www.cad-journal.net
Computer-Aided Design AND APPLICATIONS™

Deep Learning-Based Pose and Shape Estimation of the Human Body at a Disaster Site Utilizing Synthetic Disaster Scene Generation

Ken Nishioka[1], Zechen Zhu[2], Satoshi Kanai[3], Hiroaki Date[4], Atsushi Konno[5], Soichi Murakami[6], Toshiaki Shichinohe[7]

[1]nishioka@ist.hokudai.ac.jp, Hokkaido University
[2]Zhu-Zechen@outlook.com, Hokkaido University
[3]kanai@ssi.ist.hokudai.ac.jp, Hokkaido University
[4]hdate@ssi.ist.hokudai.ac.jp, Hokkaido University
[5]konno@ssi.ist.hokudai.ac.jp, Hokkaido University
[6]so-ichi@umin.ac.jp, Hokkaido University Hospital
[7]shichino@med.hokudai.ac.jp, Hokkaido University Hospital

Corresponding author: Ken Nishioka, nishioka@ist.hokudai.ac.jp

Abstract. In large-scale disasters, people are often trapped under the debris of collapsed buildings. In such situations, rescue and emergency medical services are often asked to identify the survivor's position and pose and to estimate which part of their body is caught under the debris. In addition, they sometimes communicate this information in real-time with medical experts in remote locations. Therefore, image processing technology must be used to support such emergency activities. In this study, a prototype system that automatically generates synthetic images and annotations was developed to supervise deep learning–based methods that can estimate the pose and shape of a human body in disaster scenes from a monocular image. In most disaster scenes, a part of the survivors' body is partially hidden. The existing deep learning–based human pose and shape (HPS) estimation methods have been trained on everyday scenes and do not work well in disaster scenes without-of-the-ordinary human poses and large occlusions with debris. In addition, it is practically impossible, both technically and ethically, to generate a large annotated image dataset of people in need of rescue in disaster scenes. To address this issue, we developed a synthetic dataset generation system that uses a game engine (Unreal Engine). Training an existing deep learning–based HPS estimation model on our synthetic dataset significantly improved the accuracy of the existing datasets. The results demonstrate the effectiveness of the proposed synthetic dataset for disaster scene applications.

Keywords: human pose and shape estimation, deep learning, game engine, synthetic dataset, disaster medicine.
DOI: https://doi.org/10.14733/cadaps.2026.572-589

1 INTRODUCTION

In large-scale disasters, people are often trapped under the debris of collapsed buildings. In such situations, emergency and rescue teams are required to locate survivors, determine which parts of their bodies are trapped, and estimate the extent of damage. Furthermore, sharing information in real-time with medical experts in remote locations and receiving advice and instructions on appropriate treatment for survivors can improve survival rates [33]. Thus, advanced IT systems are in increasing demand to support such emergency activities. A crucial element of these systems is an image processing technique that estimates the full-body posture and position of survivors from partially hidden images captured at the scene.

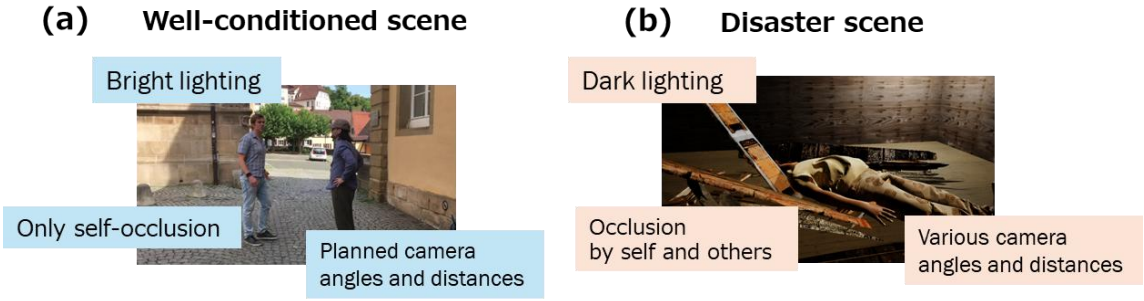

Figure 1: Difference between well-conditioned and disaster scenes: (a) an image from the dataset 3DPW [30], and (b) an image from the dataset "Disaster" created in this study.

Recently, deep learning methods for estimating the three-dimensional (3D) human body pose and shape from monocular images have been developed [28]. However, most deep learning–based methods for estimating human pose and shape (HPS) from images are only trained on data taken from well-conditioned and everyday scenes (Figure 1-a), making them vulnerable to partially occluded human bodies by debris and recognition under irregular lighting conditions (Figure 1-b). Another critical issue is the difficulties in collecting large amounts of training data from disaster scenes. Obtaining image data from actual disaster scenes and annotating human poses and shapes is challenging, and ethical considerations are also required. Therefore, a 3D HPS estimation method is required to make stable estimations from images using deep learning for disaster scenes, and a system that can systematically and efficiently generate training datasets for that HPS.

Most current HPS estimation methods [28] use the Skinned Multi-Person Linear Model (SMPL) [16], a parametric model of the human body, to construct learning models that estimate body pose and shape parameters. However, when using SMPL for deep learning–based HPS, collecting ground-truth (GT) parameter values for human poses and shapes in real-world training datasets remains labor-intensive and error-prone.

To address this issue, Black et al. [1] recently proposed BEDLAM, a large-scale synthetic dataset for HPS estimation. BEDLAM enables the efficient construction of large synthetic datasets containing human bodies with various poses and appearances and exact SMPL pose and shape parameters. Notably, deep learning models for HPS trained only on BEDLAM's synthetic data achieve the same or better estimation accuracy as those trained on real-image datasets. However, BEDLAM is designed to estimate the HPS only in general indoor and outdoor scenes. It lacks synthetic data, including out-of-the-ordinary human poses and scenes with collapsed structures and scattered debris in typical disaster scenes.

This study developed a prototype system that automatically generates synthetic images and annotations of the HPS for disaster scenes. Inspired by BEDLAM's approach, we simulate an indoor disaster scene under earthquakes using a parametric 3D human body model and a game engine. We then examine how these synthetic disaster datasets improve the stability and accuracy of existing deep learning–based HPS estimation methods.

2 RELATED WORK

2.1 3D Human Pose and Shape Estimation (HPS) Methods

In recent years, parametric human body models have been used primarily to estimate the posture and shape of a 3D human body from a single image. Furthermore, methods for estimating the parameters of human body models can be classified into two: optimization-based and regression-based methods [28]. The optimization-based method uses iterative gradient descent, whereas the regression-based method mainly uses deep learning models that directly estimate parameters trained with a large amount of training data. These are explained in detail below.

2.1.1 Parametric human body models

SMPL [16] is a skinned vertex-based model that accurately represents various body shapes in natural human poses. The model parameters were learned from the rest pose template, blend weights, pose-dependent blend shapes, identity-dependent blend shapes, and a vertices-to-joint regressor. The model is instantiated by 10 shape (size) parameters $\boldsymbol{\beta} \in R^{10}$ and 72 pose parameters $\boldsymbol{\theta} \in R^{72}$. Figure 2 shows three SMPL models with different parameters. The SMPL model has 6,890 vertices and 24 joints, enabling the reconstruction of a mesh that approximates the HPS.

Most monocular image-based HPS methods use deep learning to regress the SMPL shape and pose parameters $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ from an input image. After the development of the SMPL, several models that extend the SMPL have been proposed [21] [23] [25]. SMPL-X [23] is an extension of SMPL with refined hand and facial modeling. However, SMPL-X has more parameters than SMPL's, making pose estimation computationally expensive. In addition, when estimating SMPL-X parameters from images, faces and fingers are susceptible to two-dimensional (2D) detection errors because of their small area.

