

Bodies at Rest: 3D Human Pose and Shape Estimation from a Pressure Image using Synthetic Data

Henry M. Clever¹, Zackory Erickson¹, Ariel Kapusta¹, Greg Turk¹, C. Karen Liu², and Charles C. Kemp¹

¹Georgia Institute of Technology, Atlanta, GA, USA, ²Stanford University, Stanford, CA, USA

{henryclever, zackory, akapusta}@gatech.edu, turk@cc.gatech.edu, karenliu@cs.stanford.edu, charlie.kemp@bme.gatech.edu

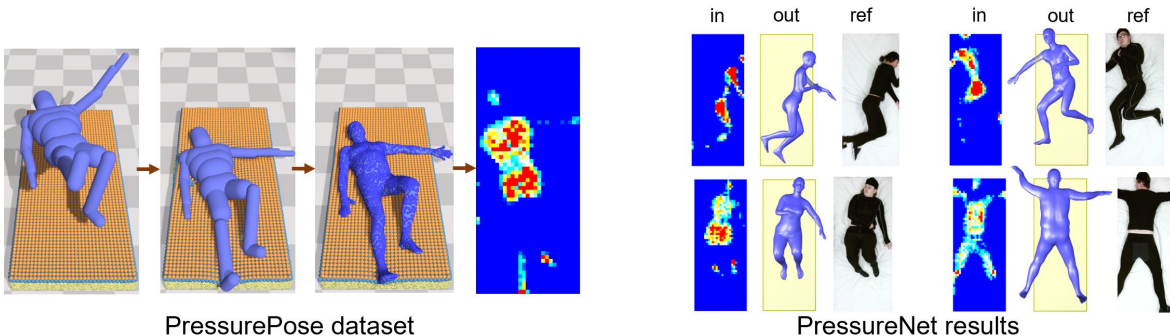


Figure 1. Left: The PressurePose dataset has 206K 3D human poses and shapes with pressure images generated by physics simulations that drop articulated rigid body models and soft body models on a soft body model of a bed and pressure sensing mat. Right: PressureNet is a deep learning model trained on synthetic data that performs well on real data: pressure image input with gender (in), 3D human mesh output (out), RGB image for reference (ref).

Abstract

People spend a substantial part of their lives at rest in bed. 3D human pose and shape estimation for this activity would have numerous beneficial applications, yet line-of-sight perception is complicated by occlusion from bedding. Pressure sensing mats are a promising alternative, but training data is challenging to collect at scale. We describe a physics-based method that simulates human bodies at rest in a bed with a pressure sensing mat, and present PressurePose, a synthetic dataset with 206K pressure images with 3D human poses and shapes. We also present PressureNet, a deep learning model that estimates human pose and shape given a pressure image and gender. PressureNet incorporates a pressure map reconstruction (PMR) network that models pressure image generation to promote consistency between estimated 3D body models and pressure image input. In our evaluations, PressureNet performed well with real data from participants in diverse poses, even though it had only been trained with synthetic data. When we ablated the PMR network, performance dropped substantially.

1. Introduction

Humans spend a large part of their lives resting. While resting, humans select poses that can be sustained with little physical exertion. Our primary insight is that human bodies at rest can be modeled sufficiently well to generate synthetic

data for machine learning. The lack of physical exertion and absence of motion makes this class of human activities amenable to relatively simple biomechanical models similar to the ragdoll models used in video games [35].

We apply this insight to the problem of using a pressure image to estimate the 3D human pose and shape of a person resting in bed. This capability would be useful for a variety of healthcare applications such as bed sore management [17], tomographic patient imaging [18], sleep studies [9], patient monitoring [10], and assistive robotics [13]. To this end, we present the PressurePose dataset, a large-scale synthetic dataset consisting of 3D human body poses and shapes with pressure images (Fig. 1, left). We also present PressureNet, a deep learning model that estimates 3D human body pose and shape from a low-resolution pressure image (Fig. 1, right).

Prior work on the problem of human pose estimation from pressure images [9, 13, 18, 22, 29] has primarily used real data that is challenging to collect. Our PressurePose dataset has an unprecedented diversity of body shapes, joint angles, and postures with more thorough and precise annotations than previous datasets (Table 2). While recent prior work has estimated 3D human pose from pressure images, [9, 13], to the best of our knowledge PressureNet is the first system to also estimate 3D body shape.

Our synthetic data generation method first generates diverse samples from an 85 dimensional human pose and

shape space. After rejecting samples based on self-collisions and Cartesian constraints, our method uses each remaining sample to define the initial conditions for a series of two physics simulations. The first finds a body pose that is at rest on a simulated bed. Given this pose, the second physics simulation generates a synthetic pressure image.

Our method uses SMPL [32] to generate human mesh models and a capsulized approximation of SMPL [4] to generate articulated rigid-body models. The first physics simulation drops a capsulized articulated rigid-body model with low-stiffness, damped joints on a soft-body model of a bed and pressure-sensing mat. Once the articulated body has settled into a statically stable configuration, our method converts the settled capsulized model into a particle-based soft body without articulation. This soft body model represents the shape of the body, which is important for pressure image synthesis. The second physics simulation drops this soft-body model from a small height onto the soft-body bed and sensor model. Once settled, the simulated sensor produces a pressure image, which is stored along with the settled body parameters.

Our deep learning model, PressureNet, uses a series of two networks modules. Each consists of a convolutional neural network (CNN) based on [13], a kinematic embedding model from [27] that produces a SMPL mesh [32], and a pressure map reconstruction (PMR) network. The PMR network serves as a model of pressure image generation. It is a novel component that encourages consistency between the mesh model and the pressure image input. Without it, we found that our deep learning models would often make mistakes that neglected the role of contact between the body and the bed, such as placing the heel of a foot at a location some distance away from an isolated high pressure region.

When given a mesh model of the human body, the PMR network outputs an approximate pressure image that the network can more directly compare to the pressure image input. These approximate pressure images are used in the loss function and as input to a second residual network trained after the first network to correct these types of errors and generally improve performance.

In our evaluation, we used a commercially available pressure sensing mat (BodiTrak BT-3510 [34]) placed under the fitted sheet of an Invacare Homecare Bed [26]. This sensing method has potential advantages to line-of-sight sensors due to occlusion of the body from bedding and other sources, such as medical equipment. However, the mat we used provides low-resolution pressure images (64×27) with limited sensitivity and dynamic range that make the estimation problem more challenging.

