind. 2021. Hypersim: A Photorealistic Synthetic Dataset for Holistic Indoor Scene Understanding. arXiv:2011.02523 [cs.CV]

[82] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M. Lopez. 2016. The SYNTHIA Dataset: A Large Collection of Synthetic Images for Semantic Segmentation of Urban Scenes. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 3234–3243. https://doi.org/10.1109/CVPR.2016.352

[83] Ben Sapp and Ben Taskar. 2013. MODEC: Multimodal Decomposable Models for Human Pose Estimation. *Proceedings / CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 3674–3681. https://doi.org/10.1109/CVPR.2013.471

[84] Patrick Schober, Christa Boer, and Lothar A Schwarte. 2018. Correlation coefficients: appropriate use and interpretation. *Anesthesia & analgesia* 126, 5 (2018), 1763–1768.

[85] Sander Schreven, Peter J. Beek, and Jeroen B. J. Smeets. 2015. Optimising Filtering Parameters for a 3D Motion Analysis System. *Journal of Electromyography and Kinesiology* 25, 5 (Oct. 2015), 808–814. https://doi.org/10.1016/j.jelekin.2015.06.004

[86] Bokui Shen, Fei Xia, Chengshu Li, Roberto Martín-Martín, Linxi Fan, Guanzhi Wang, Claudia Pérez-D'Arpino, Shyamal Buch, Sanjana Srivastava, Lyne P. Tchapmi, Micael E. Tchapmi, Kent Vainio, Josiah Wong, Li Fei-Fei, and Silvio Savarese. 2021. iGibson 1.0: a Simulation Environment for Interactive Tasks in Large Realistic Scenes. arXiv:2012.02924 [cs.AI]

[87] Joshua Snoke, Gillian M Raab, Beata Nowok, Chris Dibben, and Aleksandra Slavkovic. 2018. General and specific utility measures for synthetic data. *Journal of the Royal Statistical Society Series A: Statistics in Society* 181, 3 (2018), 663–688.

[88] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. 2019. Deep High-Resolution Representation Learning for Human Pose Estimation. arXiv:1902.09212 [cs.CV]

[89] NICOLÁS SALAZAR SUTIL. 2015. *Motion and Representation: The Language of Human Movement*. The MIT Press. http://www.jstor.org/stable/j.ctt17kk8zx

[90] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. 2022. Human motion diffusion model. *arXiv preprint*

arXiv:2209.14916 (2022).

[91] Yonglong Tian, Lijie Fan, Phillip Isola, Huiwen Chang, and Dilip Krishnan. 2023. StableRep: Synthetic Images from Text-to-Image Models Make Strong Visual Representation Learners. arXiv:2306.00984 [cs.CV]

[92] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. 2017. Domain Randomization for Transferring Deep Neural Networks from Simulation to the Real World. arXiv:1703.06907 [cs.RO]

[93] Emanuel Todorov, Tom Erez, and Yuval Tassa. 2012. MuJoCo: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 5026–5033. https://doi.org/10.1109/IROS.2012.6386109

[94] Jonathan Tremblay, Aayush Prakash, David Acuna, Mark Brophy, Varun Jampani, Cem Anil, Thang To, Eric Cameracci, Shaad Boochoon, and Stan Birchfield. 2018. Training Deep Networks with Synthetic Data: Bridging the Reality Gap by Domain Randomization. arXiv:1804.06516 [cs.CV]

[95] Shari Trewin. 2018. AI fairness for people with disabilities: Point of view. *arXiv preprint arXiv:1811.10670* (2018).

[96] Rigas Tzikas. 2022. *How realistic is my synthetic data? A qualitative approach.* Master's thesis.

[97] Unity Technologies. 2022. Unity SynthHomes: A Synthetic Home Interior Dataset Generator. https://github.com/Unity-Technologies/SynthHomes.

[98] Skanda Upadhyaya, Shravan Bhat, Siddhanth P. Rao, V Ashwin, and Krishnan Chemmangat. 2022. A cost effective eye movement tracker based wheel chair control algorithm for people with paraplegia. arXiv:2207.10511 [cs.HC]

[99] Svetozar Zarko Valtchev and Jianhong Wu. 2021. Domain Randomization for Neural Network Classification. *Journal of Big Data* 8, 1 (July 2021), 94. https://doi.org/10.1186/s40537-021-00455-5

[100] Gül Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J. Black, Ivan Laptev, and Cordelia Schmid. 2017. Learning from Synthetic Humans. In *CVPR*.