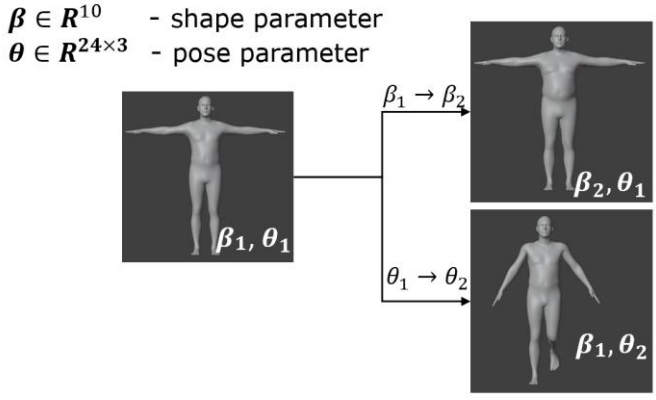

Figure 2: Change in the body surfaces of SMPL when changing SMPL parameters.

2.1.2 Optimization-based HPS methods

SMPLify [2] is an SMPL parameter estimation method that uses gradient-based optimization to minimize the objective function. This function comprises one error and four regularization terms, including the reprojection error between 2D key points (either annotated or detected) and

regularization terms for pose and shape. The regularization terms contain the difference from the human pose distribution (pose prior). SMPLify uses a mixed normal distribution as its pose prior, and many other methods have been proposed, such as the Variational Auto Encoder (VAE) [9] [23], Generative Adversarial Networks (GANs) [3] [6] [8], and normalizing flow [4] [10]. Although optimization-based HPS methods can achieve accurate results under ideal conditions, they are usually computationally expensive compared to regression-based HPS methods because of the iterative optimization. For example, SMPLify requires 20–60 s per image [8]. In addition, because the optimization is based on visible feature points, such as 2D joint positions, the results may become unstable when the hiding ratio is large.

2.1.3 Regression-based HPS methods

Recently, almost all regression-based estimation methods have used deep learning to directly derive SMPL parameters from input monocular images [15]. For example, the human mesh recovery (HMR [8] crops a sub-image region around a human and feeds it into a pretrained convolutional neural network to extract features. The HMR passes these features through a series of fully connected layers to predict SMPL parameters $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$. During HMR training, a combination of parameter regression and 2D reprojection errors is used as the loss function. For reprojection, HMR uses a camera with a fixed focal length (f = 5000 pixels). In contrast, CLIFF [12] uses information about a human's position within an image to improve global orientation estimation. Figure 3 flow of 3D HPS estimation with CLIFF. CLIFF incorporates a global 2D joint loss with the reprojection loss, allowing for a more accurate capture of perspective distortions. Although CLIFF still regresses SMPL parameters from cropped images, it more robustly handles varying focal lengths and lens distortions. However, CLIFF uses a weak-perspective camera model, which leads to projection errors in close-range, wide-angle scenes. Although jointly estimating full-perspective camera parameters is challenging because of their higher dimensionality and the depth ambiguity in monocular images, more recently, several perspective-aware models (CameraHMR [22], BLADE [31], and Zolly [32]) have addressed this issue.

For regression-based models that use deep learning, the performance decreases when data deviates from the training datasets [11] [27]. Therefore, it is necessary to develop and expand the training dataset based on the intended use of the model.

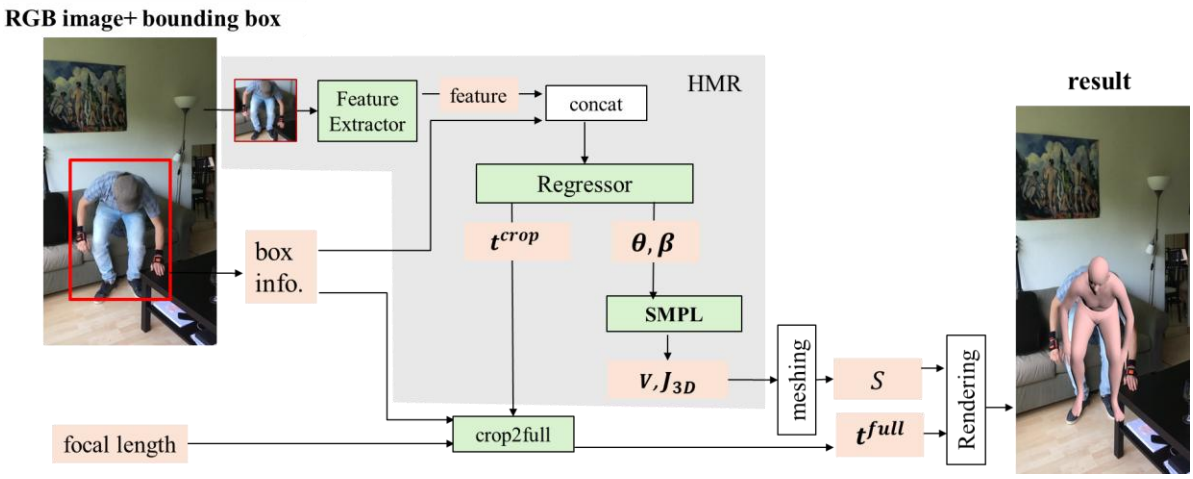

Figure 3: Process flow of 3D human pose and shape estimation with CLIFF (The figure is rewritten based on [12]).

2.2 Datasets for the HPS

Thus far, HPS methods based on regression and deep learning outperformed optimization-based methods in terms of estimation error and processing time. The regression-based HPS method requires a large, reliable image dataset that is annotated with human poses and shapes. Notably, the estimation performance of the regression-based method is highly dependent on the quality of the training dataset.

Currently, the mainstream method for generating training datasets for HPS is the use of publicly available real-world datasets, consisting of a large number of annotated real-world images. In contrast, synthesized image datasets created by computers for HPS have also been proposed recently. The following sections describe the datasets and associated challenges.

2.2.1 Real-world datasets

Real-world datasets can be created using a motion capture (MoCap) system [7] [18] [19], sensors, such as an inertial measurement unit (IMU) [30], or pseudo-annotations derived from images [13].

Human3.6M [7] was created in a multicamera studio using the MoCap system. The dataset contains 6 million frames from 11 subjects recorded by four calibrated high-resolution cameras. It provides 3D joint annotations, depth data, and 3D body scans for each subject, making it a standard benchmark for both 2D and 3D pose estimation tasks.

MPI-INF-3DHP [19] is also created using the MoCap system with markerless multiview capture in a green-screen studio, plus outdoor scenes. Indoor data were captured using 14 calibrated cameras, whereas outdoor data were captured using a portable setup with fewer cameras and a green screen for background removal. GT 3D poses were obtained through multiview reconstruction.

AMASS (Archive of Motion Capture as Surface Shapes) [18] incorporates 15 existing MoCap datasets into a unified SMPL representation. The dataset contains over 40 h of motion data from more than 300 subjects and 11,265 motion sequences. The dataset enables full 3D mesh reconstruction with 6,890 vertices per frame, making it valuable for training or pretraining models that handle diverse motions and body shapes.

3DPW (3D Poses in the Wild) [30] uses wearable IMUs combined with handheld video capture in real-world environments. The proposed 3DPW contains 60 video sequences (~51,000 frames) that capture everyday activities both outdoors and indoors. The 3DPW provides GT SMPL parameters by hybrid optimizing the IMU and video data. It is notable for in-the-wild settings where lighting, occlusion, and background variation are significant.