We only trained PressureNet using synthetic data, yet it performed well in our evaluation with real data from 20 people, including successfully estimating poses that have not previously been reported in the literature, such as supine

work	data: (R)real, (S)synth	modality: (P)ressure, (D)epth (T)hermal, (IRS) - infrared selective	3D: (Y)es, (N)o	human representation: (S)keleton, (M)esh	postures	# joints	# identities	# images
[22]	R	P	Y	M	SP+, K	18	1	?
[18]	R	D, P	N	S	SP, L, P	10	16	1.1 K
[29]	R	P	N	S	SP, L	8*	12	1.4 K
[1]	R	D	Y	S	I/O, SP, L	14	10	180 K
[10]	R	RGB	N	S	SP, UNK	7	3	13 K
[13]	R	P	Y	S	SP, ST, K	14	17	28 K
[9]	R	P	Y	S	SP+, L+, ST	14	6	60
[31]	R	IRS	N	S	SP+, L+	14	2	419
[30]	R	T	N	S	SP+, L+	14	109	14 K
Ours	S/ R	P	Y	M	SP+, L+, P+, K, CL HBH, PHU	24	200K/ 20	200K/ 1K

posture key: SP - supine. L - lateral. P - prone. K - knee raised. I/O - getting in/out of bed. ST - sitting. CL - crossed legs. HBH - hands behind head. PHU - prone hands up. + indicates a continuum between postures. * indicates limbs.

Table 1. Comparison of Literature: Human Pose in Bed.

poses with hands behind the head. To improve the performance of the model with real data, we used custom calibration objects and an optimization procedure to match the physics simulation to the real world prior to synthesizing the training data. We also created a noise model in order to apply noise to the synthetic pressure images when training PressureNet.

Our contributions include the following:

- A physics-based method to generate simulated human bodies at rest and produce synthetic pressure images.
- The PressurePose dataset, which consists of (1) 206K synthetic pressure images (184K train / 22K test) with associated 3D human poses and shapes¹ and (2) 1,051 real pressure images and RGB-D images from 20 human participants².
- PressureNet³, a deep learning model trained on synthetic data that estimates 3D human pose and shape given a pressure image and gender.

2. Related work

Human pose estimation. There is long history of human pose estimation from camera images [2, 29, 37, 45, 46] and the more recent use of CNNs [48, 49]. The field has been moving rapidly with the estimation of 3D skeleton models [40, 53], and human pose and shape estimation as a 3D mesh [4, 27, 39] using human body models such as SCAPE [3] and SMPL [32]. These latter methods enforce physical constraints to provide kinematically feasible pose estimates, some via optimization [4] and others using learned embedded kinematics models [13, 27, 53]. Our ap-

¹Synthetic dataset: doi.org/10.7910/DVN/IAP10X

²Real dataset: doi.org/10.7910/DVN/KO44ML

³Code: github.com/Healthcare-Robotics/bodies-at-rest

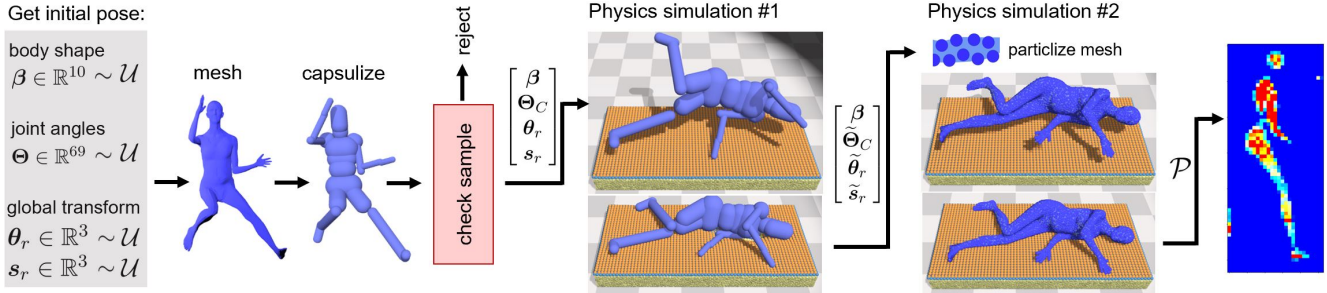


Figure 2. We generate the initial pose from scratch, using random sampling of the body shape, joint angles, and global transform on the bed. We use rejection sampling to distribute the poses and remove self-collisions. Then, we rest a dynamic capsulized human model onto a soft bed using DartFlex, a fusion of DART and FleX simulators, to get an updated resting pose. Because this model is a rather rough approximation of human shape, we then use FleX to particulate a finer body representation to get the pressure image.

proach builds on these works both directly through the use of available neural networks (e.g. SMPL embedding) and conceptually.

While pressure image formation differs from conventional cameras, the images are visually interpretable and methods developed in the vision community are well suited to pressure imagery [8, 27, 49]. PressureNet’s model of pressure image generation relates to recent work on physical contact between people and objects [6, 23, 24]. It also relates to approaches that fine-tune estimates based on spatial differences between maps at distinct stages of estimation [7, 8, 36, 49].

Human pose at rest. Human pose estimation has tended to focus on active poses. Poses in bed have attracted special attention due to their relevance to healthcare. Table 2 provides an overview of work on the estimation of human pose for people in bed. These efforts have used a variety of sensors including RGB cameras [10], infrared lighting and cameras for darkened rooms [31], depth cameras to estimate pose underneath a blanket profile [1], thermal cameras to see through a blanket [30], and pressure mats underneath a person [9, 13, 14, 18, 22, 29].

Researchers have investigated posture classification for people in bed [17, 18, 38]. There has been a lack of consensus on body poses to consider, as illustrated by Table 2. Some works focus on task-related poses, such as eating [1], and stretching [9]. Poses can increase ambiguity for particular modalities, such as lack of contact on a pressure mat (e.g. knee in the air) [13, 21] or overlapping body parts facing a thermal camera [30].

Large datasets would be valuable for deep learning and evaluation. While some bed pose work has used thousands of images they have either had few participants [10] or poses highly concentrated in some areas due to many frames being captured when there is little motion [1, 9, 13]. An exception is recent work by Liu et al. [30], which has 109 participants.

Generating data in simulation. Approaches for generating synthetic data that model humans in the context of deep learning include physics-based simulators such as

DART [28] and PyBullet [16] and position-based dynamics simulators such as PhysX [15] and FleX [33]. Some have used these tools to simulate deformable objects like cloth [12, 15]. For vision, creating synthetic depth images is relatively straightforward (e.g. [1]) while RGB image synthesis relies on more complex graphics approaches [11, 51, 52].