[101] Gul Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J Black, Ivan Laptev, and Cordelia Schmid. 2017. Learning from synthetic humans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 109–117.

[102] Andres Vasquez, Marina Kollmitz, Andreas Eitel, and Wolfram Burgard. 2017. Deep Detection of People and their Mobility Aids for a Hospital Robot. arXiv:1708.00674 [cs.RO]

[103] Nicolas Vignier, Jean-François Ravaud, Myriam Winance, François-Xavier Lepoutre, and Isabelle Ville. 2008. Demographics of wheelchair users in France: Results of National community-based handicaps-incapacités-dépendance surveys. *Journal of rehabilitation medicine : official journal of the UEMS European Board of Physical and Rehabilitation Medicine* 40 (04 2008), 231–9. https://doi.org/10.2340/16501977-0159

[104] Yilin Wang, Sasi Inguva, and Balu Adsumilli. 2019. YouTube UGC Dataset for Video Compression Research. In *2019 IEEE 21st International Workshop on Multimedia Signal Processing (MMSP)*. 1–5. https://doi.org/10.1109/MMSP.2019.8901772

[105] Zhenzhen Weng, Laura Bravo-Sánchez, and Serena Yeung. 2023. Diffusion-HPC: Generating Synthetic Images with Realistic Humans. *arXiv preprint arXiv:2303.09541* (2023).

[106] Meredith Whittaker, Meryl Alper, Cynthia L Bennett, Sara Hendren, Liz Kaziunas, Mara Mills, Meredith Ringel Morris, Joy Rankin, Emily Rogers, Marcel Salas, et al. 2019. Disability, bias, and AI. *AI Now Institute* 8 (2019).

[107] Magnus Wrenninge and Jonas Unger. 2018. Synscapes: A Photorealistic Synthetic Dataset for Street Scene Parsing. arXiv:1810.08705 [cs.CV]

[108] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. 2019. Detectron2. https://github.com/facebookresearch/detectron2.

[109] Kevin Xie, Tingwu Wang, Umar Iqbal, Yunrong Guo, Sanja Fidler, and Florian Shkurti. 2022. Physics-based Human Motion Estimation and Synthesis from Videos. arXiv:2109.09913 [cs.CV]

[110] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. 2023. ViTPose++: Vision Transformer Foundation Model for Generic Body Pose Estimation. arXiv:2212.04246 [cs.CV]

[111] Zhitao Yang, Zhongang Cai, Haiyi Mei, Shuai Liu, Zhaoxi Chen, Weiye Xiao, Yukun Wei, Zhongfei Qing, Chen Wei, Bo Dai, Wayne Wu, Chen Qian, Dahua Lin, Ziwei Liu, and Lei Yang. 2023. SynBody: Synthetic Dataset with Layered Human Models for 3D Human Perception and Modeling. arXiv:2303.17368 [cs.CV]

[112] Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Shaoli Huang, Yong Zhang, Hongwei Zhao, Hongtao Lu, and Xi Shen. 2023. T2M-GPT: Generating Human Motion from Textual Descriptions with Discrete Representations. arXiv:2301.06052 [cs.CV]

[113] Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. 2022. MotionDiffuse: Text-Driven Human Motion Generation with Diffusion Model. arXiv:2208.15001 [cs.CV]

[114] Ce Zheng, Wenhan Wu, Chen Chen, Taojiannan Yang, Sijie Zhu, Ju Shen, Nasser Kehtarnavaz, and Mubarak Shah. 2023. Deep Learning-Based Human Pose Estimation: A Survey. arXiv:2012.13392 [cs.CV]

[115] Michael Zwölfer, Dieter Heinrich, Kurt Schindelwig, Bastian Wandt, Helge Rhodin, Jörg Spörri, and Werner Nachbauer. 2023. Deep Learning-Based 2D Keypoint Detection in Alpine Ski Racing – A Performance Analysis of State-of-the-Art Algorithms Applied to Regular Skiing and Injury Situations. *JSAMS Plus* 2 (2023), 100034. https://doi.org/10.1016/j.jsampl.2023.100034

A APPENDIX

A.1 Animation Clip Fix

To ensure the human model's do not overlap with the wheelchair footplates, a 35 degree rotation of the hips up toward the sky is applied. This hip rotation decreases the amount of clipping between the legs and the wheelchair. Clipping is a common occurrence in 3D modeling where when objects are within each other, only the object closer to the camera will be rendered and obscure the overlapped object. Examples of this extra clipping are shown in Figure 1.