COCO (Common Objects in Context) [13] is a large-scale general-purpose dataset commonly used for 2D pose estimation but often used indirectly for 3D pose estimation tasks via SMPLify [2] for pseudo-3D annotations.

2.2.2 Synthetic datasets

The process of collecting and annotating data are often time-consuming and expensive, leading to poor data and insufficient data quantities. In recent years, synthetic datasets have been widely introduced into machine learning fields [17] to address this issue. Regression and deep learning–based HPS methods have also adopted this trend.

SURREAL [29] is a synthetic dataset generated by fitting accurate MoCap data to the SMPL model and then rendering posed SMPL meshes with various textures and lighting against random real-image backgrounds. It contains over 6 million frames of annotated synthetic images and videos, providing 2D and 3D joint locations, body-part segmentation masks, depth maps, optical flow, and precise 3D mesh (SMPL) parameters for each frame. SURREAL covers 4,300 body shapes, 2,000 motion sequences across 23 actions, and 40,000 diverse backgrounds. It is widely used to pretrain deep models for 3D pose estimation and human parsing, and it offers robust performance transfer to real datasets.

BEDLAM [1] is a synthetic dataset that uses SMPL-X and a physics-based approach to simulate realistic clothing and human motion. Virtual humans are animated using motion data and are placed

in rich 3D environments with varied backgrounds, lighting, and camera movements. A commercial physics engine is used to manage clothing deformation to produce realistic garment wrinkles and dynamics. BEDLAM provides monocular RGB videos with SMPL-X annotations (3D body pose, shape, hand pose, and facial pose) and diverse body shapes, motions, and clothing styles. It is used for training and benchmarking 3D HPS estimators and achieves state-of-the-art performance on real datasets. In addition, it supports cloth motion modeling, multiperson pose estimation, and adaptation to challenging visual conditions.

Synbody [34] is a large-scale synthetic dataset designed to improve 3D human modeling and perception. Based on a layered parametric model, the proposed model provides high-quality 3D annotations with rich body shapes, clothing, hair, and motion variations. It supports HPS and human neural rendering (NeRF) [20].

BlendMimic3D [14] is a synthetic dataset for 3D human pose estimation, which is a task to estimate only 3D joint positions. BlendMimic3D was created using Blender [36] and contains various types of occlusion situations, including self, object-based, and out-of-frame occlusions.

2.3 Issues of HPS in Disaster Scenes and our Contributions

To apply the current HPS to disaster scenes, a regression-based model is essential because optimization-based models have difficulty handling occlusion and longer processing times. Most existing regression models are based on deep learning and require substantial training data. However, current datasets (both real and synthetic) only cover human body movements in everyday life and do not reflect situations unique to disaster scenes. They lack data for unusual postures, such as partially debris-occluded bodies, lying down, and unnatural body postures. Furthermore, estimation accuracy is not guaranteed under nonuniform lighting conditions, strong shadows, and dark areas typical of disaster scenes. Existing synthetic dataset generation methods do not adequately model human behavior and its interactions with the environment based on physical constraints.

This research addresses these issues through the following contributions:

1. **Generation of disaster scene-specific synthetic data based on physics simulations:**
   The natural behavior of the human body after a collision with debris is reproduced using a physics engine to construct a dataset with realistic postures and occlusions.
2. **Ensures the diversity of environmental conditions:**
   The lighting conditions, viewpoints, and environmental elements are systematically varied to train robust models that can handle the various situations expected in real disaster scenes.
3. **Automatically generate annotations**:
   Automatically generate detailed annotations such as accurate 3D posture and geometry parameters, joint visibility information, and camera parameters, using a game engine.
4. **Verifying the effectiveness of the synthetic dataset for deep learning-based HPS:**
   The experiments demonstrate that the HPS model trained on the proposed synthetic dataset outperforms the model trained on the existing dataset in body estimation accuracy.

3 SIMULATION OF DISASTER SCENES AND GENERATION OF THE SYNTHETIC DATASET FOR HPS

3.1 System Overview

In this study, we developed a synthetic data generation system that can generate disaster scenes, including a human being buried in debris (hereinafter referred to as the "generation system"). We then trained a deep learning model for HPS using the synthetic dataset produced from the generation system to improve the performance of the pose and shape estimation of a survivor in need of rescue. The proposed generation system automatically synthesizes disaster scenes and renders RGB images, as well as annotations of human poses and shapes (Figure 4). The generation system was implemented on a game engine (Unreal Engine, UE) [38].

We developed two methods for generating disaster scenes. The first method is the "manual-based scene-generation method" (Section 3.3), in which the initial human poses of the survivors and the spatial positions of the debris are roughly defined in advance and manually placed in 3D space. Then, fine adjustments of the human poses and debris positions are automatically performed using the physics engine functions of the game engine. The second method is the "simulation-based scene-generation method" (Section 3.4), in which the human body is in a prone position, and debris is randomly and automatically placed in the game engine space. A falling simulation is then performed using the physics engine to dynamically simulate the changes in the poses of the falling objects and human body. For both methods, it is necessary to prepare 3D models of the disaster environment, human body, and debris in a format compatible with the UE. The details of each function of this generation system are described in the following sections.

The images in our synthetic dataset depict injured survivors; however, no real identities or biometric data were used. The dataset is currently used only internally. If disclosed, any distribution will be restricted to research purposes and will require approval from the institution's ethics review board and user consent.

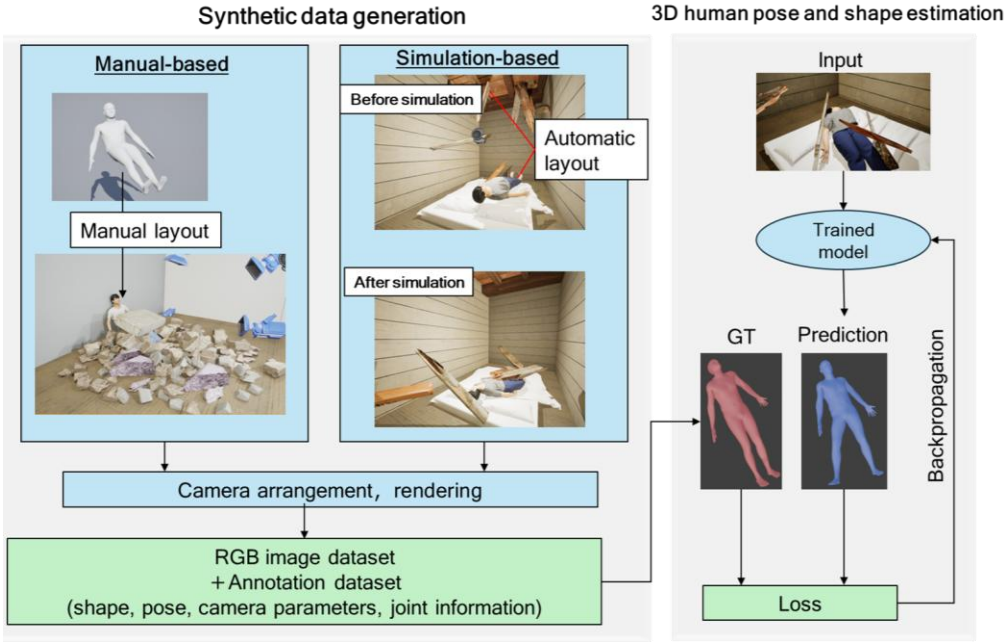

Figure 4: Synthetic data generation system to simulate disaster scenes for 3D human pose and shape estimation.