Some past works have simulated pressure sensors. One approach is to model the array as a deformable volume that penetrates the sensed object, where force is a function of distance penetrated [42]. Others model pressure sensing skin as a mass-spring-damper array [19, 25]; the former considers separate layers for the skin and the sensor, a key attribute of pressure arrays covering deformable objects.

3. PressurePose Dataset Generation

Our data generation process consists of three main stages, as depicted in Fig. 2: sampling of the body pose and shape; a physics simulation to find a body pose at rest; and a physics simulation to generate a pressure image. We use two simulation tools, FleX (Section 3.1) for simulating soft body dynamics, and DART (Section 3.2) for articulated rigid body dynamics.

Sample initial pose and shape. We sample initial pose (i.e. joint angles) and body shape parameters from the SMPL human model [32]. The pose consists of 69 joint angles, $\Theta \in \mathbb{R}^{69}$, which we sample from a uniform distribution, \mathcal{U} , bounded by joint angle limits defined for the hips, knees, shoulders, and elbows in [5, 44, 47]. We initialize the human body above the bed with a uniformly sampled roll $\theta_{r,1}$, yaw $\theta_{r,3}$, and 2D translation $\{s_{r,1}, s_{r,2}\}$ across the surface of the bed. The pitch $\theta_{r,2}$ is set to 0 and the distance normal to the bed $s_{r,3}$ is based on the position of the lowest initial joint position. This determines the global transform, $\{\theta_r, s_r\} \in \mathbb{R}^6$. The shape of a SMPL human is determined from a set of 10 PCA parameters, $\beta \in \mathbb{R}^{10}$, which we also sample uniformly, bounded by $[-3, 3]$ following [43]. We use rejection sampling in three ways for generating initial poses: to more uniformly distribute overall pose about the Cartesian space (rather than the uniformly

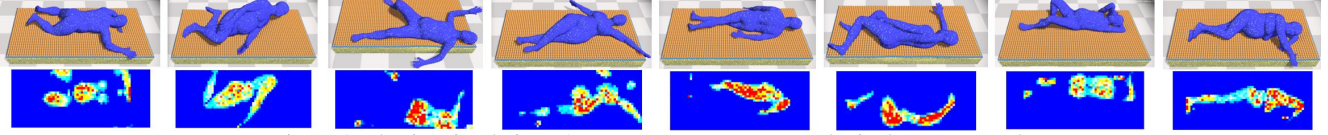


Figure 3. Physics simulation #2 output: PressurePose synthetic dataset examples.

sampled joint space), to create a variety of data partitions representing specific common postures (e.g. hands behind the head), and to reject pose samples when there are self-collisions. See Appendix A.1. This step outputs pose and shape parameters $\{\beta, \Theta_C, \theta_r, s_r\}$, where Θ_C is a set of joint angles conditioned on β that has passed these criteria.

Physics Simulation #1: Resting Pose. We use FleX [33] to simulate a human model resting on a soft bed, which includes a mattress and a synthetic pressure mat on the surface of the mattress (Fig 2). The human is modelled as an articulated rigid body system made with capsule primitives, which is a dynamic variant of the SMPL model. Once the simulation nears static equilibrium, we record the resting pose $\{\tilde{\Theta}_C, \tilde{\theta}_r, \tilde{s}_r\}$.

FleX is a position-based dynamics simulator with a unified particle representation that can efficiently simulate rigid and deformable objects. However, FleX does not currently provide a way for particles to influence the motions of rigid capsules. To overcome this limitation, we use DART [28] to model the rigid body dynamics of the capsulized human model. We combine FleX and DART through the following loop: 1) DART moves the capsulized articulated rigid body based on applied forces and moments. 2) FleX moves the soft body particles in response to the motions of the rigid body. 3) We compute new forces and moments to apply in DART based on the state of the FleX particles and the capsulized articulated rigid body. 4) Repeat. We call the combination of the two simulators DartFleX and Section 3.2 provides further details.

Physics Simulation #2: Pressure Image. The settled, capsulized body is insufficient for producing a realistic pressure image: it approximates the human shape too roughly. Instead, we create a weighted, particlized, soft human body in FleX (Figs. 2 and 3) from the SMPL [32] mesh using body shape and resting pose $\{\beta, \Theta_C, \theta_r\}$. We initialize the particlized human with 2D translation over the surface of the mattress $\{\tilde{s}_{r,1}, \tilde{s}_{r,2}\} \in \tilde{s}_r$. We set $s_{r,3}$, the position normal to gravity, so the body is just above the surface of the bed. We then start the simulation, resting the particlized body on the soft bed, and record the pressure image \mathcal{P} once the simulation has neared static equilibrium. We note that this particlized representation has no kinematics and cannot be used to adjust a body to a resting configuration; thus our use of two separate dynamic simulations.

3.1. Soft Body Simulation with FleX.

We simulate the sensing array by connecting FleX particles in a way that mimics real pressure sensing fabric, and

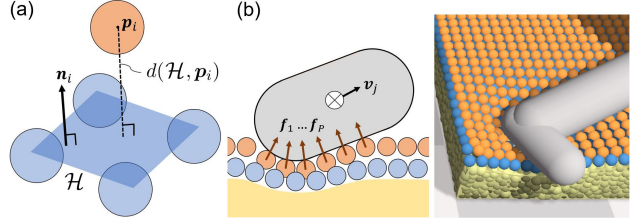


Figure 4. (a) Synthetic pressure mat structure. Pressure is a function of the penetration of the top layer array particle into the four underlying particles. (b) DartFleX collision between a capsulized limb and the simulated bed and pressure-sensing mat.

model the mattress with a soft FleX object.

Soft Mattress and Pressure Sensing Mat. Here we describe the soft mattress and pressure sensing array within the FleX environment, as shown in Fig. 4 and further described in Appendix A.3. The mattress is created in a common twin XL size with clusters of particles defined by their spacing, D_M , radius, R_M , stiffness, K_M , and particle mass, m_M , parameters. We then create a simulated pressure sensing mat on top of the mattress that is used to both generate pressure images and to help the human model reach a resting pose by computing the force vectors applied to the various segments of the human body. The mat consists of two layers of staggered quad FleX cloth meshes in a square pyramid structure, where each layer is defined by its stretching, K_σ , bending, K_B , and shear, K_τ , stiffnesses, which are spring constraints on particles that hold the mat together. A compression stiffness, K_C , determines the bond strength between the two layers, and its mass is defined by m_L .