A.2 Posture to *AnimationClip* Conversion

We convert the postures resulting from previous sections into human images using human models in Unity which take *AnimationClips* as input for pose configurations. To convert pose frames into *AnimationClips*, all motion sequences from each set, represented by a series of joint rotations, are individually imported into Blender. Each frame's joint rotations are applied to the corresponding joint in a Unity Perception human model Blender template and exported as an FBX file. Upon importing an FBX file into Unity, Unity will automatically convert all baked animations into Unity-readable *AnimationClip* files which can be used in Unity Perception. All *AnimationClips* are then set to read as Unity Humanoid animations for use in data synthesis.

A.3 *WheelPose* Randomizers

Unity Perception enables the use of the "randomizer" paradigm to enable users to configure the *domain randomization* of individual parameters [15]. *WheelPose* uses multiple Unity Perception default randomizers, PSP custom randomizers [28], and a collection of custom *WheelPose* randomizers. It is important to note that many of PSP's custom randomizers have made it into the release version of Unity Perception (1.0.0). We have chosen to maintain the original randomizers used for a direct comparison between data synthesized between PSP and *WheelPose*. Like in PSP, our randomizers are regarded as further data augmentation techniques which limits the need for data augmentations during training itself. All randomizers sampled values from a uniform distribution. Table 1 outlines the statistical distributions for our randomizer parameters. A brief description of each randomizer used in *WheelPose* is described below.

BackgroundObjectPlacementRandomizer. Randomly spawns background and occluder objects within a user-defined 3D volume. Separation distance can be set to dictate the proximity of objects from each other. Poisson-Disk sampling [16] is used to randomly place objects sourced from a set of primitive 3D game objects (cubes, cylinders, spheres, etc.) from Unity Perception in a given area.

BackgroundOccluderScaleRandomizer. Randomizes the scale of the background and occluder objects.

RotationRandomizer. Randomizes the 3D rotation of background and occluder objects.

ForegroundObjectPlacementRandomizer. Similar to *BackgroundObjectPlacementRandomizer*. Randomly spawns foreground objects selected from the default set of PSP models affixed in wheelchair models.

ForegroundScaleRandomizer. Similar to *BackgroundOccluderScaleRandomizer*. Randomizes the scale of foreground objects.

TextureRandomizer. Randomizes the texture of predefined objects provided as a JPEG or PNG. We used the set of example textures from Unity Perception which are applied to the background and occluder objects as well as to the background wall when no specific background is set.

HueOffsetRandomizer. Randomizes the hue offset applied to textures on the object. Applied to background and occluder objects as well as to the background wall when no specific background is set.

SpriteRandomizer. Randomizes the background wall. Used as an alternative to the *TextureRandomizer* when images should not be stretched to fill a canvas.

HumanGenerationRandomizer. Randomizes the age, sex, ethnicity, height, weight, and clothing of spawned human assets. Humans are spawned in batches called pools which are periodically regenerated through the simulation process. All humans are spawned within a predefined base which contains the wheelchair model used. All textures and models used are sourced directly from SyntheticHumans.

NonTagAnimationRandomizer. Randomizes the pose applied to a character. The pose is a randomly selected frame from a randomly selected *AnimationClip* taken from a universal pool of *AnimationClips*. Provides a custom alternative to the Unity Perception AnimationRandomizer for randomizing animations taken from a single pool.

TransformPlacementRandomizer. Randomizes the position, rotation, and size of generated SyntheticHumans. Rotations around the *X,Z*-axis are limited to better represent real world data where users are rarely seen in such orientations.

SunAngleRandomizer. Randomizes a directional light's intensity, elevation, and orientation to mimic the lighting effects of the Sun.

LightRandomizer. Randomizes a light's intensity and color (RGBA). Also enables the randomization of a light's on/off state.