3.2 Preparation of the 3D Model Data

- **Environmental model**: Three types of floors, two types of beds, one type of ceiling, and six types of walls were prepared. 3D-textured meshes represent the beds and ceiling, while planar textures represent the other models (floors, walls).
- **Human body models:** One male and one female template model with SMPL shape parameters were prepared from the BEDLAM dataset. Then, either of the template models is instantiated by specifying the SMPL pose parameter values, which define 23 joint angles for each human body part and the orientation of the entire body. The human body models were implemented as skeletal meshes in the UE.
- **Clothing and hair models:** To ensure various human appearances, we prepared a 3D clothing model for each gender, 10 clothing texture patterns, and 12 skin texture patterns for each

gender. A separate 3D hair model was also used for the human body model with manual adjustment.

3.3 Manual-based Scene-generation Method

The following procedure outlines the manual-based scene-generation method (Figure 4, left).

(1) The user manually sets the initial position and pose (e.g., sitting pose) of the human body and debris fragment model in the indoor environment model.
(2) The physics simulation in the UE is executed automatically to fine-tune the human body's pose, ensuring that it naturally contacts the debris and floor walls.
(3) Textures, lighting conditions, virtual camera position, and poses of the environment and human body are randomly and automatically changed in the scene generated in step (2), followed by rendering the images. This process enables the semiautomatic acquisition of many accurate GT values for images and body poses for deep learning.

The advantage of this scene-generation method is its ability to easily reflect the characteristics of the disaster environment into the model. For example, the scene shown on the left side of Figure 4 was created on the basis of a realistic image of a simulated debris field, ensuring that the wall texture and debris distribution closely resemble those of an actual disaster scene. However, this manual-based method requires adjustments to the body's position and pose, which limits the variety of poses.

3.4 Simulation-based Scene-generation Method

To address the shortcomings of the manual-based scene-generation method, we developed a simulation-based scene-generation method. The following steps generate synthetic data in this method (Figure 5):

(1) **Arrangement of the static models:** The floor, walls, roof, and bed in an eight-square-meter room are initially arranged. The floor aspect ratio, roof height, bed position, and lighting conditions are randomly changed. The objects are treated as static and remain stationary during the simulation.
(2) **Initial placement of dynamic models:** An instance of the human body model (gender, skin, and clothing textures also randomly selected) is placed at random positions in the room. In addition, 0–100 debris models are randomly placed in different positions and orientations. Each debris was subjected to a gravity force proportional to its volume.
(3) **Running the simulation:** Physics simulations are performed under gravity by applying rigid body motion to the debris and ragdoll physics to the human body. Ragdoll is a dynamic model in which the human body is modeled as an assembly of multiple capsule rigid bodies, and the capsules are constrained by joints with only rotational degrees of freedom. The simulation ran for 10 s, and disaster scene data was acquired after all the objects' motions had come to a standstill.
(4) **Camera placement and rendering:** Ten cameras are randomly placed around the human model, and RGB images are rendered. Figure 6 shows examples of the disaster scene images generated from the simulation.
(5) **Generation of an annotation dataset for deep learning–based HPS:** The SMPL parameter values of the human body at the final state are archived into the annotation dataset. In addition, the mask image (human body = 1, others = 0), visibility flags of each joint from a camera, and camera intrinsic/extrinsic parameters are logged into the dataset. Figure 7 image and annotations for a camera.

By repeating the process, many training samples (RGB images, SMPL parameter values, visibility flags, and camera parameters) can be automatically synthesized and stored in the annotation dataset.

Figure 5: Simulation-based scene generation procedure (1) Arrangement of static models; (2) Initial placement of dynamic models;(3) Running the simulation;(4) Camera placement and rendering.

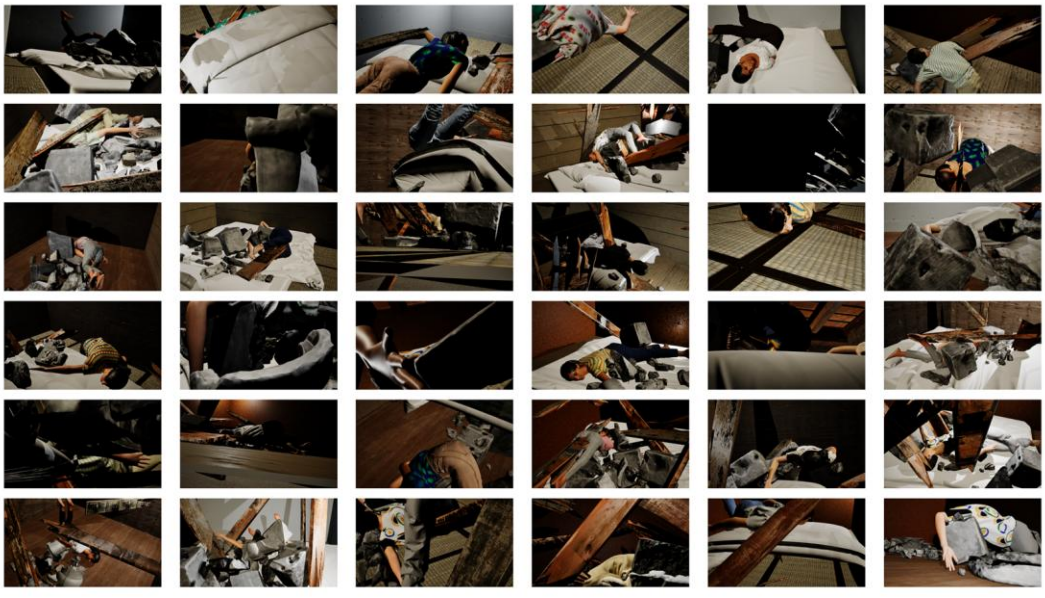

Figure 6: Example of synthetic data generation for a disaster scene.

The proposed method effectively captured collisions with debris and pose changes in a disaster scene and efficiently generated various human body poses. However, simply running the simulation can lead to physically unnatural joint angles in the human body. To address this problem, this study references physiotherapy literature [24] to set a range-of-motion (ROM) limit for joint angles. This prevents the arms and legs from deviating from the ROM during the simulation, thereby enabling the generation of natural poses.

Although our simulation captured numerous debris–body contacts, it did not fully model the complexity of human–debris interactions. The following three key limitations remained:

(1) **Independent ROM constraints**: The lack of inter-joint coupling can result in un-natural poses. Future work will integrate coupled-joint pose priors—such as VAE [23], GAN [3] [8], or normalizing flow [4] models—directly into our data-generation system.

(2) **Collider approximation:** During physics simulations, the human body is approximated by multiple capsule colliders, which may cause small gaps or mutual penetration between the body and debris. In addition, clothing lacks dedicated colliders, allowing garments to penetrate debris. Future work will address this limitation by implementing mesh-based colliders and improving cloth simulation accuracy.

(3) **Passive-victim assumption:** The subjects were treated as unconscious ragdolls without voluntary movements. Limited active movements, such as extending the arms forward during falls or light debris avoidance, could enable more natural human–debris interaction.