We model force applied to the mat as a function of the particle penetration vector x_i based on the pyramid structure in Fig. 4 (a). Force increases as the i^{th} particle on the top layer, p_i , moves closer to the four particles underneath.

$$x_i = (d_0 - d(\mathcal{H}, p_i))n_i \quad (1)$$

where d is the distance between particle p_i and an approximate underlying plane \mathcal{H} , d_0 is the initial distance at rest prior to contact, and n_i is the normal vector of the approximate underlying plane.

Sensor Model. The BodiTrak pressure-sensing mat has an array of pressure-sensing taxels (tactile pixels). The four particles at the base of the pyramid structure in Fig. 4 (a) model the 1” square geometry of a single pressure-sensing taxel. We model the pressure output, u_i , of a single taxel, i , using a quadratic function of the magnitude of the penetration vector x_i .

$$u_i = (C_2|\mathbf{x}_i|^2 + C_1|\mathbf{x}_i| + C_0) \quad (2)$$

where C_2 , C_1 , and C_0 are constants optimized to fit calibration data, as described in Section 3.3.

3.2. DartFlex: Resting a Dynamic Ragdoll Body

The purpose of DartFlex is to allow rigid body kinematic chains to interact with soft objects by coupling the rigid body dynamics solver in DART to the unified particle solver in FleX as shown in Fig. 4 (b).

Dynamic rigid body chain. Our rigid human body model relies on a capsulized approximation to the SMPL model, following [4]. To use this model in a dynamics context, we calculate the per-capsule mass based on volume ratios from a person with average body shape $\tilde{\beta} = \mathbf{0}$, average body mass, and mass percentage distributions between body parts as defined by Tozeren [50]. For joint stiffnesses $\mathbf{k}_\theta \in \mathbb{R}^{69}$, we tune parameters to achieve the low stiffness characteristics of a ragdoll model that can settle into a resting pose on a bed due to gravity. We set torso and head stiffness high so that they are effectively immobile, and joint damping $\mathbf{b}_\theta = 15\mathbf{k}_\theta$ to reduce jitter.

DartFlex Physics. We initialize the same capsulized model in both DART and FleX. We apply gravity in DART, and take a step in the DART simulator. We get a set of updated dynamic capsule positions and orientations, and move the static geometry counterparts in FleX accordingly. In order to transfer force data from FleX to DART, we first check if any top layer pressure mat particles are in contact. Each particle i in contact has a penetration vector $\mathbf{x}_i(t)$ (see equation 1) at time t , which we convert to normal force vector $\mathbf{f}_{N,i} \in \mathbb{R}^3$ using a mass-spring-damper model [41]:

$$\mathbf{f}_{N,i} = k\mathbf{x}_i(t) + b\dot{\mathbf{x}}_i(t), \quad (3)$$

where k is a spring constant, b is a damping constant, and $\mathbf{f}_{N,i} \perp \mathcal{H}$. We then assign each force to its nearest corresponding capsule j . Given the velocity, \mathbf{v}_j , of capsule j and a friction coefficient, μ_k , we compute the frictional force $\mathbf{f}_{T,i}$ for the i^{th} particle in contact:

$$\mathbf{f}_{T,i} = -\mu_k |\mathbf{f}_{N,i}| \frac{\mathbf{v}_j - \text{proj}_{\mathbf{f}_{N,i}} \mathbf{v}_j}{|\mathbf{v}_j - \text{proj}_{\mathbf{f}_{N,i}} \mathbf{v}_j|} \quad (4)$$

where proj is an operator that projects \mathbf{v}_j orthogonally onto a straight line parallel to $\mathbf{f}_{N,i}$. In our simulation, we set $b = 4k$ and $\mu_k = 0.5$, and we find k through a calibration sequence described in Section 3.3. We can then compute the total particle force, \mathbf{f}_i :

$$\mathbf{f}_i = \mathbf{f}_{N,i} + \mathbf{f}_{T,i} \quad (5)$$

We then compute a resultant force \mathbf{F}_j in FleX for the j^{th} body capsule, based on the sum of forces from P particles in contact with the capsule plus gravity, \mathbf{F}_g :

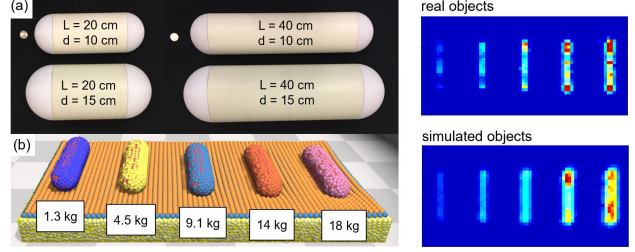


Figure 5. (a) Rigid calibration capsules with quarters (U.S. coins) shown for size. (b) Simulated capsules. (right) Real and simulated pressure images prior to calibration.

$$\mathbf{F}_j = \sum_{i=1}^P \mathbf{f}_i + \mathbf{F}_g \quad (6)$$

Moment M_j is computed on each capsule j from P particles in contact, where \mathbf{r}_i is the moment arm between a particle and the capsule center of mass:

$$\mathbf{M}_j = \sum_{i=1}^P \mathbf{r}_i \times \mathbf{f}_i \quad (7)$$

The resultant forces and moments are applied in DART, a step is taken with the forces and gravity applied to each body part, and the DartFlex cycle repeats. We continue until the capsulized model settles and then record resting pose $\tilde{\Theta}_C$, root position \tilde{s}_r , and root orientation $\tilde{\theta}_r$.

3.3. Calibration

We calibrated our simulation using the rigid capsule shapes in Fig. 5 (a). We placed varying weights on them on the real pressure-sensing mat and recorded data, and then created matching shapes in simulation. We first calibrated the FleX environment using the particlized capsules shown in Fig. 5 (b) using the covariance matrix adaptation evolution strategy (CMA-ES) [20] to match synthetic pressure images and real pressures images of the calibrated objects by optimizing D_M , R_M , K_M , m_M , d_0 , K_σ , K_B , K_τ , K_C , m_L , C_2 , C_1 , and C_0 .

We also measure how much the real capsules sink within the mattress. We use these measurements to calibrate the mass-spring-damper model in equation 3. We fit the simulated capsule displacement to the real capsule displacement to solve for the spring constant k and then set $b = 4k$ and $\mu_k = 0.5$. See Appendix A.4 and A.5 for details.