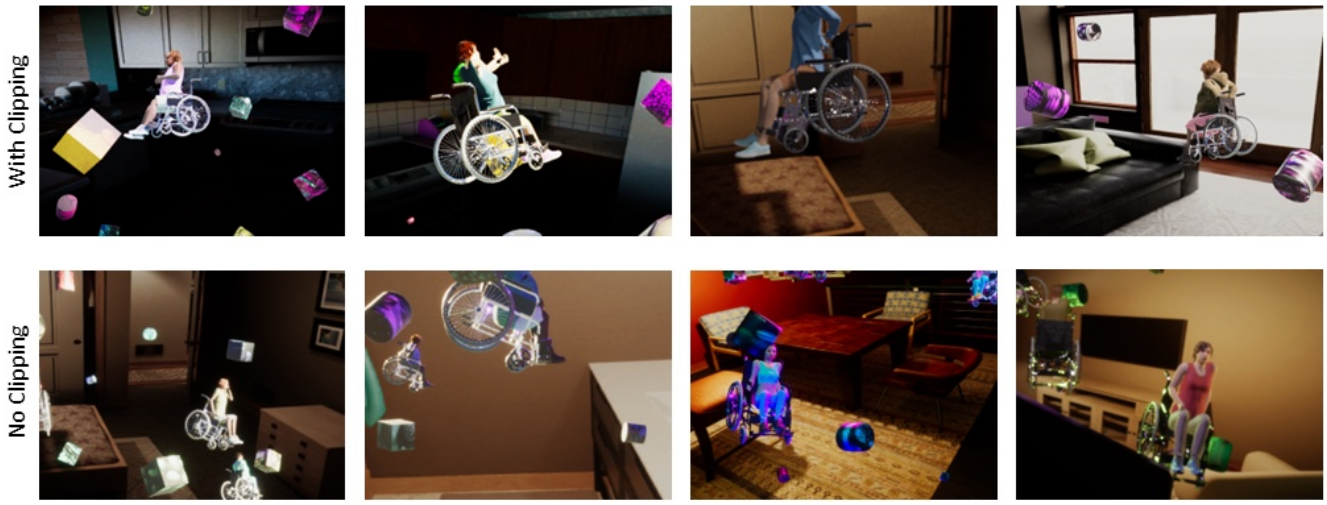
LightPositionRotationRandomizer. Randomizes a light's global position and rotation in the scene.

CameraRandomizer. Randomizes the extrinsic parameters of a camera including its global position and rotation. Enables the randomization of intrinsic camera parameters including field of view and focal length to better mimic a physical camera. Adds camera bloom and lens blur around objects that are out of focus to capture more diverse perspectives of the scene.

PostProcessVolumeRandomizer. Randomizes select post processing effects including vignette, exposure, white balance, depth of field, and color adjustments.

A.4 Testing Dataset Action Classes

A set of real-world wheelchair data was collected from *YouTube*. A predefined set of 16 action classes was defined before being used as keyword searches to identify relevant videos. Action classes were selected based on a mix of common actions and unique wheelchair movements. A total of 2, 464 images were collected. More information on action classes is found in table 2.



Appendix figure 1: Examples of generated data with and without increased lower body clipping. Notice the overlap between the ankles and the footplate in the motion sequences with clipping creates models where it looks like the human model has fused into the wheelchair.

A.5 Impacts of Keypoint Location Definitions

Unity Perception enables users to redefine different keypoint definitions for image annotations. Unity Perception provides a default COCO 17-keypoint annotation schema which places each keypoint directly on the joint between two bones. However, in human-annotated datasets, many evaluators place the hip much higher than the actual joint between the hips and the femur. We test Unity's default keypoint schema with lower hips against our own custom COCO 17-keypoint schema which raises the hip keypoint up the torso by $\approx$ 9cm. All data is generated in the exact same way with the only difference of how the hip keypoints are defined. Examples of how the annotation schema affects predictions are displayed in Figure 2. We see that changes in the definition of keypoints in synthetic data can drastically change the position of the changed keypoint in predictions. We believe this concept can be used to adapt and tune existing pose estimation models with different keypoint definitions through the use of only synthetic data.