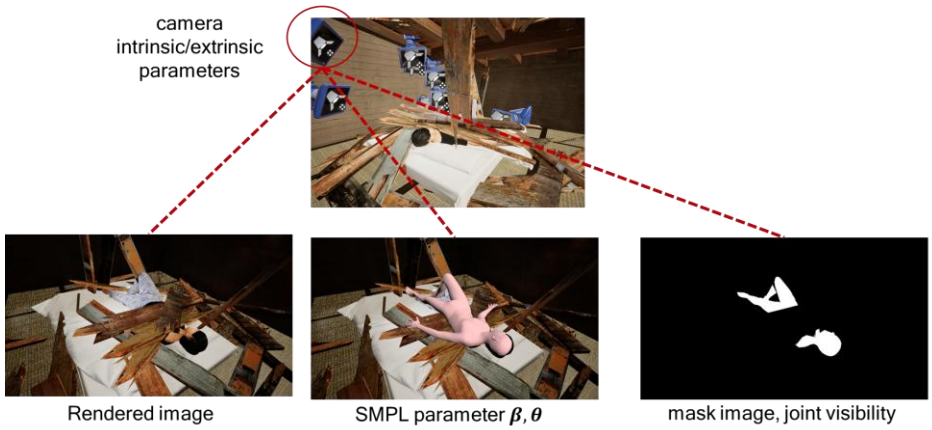

Figure 7: Example of an image and annotation for a camera.

3.5 Implementation

In UE, visual programming known as "*Blueprint*," Python, and C++ application programming interface (API) can be used to automate processes. This study used Blueprint and Python to automatically place objects and perform simulations. However, the joint position and angle data in the human body model after simulation were implemented separately using the C++ API because neither Blueprint nor Python offers an acquisition API for this purpose. The graphical user interface operations (button presses) on the UE can be automated using the Python automation library PyAutoGUI [37], which can automate most of the synthetic data generation process.

4 EXPERIMENTS

This section describes the experiments demonstrating how the estimation performance of an existing HPS network can be improved by retraining it with synthetic image datasets of the disaster scenes generated using the developed system.

4.1 HPS Method

We conducted experiments using BEDLAM [1] as a reference, which is existing research on HPS based on learning with synthetic datasets. We performed our experiments using CLIFF because BEDLAM uses a deep learning model called CLIFF [12] for HPS. CLIFF allows fair comparisons of our results under the same conditions. It also estimates the pose and shape while considering the subject's position in the entire frame. This capability is particularly useful for disaster scenarios, where subjects in photographs are often off-center rather than centrally positioned.

CLIFF uses an estimation framework based on the parametric human body model SMPL [16] to estimate the SMPL parameters for 3D HPS from the input images (Figure 3). In this framework, the human position in the image is first detected by using an object detection process. The image subregion of the human part was then sent to the CLIFF deep learning model, comprising the feature extractor and regressor. The feature extractor uses a deep learning–based 2D pose estimation model, HRNet [26]. The regressor comprises multiple fully connected and dropout layers and outputs the 3D HPS fitted to the subregion. The 3D coordinates of the surface skin mesh vertices and joint

positions were estimated by inputting the estimated parameters into the SMPL. When rendering the human body shape as an image, the surface skin mesh vertices and joints were projected onto the image using the camera's intrinsic and extrinsic parameters. In this experiment, the feature extractor and regressor were simultaneously fine-tuned using disasters.

4.2 Dataset

A summary of the datasets used in this study is shown in Table 1. The existing datasets 3DPW [30] and BEDLAM [1] were used for comparison. *Disaster-Sim* is a dataset created using the simulation-based method described in Section 3.4 and used as the training and validation data. In contrast, the *Disaster-Manual* was created using the manual-based method described in Section 3.3 and was used as test data because the manual-based method can create more natural scenes than the simulation-based method. However, the drawback of this method is that it is time-consuming to create such scenes. In addition, real-world images of the simulated debris field were used as test data. The datasets are described as follows.

4.2.1 Training and Validation data

The RGB images and annotations generated using the simulation-based method were used for training and validation. Two human models, one male and one female, were used in the simulation, and the body shape parameter β of each model was fixed. The debris pieces were randomly placed among 115 3D models of concrete blocks and pieces of wood (0.1–1.0 m^3) prepared in advance, and the physics engine was used to simulate the falling and colliding debris models with the human model.

4.2.2 Test data

Three different datasets (*Disaster-Manual, Real-DebrisField, Real-Lab*) were used for the test data. *Disaster-Manual* (Figure 10, left) comprised synthetic data, including synthetic images and their annotations, whereas *Real-DebrisField* and *Real-Lab* comprised actual images and annotations. The *Disaster-Manual* includes 802 images generated from five disaster scenes created using manual methods. The *Real-DebrisField* (Figure 10, right) consists of 17 real-world images taken at a simulated debris field in *the Hirosaki University of Health and Welfare Junior College USAR facility*. These images were taken under the assumption of a disaster situation in which a survivor's body was sandwiched between concrete blocks, and the entire body was not necessarily visible because of the blocks. In addition, the image was taken a short distance with a wide-angle camera, resulting in a pronounced perspective projection effect. Thus far, the number of real images at disaster sites for test data is limited because of the difficulty in obtaining them. However, we plan to collaborate with disaster training organizations to obtain more diverse and annotated authentic images for future disaster rescue training sites.

For quantitative evaluation using actual scenes, we used the *Real-Lab* dataset, which consists of images taken in the laboratory (Figure 8 and Figure 9). The images were taken from four directions (Left, Front, Face, and Back sides) using a small action camera (DJI action5 pro). Two subjects were recorded at different occlusion rates (100%, 90%, 80%, 60%, 50%, 40%, 20%, and 0%) (Figure 9). The images of the subjects were taken from the same position; however, only the occlusion board was moved. After shooting, the 2D joint positions of the subjects and bounding boxes were manually annotated.

Datasets	#Samples	Used for	Annotation
3DPW [30]	22,735	Training, Validation	Fully
BEDLAM [1]	1,014,622	Training, Validation	Fully
Disaster-Sim	7,010	Training, Validation	Fully
Disaster-Manual	802	Testing	Fully

Real-DebrisField	17	Testing	None
Real-Lab	69	Testing	2D joint position only

Table 1: Number of annotated images included in the experimental datasets.

Figure 8: Recording direction. (A: Side, face up; B: Side, backup; C: Front, face up; D: Front, backup).

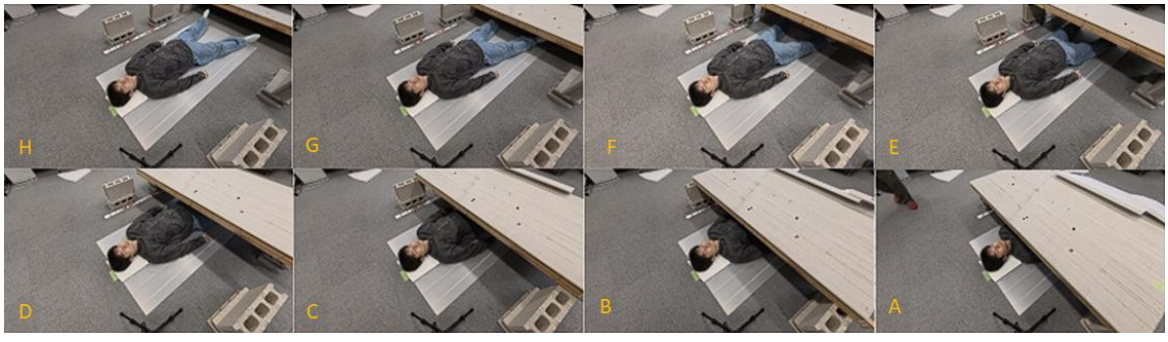

Figure 9: Occlusion degree. (From A to H: 100%, 90%, 80%, 60%, 50%, 40%, 20%, 0%).