4. PressureNet

Given a pressure image of a person resting in bed and a gender, PressureNet produces a posed 3D body model. PressureNet (Fig. 6 (a)) consists of two network modules trained in sequence (“Mod1” and “Mod2”). Each takes as input a tensor consisting of three channels: pressure, edges,

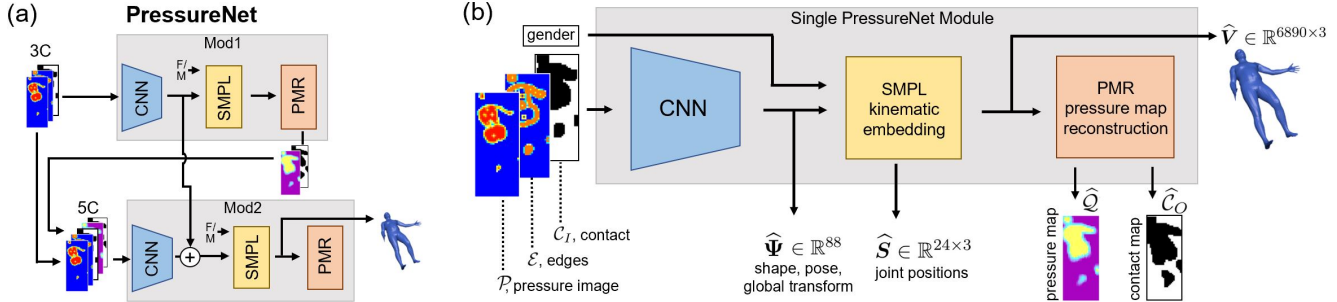


Figure 6. (a) PressureNet: We combine two network modules (“Mod1” and “Mod2”) in series. Mod1 learns a coarse estimate and Mod2 fine-tunes, by learning a residual that takes as input the two maps reconstructed by Mod1 combined with the input to Mod1. (b) Detailed description of a single PressureNet module showing the novel PMR network that reconstructs pressure and contact maps.

and contact $\{\mathcal{P}, \mathcal{E}, \mathcal{C}_I\} \in \mathbb{R}^{128 \times 54 \times 3}$, which are shown in Fig. 6 (b), as well as a binary flag for gender. \mathcal{P} is the pressure image from a pressure sensing mat, \mathcal{E} results from an edge detection channel consisting of a sobel filter applied to \mathcal{P} , and \mathcal{C}_I is a binary contact map calculated from all non-zero elements of \mathcal{P} . Given this input, each module outputs both an SMPL mesh body and two reconstructed maps produced by the PMR network, $\{\hat{Q}, \hat{C}_O\}$, that estimate the pressure image that would be generated by the mesh body. Mod2 has the same structure as Mod1, except that it takes in two additional channels: the maps produced by PMR in Mod1 $\{\hat{Q}_1, \hat{C}_{O,1}\}$. We train PressureNet by training Mod1 to produce a coarse estimate, freezing the learned model weights, and then training Mod2 to fine-tune the estimate.

CNN. The first component of each network module is a CNN with an architecture similar to the one proposed by Clever et al [13]. Notably, we tripled the number of channels in each convolutional layer. See Appendix B.1 for details. During training, only the weights of the CNNs are allowed to change. All other parts of the networks are held constant. The convolutional model outputs the estimated body shape, pose, and global transform, $\hat{\Psi} = \{\hat{\Theta}, \hat{\beta}, \hat{s}_r, \hat{x}_r, \hat{y}_r\}$, with the estimated joint angles $\hat{\Theta} \in \mathbb{R}^{69}$, body shape parameters $\hat{\beta} \in \mathbb{R}^{10}$, global translation of the root joint with respect to the bed $\hat{s}_r \in \mathbb{R}^3$, and parameters \hat{x}_r, \hat{y}_r which define a continuous orientation for the root joint of the body with $\{x_u, x_v, x_w\} \in \mathbf{x}_r$, $\{y_u, y_v, y_w\} \in \mathbf{y}_r$ for 3 DOF, i.e. $\theta_{r,u} = \text{atan2}(y_u, x_u)$ and $\{\theta_{r,u}, \theta_{r,v}, \theta_{r,w}\} \in \theta_r \in \mathbb{R}^3$.

SMPL kinematic embedding. $\hat{\Psi}$ feeds into a kinematic embedding layer (see Fig. 6), which uses the SMPL differentiable kinematics model from [27] to learn to estimate the shape, pose, and global transform. This embedding outputs joint positions for the human body, \hat{S} , and a SMPL mesh consisting of vertices \hat{V} ; and relies on forward kinematics to ensure body proportions and joint angles match real humans.

PMR. The final component of each module, the PMR network, reconstructs two maps based on the relationship between the SMPL mesh \hat{V} and the surface of the bed. The

reconstructed pressure map (\hat{Q}) corresponds with the input pressure image, \mathcal{P} , and is computed for each pressure image taxel based on the distance that the human mesh sinks into the bed. The reconstructed contact map (\hat{C}_O) corresponds with the input contact map, \hat{C}_I , and is a binary contact map of \hat{Q} . See Appendix B for details.

Loss function. We train Mod1 in PressureNet with the following loss function, given $N = 24$ Cartesian joint positions and $S = 10$ body parameters:

$$\mathbb{L}_1 = \frac{1}{N\sigma_s} \sum_{j=1}^N \|s_j - \hat{s}_{j,1}\|_2 + \frac{1}{S\sigma_\beta} \|\beta - \hat{\beta}_1\|_1 \quad (8)$$

where $s_j \in \mathcal{S}$ represents the 3D position of a single joint, and σ_s and σ_β are standard deviations computed over the whole dataset to normalize the terms.

In our evaluations (Section 6), sequentially training two separate network modules improved model performance and the resulting human mesh and pose predictions. For a pressure array of T taxels, we compute a loss for Mod2 by adding the error between the reconstructed pressure maps and the ground truth maps from simulation.

$$\mathbb{L}_2 = \mathbb{L}_1 + \frac{1}{T\sigma_Q} \|Q - \hat{Q}_2\|_2^2 + \frac{1}{T\sigma_{C_O}} \|C_O - \hat{C}_{O,2}\|_1 \quad (9)$$

where \mathbb{L}_1 uses Mod2 estimates (i.e. $\hat{S}_2, \hat{\beta}_2$), Q and C_O are ground truth maps precomputed by setting $\hat{\Psi} = \Psi$, and σ_Q, σ_{C_O} are computed over the dataset.

5. Evaluation

To evaluate our methods, we trained our CNN on synthetic data and tested it on both synthetic and real data. We generated 206K synthetic bodies at rest with corresponding pressure images (184K train / 22K test), which we partitioned to represent both a uniformly sampled space and common resting postures. By posture, we mean common recognized categories of overall body pose, such as sitting, prone, and supine. We tested 4 network types and 2 training data sets of different size.