Appendix table 1: Domain randomization parameters of *WheelPose*

Category	Randomizer	Parameter	Distribution
Background/Occluder Objects	BackgroundObjectPlacementRandomizer	object placement	`Cartesian[Uniform(-7.5, 7.5), Uniform(-7.5, 7.5), Uniform(-7.5, 7.5)]`
		separation distance	`Cartesian[Constant(2.5), Constant(2.5), Constant(2.5)]`
	BackgroundOccluderScaleRandomizer	object scale range	`Cartesian[Uniform(1, 12), Uniform(1, 12), Uniform(1, 12)]`
	RotationRandomizer	object rotation	`Euler[Uniform(0, 360), Uniform(0, 360), Uniform(0, 360)]`
	TextureRandomizer	textures	A set of of texture assets
	SpriteRandomizer	sprites	A set of sprite assets
	HueOffsetRandomizer	hue offset	`Uniform(-180, 180)`
Human Model	HumanGenerationRandomizer	humans per iteration	`Uniform(5, 12)`
		human pool size	`Constant(50)`
		pool refresh interval	`Constant(400)`
		age	`Uniform(10, 100)`
		height	`Uniform(0.1, 1)`
		weight	`Uniform(0, 1)`
		sex	male, female
		ethnicity	Caucasian, Asian, Latin American, African, Middle Eastern
	TransformPlacementRandomizer	synthetic human placement	`Cartesian[Uniform(-7.5, 7.5), Uniform(-7.5, 7.5), Uniform(-4, 1)]`
		synthetic human rotation	`Euler[Uniform(0, 20), Uniform(0, 360), Uniform(0, 20)]`
		synthetic human size range	`Cartesian[Uniform(0.5, 3), Uniform(0.5, 3), Uniform(0.5, 3)]`
	ForegroundObjectPlacementRandomizer	predefined model placement	`Cartesian[Uniform(-7.5, 7.5), Uniform(-7.5, 7.5), Uniform(-9, 6)]`
		predefined model separation distance	`Cartesian[Constant(3), Constant(3), Constant(3)]`
	ForegroundScaleRandomizer	predefined model scale range	`Cartesian[Uniform(0.5, 3), Uniform(0.5, 3), Uniform(0.5, 3)]`
	ForegroundRotationRandomizer	predefined model rotation	`Euler[Uniform(0, 20), Uniform(0, 360), Uniform(0, 20)]`
	NonTagAnimationRandomizer	animations	A set of AnimationClips of arbitrary length
Lights	SunAngleRandomizer	hour	`Uniform(0, 24)`
		day of the year	`Uniform(0, 365)`
		latitude	`Uniform(-90, 90)`
	LightRandomizer	intensity	`Uniform(5000, 50000)`
		color	`RGBA[Uniform(0, 1), Uniform(0, 1), Uniform(0, 1), Constant( 1)]`
		enabled	$P(enabled) = 0.8, P(disabled) = 0.2$
	LightPositionRotationRandomizer	position offset from initial position	`Cartesian[Uniform(-3.65, 3.65), Uniform(-3. 65, 3.65), Uniform(-3.65, 3.65)]`
		rotation offset from initial rotation	`Euler[Uniform(-50, 50), Uniform(-50, 50), Uniform(-50, 50)]`
Camera	CameraRandomizer	field of view	`Uniform(5, 50)`
		focal length	`Uniform(1, 23)`
		position offset from initial position	`Cartesian[Uniform(-5, 5), Uniform(-5, 5), Uniform(-5, 5)]`
		rotation offset from initial rotation	`Euler(Uniform(-5, 5), Uniform(-5, 5), Uniform(-5, 5)]`
Post Processing	PostProcessVolumeRandomizer	vignette intensity	`Uniform(5, 50)`
		fixed exposure	`Uniform(5, 10)`
		white balance temperature	`Uniform(-20, 20)`
		depth of field focus distance	`Uniform(.1, 4)`
		color adjustments: contrast	`Uniform(-30, 30)`
		color adjustments: saturation	`Uniform(-30, 30)`

Appendix table 2: Distribution of activity classes in the testing dataset.

Activity Class	Percentage of Dataset
talking	21.659%
wheelchair skills	14.692%
daily routine	13.460%
dance	10.664%
basketball	8.863%
tennis	5.829%
extreme sports	5.640%
general sports	4.645%
household chores	3.318%
shopping	2.180%
cooking	2.085%
travel	1.801%
photoshoot	1.611%
rugby	1.422%
pickleball	1.185%
stretches	0.948%

(a) (b) (c) (d)

Appendix figure 2: Examples of the different prediction outputs between a lower hip definition and a higher hip definition in synthetic data. Green represents the lower hip definition and red represents the higher hip definition. Figure 2(a) to 2(d) all show examples of where other keypoint predictions are relatively similar with the exception of the hips where the lower hip annotations are placed lower on the body compared to the higher hip annotation. Each image depicts the wheelchair user in a different angle, setting, and action.