4.3 Evaluation Metrics

Mean Per Joint Position Error (MPJPE), Procrustes-Aligned Mean Per Joint Position Error (PA-MPJPE), and mean Per Vertex position Error (PVE) are generally used metrics to assess the performance of HPS [35]. MPJPE and PVE are the root mean square values of the 3D errors at the joint or vertex positions of the skin surface mesh, evaluated after the root joint positions of the GT and the estimation are aligned by translation. These metrics are measured in millimeters. The PA-MPJPE is an error measure similar to the MPJPE, but it is evaluated after the estimated joint positions are fitted with GTs using similarity transformation. These three evaluation metrics were also used in this experiment.

4.4 Results

4.4.1 Comparison of the estimation results in different training data

Table 2 compares the results of human body shape estimation when training on a dataset containing both *Disaster-Sim* and 3DPW (*Disaster-Sim + 3DPW*), in addition to training on the dataset alone, as described in the section 4.2. An example of the estimation results is shown in Figure 10.

Training datasets	MPJPE	PA-MPJPE	PVE
3DPW [30]	449.26	150.17	496.38
BEDLAM [1]	334.13	173.87	395.02
Disaster-Sim	180.06	113.62	211.67
Disaster-Sim + 3DPW	**172.04**	**107.61**	**202.43**
Disaster-Sim-RF	187.93	118.90	221.86

Table 2: Pose and shape estimation error [mm] of HPS in different training datasets.

Disaster-Sim + 3DPW yielded the most accurate results for every metric, with an MPJPE of 172.04 mm. In contrast, the HPS models trained with only 3DPW and BEDLAM produced higher errors, with MPJPEs of 449.26 and 334.13mm, respectively. In addition, models trained on *Disaster-Sim* or *Disaster-Sim + 3DPW* estimated human body shapes and poses better than those trained on 3DPW and BEDLAM alone. This result indicates that existing HPS datasets, such as 3DPW and BEDLAM, are unsuitable for pose and shape estimation from images including human bodies buried in debris, as observed in the *Disaster-Manual*. Therefore, the proposed synthetic dataset proved to be effective in estimating the HPS buried in debris.

The pose and shape estimation results from actual images taken in a simulated debris field are shown in Figure 10. Both 3DPW and BEDLAM yielded inadequate results for the supine pose and human sizes. In contrast, the *Disaster-Sim*-trained model estimated a supine pose that was closer to the actual value. However, the estimation errors of the joint angles for both models on the actual images remain significant, indicating the need for further improvement.

One potential reason for the relatively large estimation error is the inconsistency between the camera model assumed within CLIFF and that of the actual image. CLIFF estimates the shape and pose of the human body based on a weak-perspective camera in which the person exists at a certain distance compared to the focal length of the camera [12]. However, for wide-angle images, such as the actual images in this study, in which the object is located at a short distance, a weakly transparent camera may increase the estimation error. A similar issue was reportedly observed in a previous study [5]. To better address this issue, we plan to adopt full-perspective camera models and recent methods, such as CameraHMR [22], BLADE [31], and Zolly [32], in our future work.

4.4.2 Effectiveness of medically based joint ROM settings

Table 2 compares the estimation results from training with the dataset *Disaster-Sim*, which adds anatomical ROM constraints to each joint based on existing literature, and the dataset *Disaster-Sim-RF*, which was trained with a uniform range (0 to 45 degrees) of joint ROM constraints. The *Disaster-Sim-RF* and *Disaster-Sim* datasets contain 7010 images.

Table 2 shows that Disaster-Sim with anatomical joint ROM constraints gave slightly better results than *Disaster-Sim-RF* in all evaluation values.

4.4.3 Quantitative evaluation using an actual image

Table 3 joint errors L_{2D} for *Real-Lab* images taken from a laboratory, and Figure 11 shows an example of the pose and shape estimation results. In the experiment, the original image size was 4000 × 2256 pixels. For the evaluation metrics of the estimation, L_{2D} defined by the following equation (1) was evaluated as follows:

$$L_{2D} = \frac{1}{N}\sum_{i=1}^{N}\left|\left|J_{i,2D} - \hat{J}_{i,2D}\right|\right| \tag{1}$$

where N, $J_{i,2D}$ and $\hat{J}_{i,2D}$ are the number of joints ($N = 23$), the estimated and GT positions of joint i on the image, respectively.

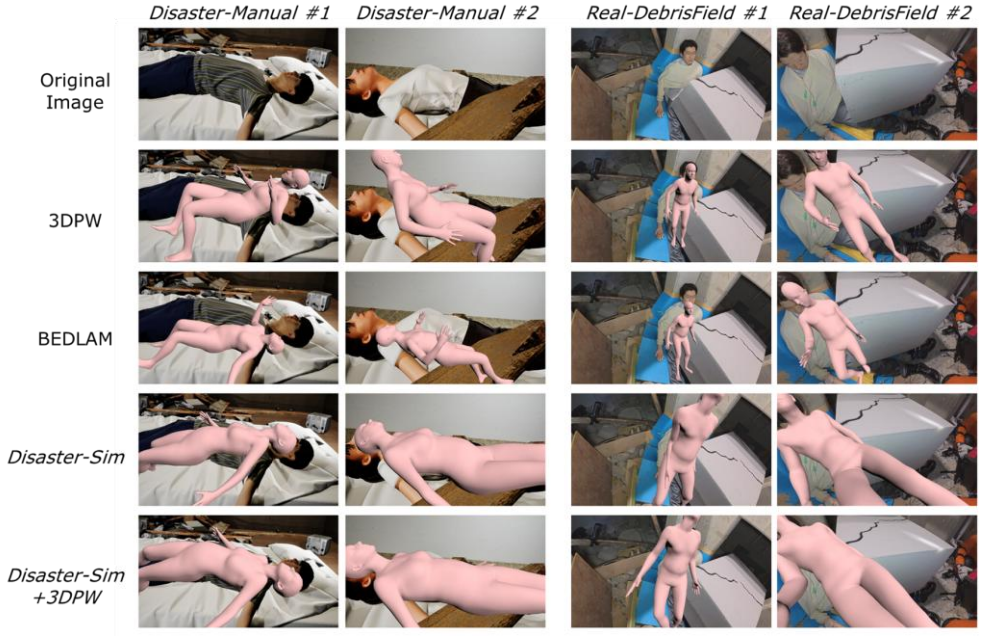

Figure 10: Results sampled from different datasets for *Disaster-Manual* and *Real-DebrisField*.

As shown in Table 3, the estimation results demonstrate that the HPS model trained by (*Disaster-Sim, Disaster-Sim + 3DPW*) achieved better accuracy than those trained by 3DPW. In addition, the subject's head position was significantly misaligned with the HPS model trained by 3DPW (Figure 10). This misalignment may be due to a few images in the 3DPW, such as when lying down. However, since most of the subjects in the *Disaster-Sim* images were lying poses, the results for the input test data in which subjects were lying down may have been better than those for datasets that only included everyday standing scenes. Therefore, the proposed synthetic dataset is effective for real-world life data involving lying poses.

4.4.4 Processing times

Table 4 lists the processing times. A "scene" in the data generation time is the data generated in 1 loop of the data generation procedure outlined in 1–5 of Section 3.4, and 10 annotated images are obtained for each scene. The generation time of *Disaster-Sim* includes the time required to create annotated images using the simulation-based method and conversion to training file format. The training time is the total time for 300 epochs, and the inference time is the time required to complete the process from image and bounding box input to rendering.