5.1. PressurePose Data Partitions

We used the rejection sampling method described in Section 3 and Appendix A.1 to generate initial poses and create dataset partitions. Our main partition, the *general* partition, consists of 116K image and label pairs. In it, we evenly distributed limb poses about the Cartesian space and randomly sampled over body roll and yaw. This partition includes supine, left/right lateral and prone postures, as well as postures in between, and has the greatest diversity of poses. We also created a *general supine* partition (58K) featuring only supine postures and evenly distributed limb poses. Finally, we generated smaller partitions representing other common postures: *hands behind the head* (5K), *prone with hands up* (9K), *supine crossed legs* (9K), and *supine straight limbs* (9K). See Appendix A.7 for details.

5.2. PressureNet Evaluation

We normalized all input data by a per-image sum of taxels. We blurred synthetic and real images with a Gaussian of $\sigma = 0.5$. We trained for 100 epochs on Mod1 with loss function \mathbb{L}_1 . Then, we pre-computed the reconstruction maps $\{\hat{Q}_1, \hat{C}_{O,1}\}$ from Mod1 for input to Mod2, and trained Mod2 for 100 epochs using loss function \mathbb{L}_2 . See Appendix B.3 for training hyperparameters and details.

We investigated 5 variants of PressureNet, which are all trained entirely with synthetic data in order to compare the effect of (1) ablating PMR, (2) adding noise to the synthetic training data, (3) ablating the contact and edge input (C_I and \mathcal{E}), and (4) reducing the training data size. Ablating PMR consists of removing the 2 reconstructed maps from the input to Mod2 and using \mathbb{L}_1 for training both Mod1 and Mod2. We compared the effect of adding noise to the training data to account for real-world variation, such as sensor noise. Our noise model includes per-pixel white noise, additive noise, multiplicative noise, and blur variation, all with $\sigma = 0.2$. We compared networks trained on 46K vs. 184K images.

5.3. Human Participant Study

We mounted a Microsoft Kinect 2 1.6m above our Invacare Homecare bed to capture RGB images and point clouds synchronized with our pressure image data. See details in Appendix A.6. We recruited 20 (10F/10M) human participants with approval from an Institutional Review Board. We conducted the study in two parts to capture (1) participant-selected poses and (2) prescribed poses from the synthetic test set. We began by capturing five participant-selected poses. For the first pose, participants were instructed to get into the bed and get comfortable. For the remaining four, participants were told to get comfortable in supine, right lateral, left lateral, and prone postures. Next, for the prescribed poses, we displayed a pose rendering on a monitor, and instructed the participants to get into

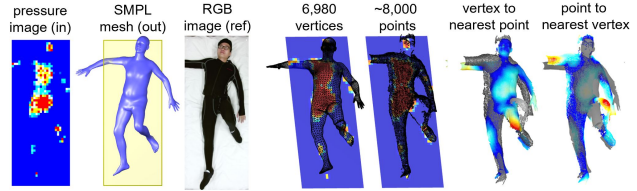


Figure 7. 3D error analysis between a human mesh (6,980 vertices) and a point cloud (~8,000 downsampled points).

the pose shown. We captured 48 prescribed poses per participant, which were sampled without replacement from the synthetic testing set: 24 general partition poses, 8 supine-only poses, and 4 from each of the remaining partitions.

5.4. Data Analysis

We performed an error analysis as depicted in Fig. 7. For this analysis, we compute the closest point cloud point to each mesh vertex, and the closest mesh vertex to each point cloud point. We introduce 3DVPE (3D vertex-point-error), which is the average of these numbers. We downsample the point cloud to a resolution of 1cm so the number of points is roughly equal to the number of mesh vertices. We clip the mesh vertices and the point cloud at the edges of the pressure mat. The point cloud only contains information from the top surface of the body facing the camera, so we clip the mesh vertices that do not have at least one adjacent face facing the camera. Finally, we normalize the mesh by vertex density: while the density of the point cloud is uniform from downsampling, the mesh vertices are highly concentrated in some areas like the face. We normalize each per-vertex error by the average of its adjacent face surface areas.

We evaluated PressureNet on the synthetic test set and compared the results to the real test set. We clip the estimated and ground truth mesh vertices and normalize per-vertex error in the same way as the real data. Additionally, we evaluated per-joint error (24 joints) using mean-per-joint-position error (MPJPE), and per-vertex error (6890 vertices) using vertex-to-vertex error (v2v) for the synthetic data. We evaluated the network’s ability to infer posture using the participant-selected pose dataset by manually labeling the inferred posture (4 labels: supine, prone, left/right lateral). We also compared to a baseline human, *BL*, where we put a body of mean shape in a supine position in the center of the bed and compare it to all ground truth poses. We positioned the legs and arms to be straight and aligned with the length of the body.

6. Results and Discussion

Overall, we found that using more synthetic data resulted in higher performance in all tests, as shown in Table 2. As expected, ablating the PMR network and ablating noise reduced performance. Ablating contact and edge inputs also reduced performance. We expect that comparable perfor-