As can be seen from Table 4, data generation takes a long time, which necessitates further improvement by eliminating waste in the processing process and parallelizing the processing. Conversely, the estimation time is very fast and can be considered real-time when limited to the estimation module.

Training dataset	L_{2D}(Face, left)	L_{2D}(Back, left)	L_{2D}(Face, Front)	L_{2D}(Back, Front)
3DPW [30]	115	181	156	158

Disaster-Sim	**63**	89	65	59
Disaster-Sim + 3DPW	71	**87**	**62**	**56**

Table 3: 2D Joint errors [pixel] in different directions and training datasets.

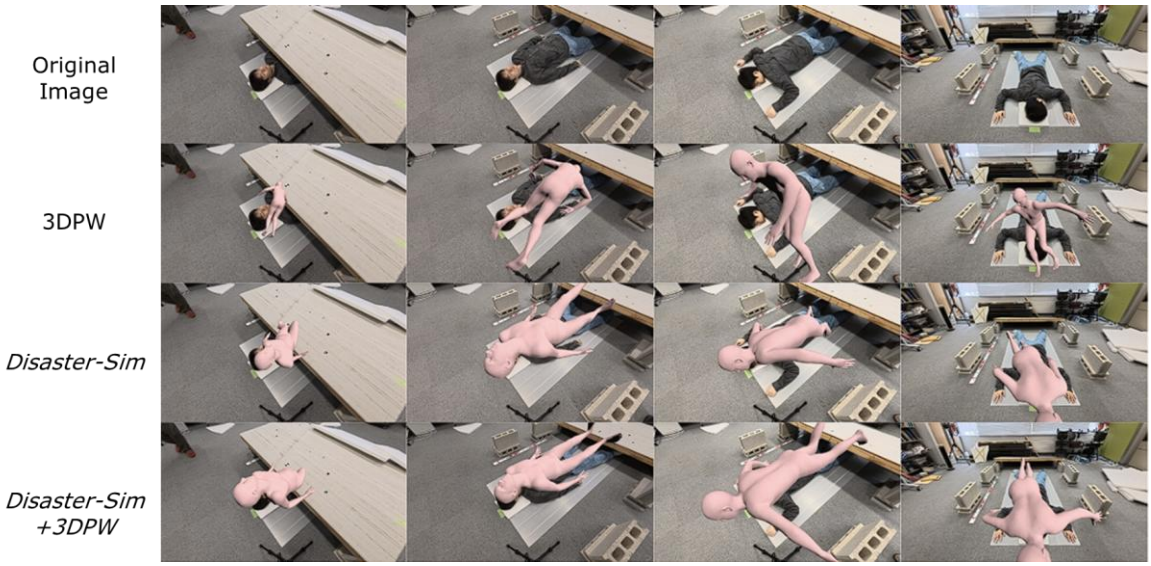

Figure 11: Results sampled from different datasets in *Real-Lab*.

Data generation time	Generation time per scene	53.9 seconds
	Generation time per image	5.39 seconds
	Generation time of *the Disaster-Sim*	11.09 hours
CLIFF processing time	Training time (using *Disaster-Sim + 3DPW)*	26.33 hours
	Inference time	21.03 FPS

Table 4: Processing times.

5 CONCLUSION AND FUTURE WORK

In this study, we proposed two new methods for synthetically creating a new training dataset that is crucial for disaster telemedicine support systems. This dataset enables accurate estimation of the 3D shape and pose of a survivor whose body is partially shielded by debris using deep learning, a game engine, and a parametric deformable human body model. The experimental results demonstrated that the HPS model trained on the proposed synthetic dataset outperformed the model trained on the existing dataset in terms of estimation accuracy.

However, the absolute estimation accuracy for real images is not yet sufficient, and future work is needed to augment the dataset to include shielding conditions closer to real environments and adapt the shape estimation model to a perspective camera model for wide-angle shots.

In addition, adding anatomical joint ROM constraints to the HPS model used for estimation is effective in improving the estimation accuracy. However, even with these constraints, the data may still contain visually unnatural poses. Therefore, the human body model must be improved by incorporating human behavior analysis to generate more realistic poses.

6 ACKNOWLEDGEMENT

This work was supported by Innovative Science and Technology Initiative for Security Grant Number JPJ004596, ATLA, Japan.

Ken Nishioka, https://orcid.org/0009-0003-2185-0290
Zechen Zhu, https://orcid.org/0009-0006-2907-8428
Satoshi Kanai, https://orcid.org/0000-0003-3570-1782
Hiroaki Date, https://orcid.org/0000-0002-6189-2044
Konno Atsushi, https://orcid.org/0000-0003-3288-8844
Soichi Murakami, https://orcid.org/0000-0003-2227-9367
Toshiaki Shichinohe, https://orcid.org/0000-0001-6614-462X

REFERENCES

[1] Black, M. J.; Patel, P.; Tesch, J.; Yang, J.: BEDLAM: A Synthetic Dataset of Bodies Exhibiting Detailed Lifelike Animated Motion, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, 2023. https://doi.org/10.1109/CVPR52729.2023.00843

[2] Bogo, F.; Kanazawa, A.; Lassner, C.; Gehler, P.; Romero, J.; Black, M. J.: Keep It SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image, in Computer Vision - ECCV 2016, 9909, 2016, 561–578. https://doi.org/10.1007/978-3-319-46454-1_34

[3] Davydov, A.; Remizova, A.; Constantin, V.; Honari, S.; Salzmann, M.; Fua, P.: Adversarial Parametric Pose Prior, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, 2022. https://doi.org/10.1109/CVPR52688.2022.01072

[4] Dünkel, O.; Salzmann, T.; Pfaff, F.: Normalizing Flows on the Product Space of SO(3) Manifolds for Probabilistic Human Pose Modeling, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, 2024. https://doi.org/10.1109/CVPR52733.2024.00222

[5] Dwivedi, S. K.; Sun, Y.; Patel, P.; Feng, Y.; Black, M. J.: TokenHMR: Advancing Human Mesh Recovery with a Tokenized Pose Representation, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, 2024. https://doi.org/10.1109/CVPR52733.2024.00132

[6] Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y.: Generative adversarial networks, Commun. ACM, 63(11), 2020, 139–144. https://doi.org/10.1145/3422622

[7] Ionescu, C.; Papava, D.; Olaru, V.; Sminchisescu, C.: Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments, IEEE Trans. Pattern Anal. Mach. Intell., 36(7), 2013, 1325–1339. https://doi.org/10.1109/TPAMI.2013.248

[8] Kanazawa, A.; Black, M. J.; Jacobs, D. W.; Malik, J.: End-to-End Recovery of Human Shape and Pose, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, 2018. https://doi.org/10.1109/CVPR.2018.00744

[9] Kingma, D. P.; Welling, M: Auto-encoding variational bayes, 2nd International Conference on Learning Representations (ICLR), Banff, AB, 2014.

[10] Kobyzev, I.; Prince, S. J. D.; Brubaker, M. A.: Normalizing Flows: An Introduction and Review of Current Methods, IEEE Trans. Pattern Anal. Mach. Intell., 43(11), 2021, 3964–3979. https://doi.org/10.1109/TPAMI.2020.2992934

[11] Koh, P. W.; Sagawa, S.; Marklund, H.; Xie, S. M.; Zhang, M.; Balsubramani, A.; Hu, W.; Yasunaga, M.; Phillips, R. L.; Gao, I.; Lee, T.; David, E.; Stavness, I.; Guo, W.; Earnshaw, B. A.; Haque, I. S.; Beery, S.; Leskovec, J.; Kundaje, A.; Pierson, E.; Levine, S.;, Finn C.; Liang, P.: WILDS: A Benchmark of in-the-Wild Distribution Shifts, International Conference on Machine Learning (ICML), virtual only, 2021.

[12] Li, Z.; Liu, J.; Zhang, Z.; Xu, S.; Yan, Y.: CLIFF: Carrying Location Information in Full Frames into Human Pose and Shape Estimation, Computer Vision – ECCV 2022, 13665, 2022, 590–606. https://doi.org/10.1007/978-3-031-20065-6_34

[13] Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C. L.: Microsoft COCO: Common Objects in Context, Computer Vision – ECCV 2014, 8693, 2014, 740-755. https://doi.org/10.1007/978-3-319-10602-1_48

[14] Lino, F.; Santiago, C.;Marques, M.: 3D Human Pose Estimation with Occlusions: Introducing BlendMimic3D Dataset and GCN Refinement, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, 2024. https://doi.org/10.1109/CVPRW63382.2024.00467

[15] Liu, Y.; Qiu, C.; Zhang, Z.: Deep learning for 3D human pose estimation and mesh recovery: A survey, Neurocomputing,596, 128049, 2024, https://doi.org/10.1016/j.neucom.2024.128049.

[16] Loper, M.; Mahmood, N.; Romero, J.; Pons-Moll, G.; Black, M. J.: SMPL: a skinned multi-person linear model, ACM Trans. Graph., 34(6), 2015, 1–16. https://doi.org/10.1145/2816795.2818013

[17] Lu, Y.; Shen, M.; Wang, H.; Wang, X.; van Rechem, C.; Fu, T.; Wei, W.: Machine Learning for Synthetic Data Generation: A Review, Jun. 30, 2024, arXiv: arXiv:2302.04062. https://doi.org/10.48550/arXiv.2302.04062

[18] Mahmood, N.; Ghorbani, N.; Troje, N. F.; Pons-Moll, G.; Black, M.: AMASS: Archive of Motion Capture As Surface Shapes, IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), 2019. https://doi.org/10.1109/ICCV.2019.00554

[19] Mehta, D.; Rhodin, H.; Casas, D.; Fua, P.; Sotnychenko, O.; Xu, W.; Theobalt, C.: Monocular 3D Human Pose Estimation in the Wild Using Improved CNN Supervision, International Conference on 3D Vision (3DV), Qingdao, China, 2017. https://doi.org/10.1109/3DV.2017.00064

[20] Mildenhall, B.; Srinivasan, P. P.; Tancik, M.; Barron, J. T.; Ramamoorthi, R.; Ng, R.: NeRF: representing scenes as neural radiance fields for view synthesis, Commun. ACM, 65(1), 2022, 99–106. https://doi.org/10.1145/3503250

[21] Osman, A. A. A.; Bolkart, T.; Tzionas, D.; Black, M. J: SUPR: A Sparse Unified Part-Based Human Representation, Computer Vision – ECCV 2022, 13662, 2022, 568–585. https://doi.org/10.1007/978-3-031-20086-1_33

[22] Patel, P. and Black, M. J.: CameraHMR: Aligning People with Perspective, 2024, arXiv: arXiv:2411.08128. https://doi.org/10.48550/arXiv.2411.08128

[23] Pavlakos, G.; Choutas, V.; Ghorbani, N.; Bolkart, T.; Osman, A. A. A.; Tzionas, D.; Black, M. J.: Expressive Body Capture: 3D Hands, Face, and Body from a Single Image, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, 2019. https://doi.org/10.1109/cvpr.2019.01123

[24] Reese, N. B. and Bandy, W. D.: Joint Range of Motion and Muscle Length Testing-E-Book: Joint Range of Motion and Muscle Length Testing-E-Book. Elsevier Health Sciences, 2016.

[25] Romero, J.; Tzionas, D.; Black, M. J.: Embodied hands: modeling and capturing hands and bodies together, ACM Trans. Graph., 36(6), 2017, 1-17. https://doi.org/10.1145/3130800.3130883

[26] Sun, K.; Xiao, B.; Liu, D.; Wang, J.: Deep High-Resolution Representation Learning for Human Pose Estimation, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, 2019. https://doi.org/10.1109/CVPR.2019.00584

[27] Taori, R.; Dave, A.; Shankar, V.; Carlini, N.; Recht, B.; Schmidt, L.: Measuring robustness to natural distribution shifts in image classification, Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS), Red Hook, NY, 2020.

[28] Tian, Y.; Zhang, H.; Liu, Y.; Wang, L.: Recovering 3D Human Mesh From Monocular Images: A Survey, IEEE Trans. Pattern Anal. Mach. Intell., 45(12), 2023, 15406–15425. https://doi.org/10.1109/TPAMI.2023.3298850
[29] Varol, G.; Romero, J.; Martin, X.; Mahmood, N.; Black, M. J.; Laptev, I.; Schmid, C.: Learning from Synthetic Humans, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017. https://doi.org/10.1109/CVPR.2017.492.
[30] Von Marcard, T.; Henschel, R.; Black, M. J.; Rosenhahn, B.; Pons-Moll, G.: Recovering Accurate 3D Human Pose in the Wild Using IMUs and a Moving Camera, Computer Vision – ECCV 2018, 11214, 2018, 614–631. https://doi.org/10.1007/978-3-030-01249-6_37
[31] Wang, S.; Li, J.; Li, T.; Yuan, Y.; Fuchs, H.; Nagano, K.; Mello, S. D.; Stengel, M.: BLADE: Single-view Body Mesh Learning through Accurate Depth Estimation, Dec. 11, 2024, arXiv: arXiv:2412.08640. https://doi.org/10.48550/arXiv.2412.08640
[32] Wang, W.; Ge, Y.; Mei, H.; Cai, Z.; Sun, Q.; Wang, Y.; Shen, C.; Yang, L.; Komura, T.: Zolly: Zoom Focal Length Correctly for Perspective-Distorted Human Mesh Reconstruction, IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 2023. https://doi.org/10.1109/ICCV51070.2023.00363
[33] Xiong, W.; Bair, A.; Sandrock, C.; Wang, S.; Siddiqui, J.; Hupert, N.: Implementing Telemedicine in Medical Emergency Response: Concept of Operation for a Regional Telemedicine Hub, J Med Syst, 36(3), 2012, 1651–1660. https://doi.org/10.1007/s10916-010-9626-5
[34] Yang, Z.; Cai, Z.; Mei, H.; Liu, S.; Chen, Z.; Xiao, W.; Wei, Y.; Qing, Z.; Wei, C.; Dai, B.; Wu, W.; Qian, C.; Lin, D.; Liu, Z.; Yang, L.: SynBody: Synthetic Dataset with Layered Human Models for 3D Human Perception and Modeling, IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 2023. https://doi.org/10.1109/ICCV51070.2023.01855
[35] Zheng, C.; Wu, W.; Chen, C.; Yang, T.; Zhu, S.; Shen, J.; Kehtarnavaz, N.; Shah, M.: Deep Learning-based Human Pose Estimation: A Survey, ACM Comput. Surv., 56(1), 2024, 1–37. https://doi.org/10.1145/3603618
[36] Blender - a 3D modelling and rendering package., https://www.blender.org/, Blender Online Community.
[37] pyautogui, https://github.com/asweigart/pyautogui
[38] Unreal Engine, https://www.unrealengine.com/, Epic Games.