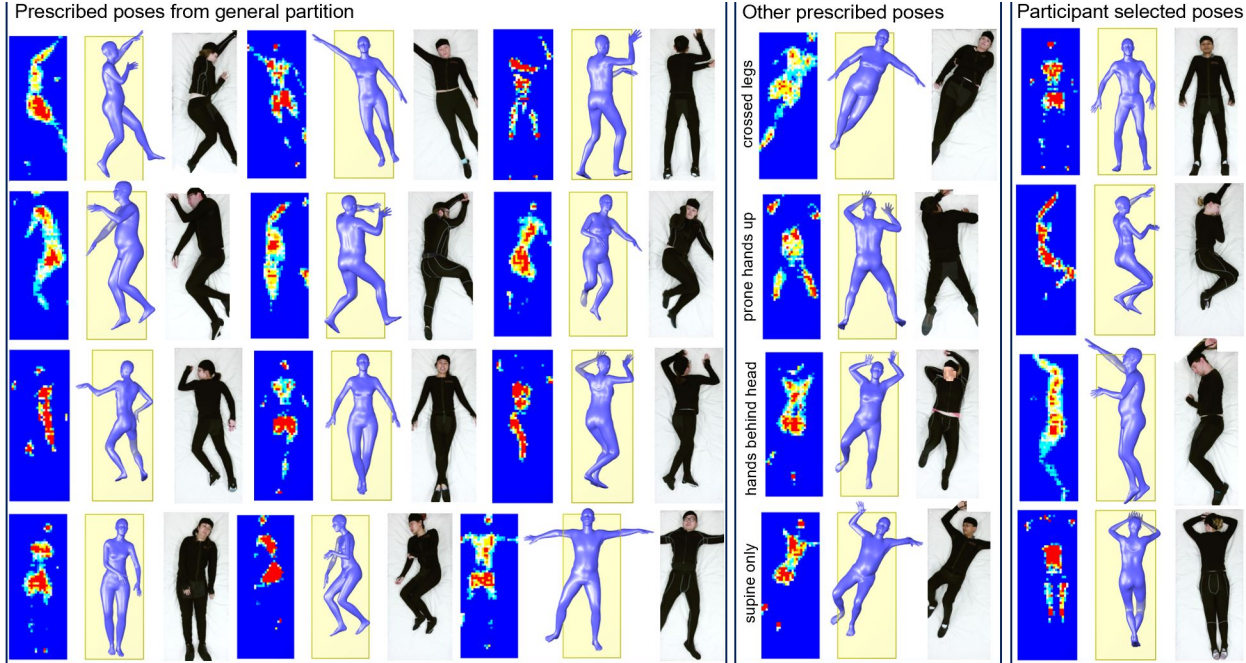


Figure 8. PressureNet results on real data with the best performing network (trained with 184K samples).

Network Description	Training data ct.	12K synth MPIPE (cm)	12K synth v2v (cm)	12K synth 3DVPE (cm)	1K real 3DVPE (cm)	99 real 3DVPE (cm)
Best	184K	11.18	13.50	3.94	4.99	4.76
Noise σ ablated	184K	11.18	13.52	3.97	5.05	4.79
Input $\mathcal{E}_t, \mathcal{C}_l$ ablated	184K	11.39	13.73	4.03	5.07	4.85
Best - small data	46K	12.65	15.28	4.35	5.17	4.89
PMR ablated	184K	12.28	14.65	4.38	5.33	4.94
Baseline - mean pose	-	33.30	38.70	8.43	6.65	5.22

Table 2. Results comparing testing data and network type.

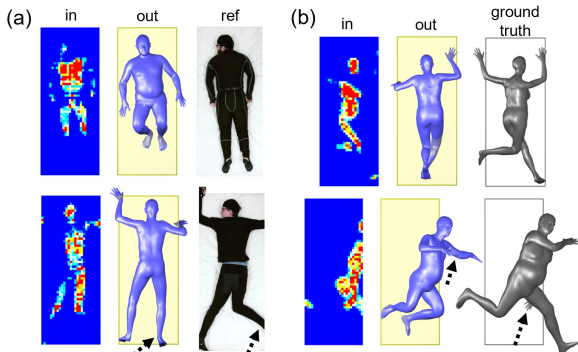


Figure 9. Some failure cases. (a) Real data. (b) Testing on synthetic training data.

mance could be achieved without them, possibly by changing the details of the CNN. Fig. 8 shows results from the best performing network with 184K training images, noise, and the PMR network.

We compared the error on a set of 99 participant selected poses, shown in Table 3, using the best performing PressureNet. Results show a higher error for lateral postures where the body center of mass is further from the mat

posture partition	test ct.	3DVPE (cm)	posture match
no instruction	19	3.93	100%
supine	20	4.02	100%
right lateral	20	5.45	100%
left lateral	20	5.37	100%
prone	20	4.96	95%*

Table 3. Results - participant selected poses. *See Fig. 9-top left.

and limbs are more often resting on other limbs or the body rather than the mat. Results on partitioned subsets of data can be found in Appendix B.4. Fig. 9 shows four failure cases.

7. Conclusion

With our physics-based simulation pipeline, we generated a dataset, PressurePose, consisting of 200K synthetic pressure images with an unprecedented variety of body shapes and poses. Then, we trained a deep learning model, PressureNet, entirely on synthetic data. With our best performing model, we achieve an average pose estimation error of < 5 cm, as measured by 3DVPE, resulting in accurate 3D pose and body shape estimation with real people on a pressure sensing bed.

Acknowledgement: We thank Alex Clegg. This work was supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE-1148903, NSF award IIS-1514258, NSF award DGE-1545287 and AWS Cloud Credits for Research.

Disclosure: Charles C. Kemp owns equity in and works for Hello Robot, a company commercializing robotic assistance technologies. Henry M. Clever is entitled to royalties derived from Hello Robot’s sale of products.

References

- [1] Felix Achilles, Alexandru-Eugen Ichim, Huseyin Coskun, Federico Tombari, Soheyl Noachtar, and Nassir Navab. Patient mocap: human pose estimation under blanket occlusion for hospital monitoring applications. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 491–499. Springer, 2016. [2](#), [3](#)
- [2] Ankur Agarwal and Bill Triggs. Recovering 3d human pose from monocular images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(1):44–58, 2006. [2](#)
- [3] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. SCAPE: shape completion and animation of people. *ACM Transactions on Graphics*, 24(3):408–416, 2005. [2](#)
- [4] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J. Black. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *ECCV*, pages 561–578. Springer, 2016. [2](#), [5](#)
- [5] Donna C. Boone and Stanley P. Azen. Normal range of motion of joints in male subjects. *Journal of Bone and Joint Surgery*, 61(5):756–759, 1979. [3](#)
- [6] Samarth Brahmabhatt, Cusuh Ham, Charles C. Kemp, and James Hays. ContactDB: Analyzing and predicting grasp contact via thermal imaging. In *CVPR*, pages 8709–8719. IEEE, 2019. [3](#)
- [7] Adrian Bulat and Georgios Tzimiropoulos. Human pose estimation via convolutional part heatmap regression. In *ECCV*, pages 717–732, 2016. [3](#)
- [8] Joao Carreira, Pulkit Agrawal, Katerina Fragkiadaki, and Jitendra Malik. Human pose estimation with iterative error feedback. In *CVPR*, pages 4733–4742. IEEE, 2016. [3](#)
- [9] Leslie Casas, Nassir Navab, and Stefanie Demirci. Patient 3d body pose estimation from pressure imaging. *International Journal of Computer Assisted Radiology and Surgery*, pages 1–8, 2019. [1](#), [2](#), [3](#)
- [10] Kenny Chen, Paolo Gabriel, Abdulwahab Alasfour, Chenghao Gong, Werner K. Doyle, Orrin Devinsky, Daniel Friedman, Patricia Dugan, Lucia Melloni, Thomas Thesen, David Gonda, Shifteh Sattar, Sonya Wong, and Vikash Gilja. Patient-specific pose estimation in clinical environments. *IEEE Journal of Translational Engineering in Health and Medicine*, 6:1–11, 2018. [1](#), [2](#), [3](#)
- [11] Wenzheng Chen, Huan Wang, Yangyan Li, Hao Su, Zhenhua Wang, Changhe Tu, Dani Lischinski, Daniel Cohen-Or, and Baoquan Chen. Synthesizing training images for boosting human 3d pose estimation. In *Conference in 3D Vision*, pages 479–488. IEEE, 2016. [3](#)
- [12] Alexander Clegg, Wenhao Yu, Zackory Erickson, Jie Tan, C. Karen Liu, and Greg Turk. Learning to navigate cloth using haptics. In *International Conference on Intelligent Robots and Systems*, pages 2799–2805. IEEE/RSJ, 2017. [3](#)
- [13] Henry M. Clever, Ariel Kapusta, Daehyung Park, Zackory Erickson, Yash Chitalia, and Charles C. Kemp. 3D human pose estimation on a configurable bed from a pressure image. In *IROS*, pages 54–61. IEEE, 2018. [1](#), [2](#), [3](#), [6](#)
- [14] Vandad Davoodnia, Saeed Ghorbani, and Ali Etemad. In-bed pressure-based pose estimation using image space representation learning. *arXiv preprint arXiv:1908.08919*, 2019. [3](#)
- [15] Zackory Erickson, Henry M. Clever, Greg Turk, C. Karen Liu, and Charles C. Kemp. Deep haptic model predictive control for robot-assisted dressing. In *International Conference on Robotics and Automation (ICRA)*, pages 1–8. IEEE, 2018. [3](#)
- [16] Zackory Erickson, Vamsee Gangaram, Ariel Kapusta, C. Karen Liu, and Charles C. Kemp. Assistive gym: A physics simulation framework for assistive robotics. In *International Conference on Robotics and Automation (ICRA)*. IEEE, 2020. [3](#)
- [17] M Farshbaf, Rasoul Yousefi, M Baran Pouyan, Sarah Ostadabbas, Mehrdad Nourani, and Matthew Pompeo. Detecting high-risk regions for pressure ulcer risk assessment. In *BIBM*, pages 255–260. IEEE, 2013. [1](#), [3](#)
- [18] Robert Grimm, Sebastian Bauer, Johann Sukkau, Joachim Hornegger, and Günther Greiner. Markerless estimation of patient orientation, posture and pose using range and pressure imaging. *International journal of computer assisted radiology and surgery*, 7(6):921–929, 2012. [1](#), [2](#), [3](#)
- [19] Ahsan Habib, Isura Ranatunga, Kyle Shook, and Dan O. Popa. Skinsim: A simulation environment for multimodal robot skin. In *International Conference on Automation Science and Engineering*, pages 1226–1231, 2014. [3](#)
- [20] Nikolaus Hansen, Youhei Akimoto, and Petr Baudis. CMA-ES/pycma on Github. Zenodo, DOI:10.5281/zenodo.2559634, Feb. 2019. [5](#)
- [21] Tatsuya Harada, Taketoshi Mori, Yoshifumi Nishida, Tomohisa Yoshimi, and Tomomasa Sato. Body parts positions and posture estimation system based on pressure distribution image. In *ICRA*, volume 2, pages 968–975. IEEE, 1999. [3](#)
- [22] Tatsuya Harada, Tomomasa Sato, and Taketoshi Mori. Pressure distribution image based human motion tracking system using skeleton and surface integration model. In *ICRA*, volume 4, pages 3201–3207. IEEE, 2001. [1](#), [2](#), [3](#)
- [23] Mohamed Hassan, Vasileios Choutas, Dimitrios Tzionas, and Michael J. Black. Resolving 3d human pose ambiguities with 3d scene constraints. In *ICCV*. IEEE, 2019. [3](#)
- [24] Yana Hasson, Gul Varol, Dimitrios Tzionas, Igor Kalevatykh, Michael J. Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *CVPR*, pages 11807–11816, 2019. [3](#)
- [25] Brayden Hollis, Stacy Patterson, Jinda Cui, and Jeff Trinkle. Bubbletouch: A quasi-static tactile skin simulator. In *Artificial Intelligence for Human-Robot Interaction*. AAAI, 2018. [3](#)
- [26] Invacare. 5410IVC Full Electric Homecare Bed. www.invacare.com/cgi-bin/imhqprd/inv_catalog/prod_cat_detail.jsp?s=0&prodID=5410IVC, Last accessed on 2020-02-27. [2](#)
- [27] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *CVPR*, pages 7122–7131. IEEE, 2018. [2](#), [3](#), [6](#)
- [28] Jeongseok Lee, Michael X. Grey, Sehoon Ha, Tobias Kunz, Sumit Jain, Yuting Ye, Siddhartha S. Srinivasa, Mike Stilman, and C. Karen Liu. Dart: Dynamic animation and

- robotics toolkit. *Journal of Open Source Software*, 3(22), 2018. 3, 4
- [29] Jason J Liu, Ming-Chun Huang, Wenyao Xu, and Majid Sarrafzadeh. Bodypart localization for pressure ulcer prevention. In *EMBC*, pages 766–769. IEEE, 2014. 1, 2, 3
- [30] Shuangjun Liu and Sarah Ostadabbas. Seeing under the cover: A physics guided learning approach for in-bed pose estimation. In *Medical Image Computing and Computer Assisted Intervention*, pages 236–245. Springer, 2019. 2, 3
- [31] Shuangjun Liu, Yu Yin, and Sarah Ostadabbas. In-bed pose estimation: Deep learning with shallow dataset. *IEEE Journal of Translational Engineering in Health and Medicine*, 7:1–12, 2019. 2, 3
- [32] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. Smpl: A skinned multi-person linear model. *ACM Transactions on Graphics*, 34(6):248, 2015. 2, 3, 4
- [33] Miles Macklin, Matthias Müller, Nuttapong Chentanez, and Tae-Yong Kim. Unified particle physics for real-time applications. *ACM Transactions on Graphics*, 33(4), 2014. 3, 4
- [34] Vista Medical. BodiTrak pressure mapping system solutions. www.boditrak.com/products/medical.php, Last accessed on 2020-02-27. 2
- [35] Ian Millington. *Game physics engine development: how to build a robust commercial-grade physics engine for your game*. CRC Press, 2010. 1
- [36] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hour-glass networks for human pose estimation. In *ECCV*, pages 483–499. Springer, 2016. 3
- [37] Ryuzo Okada and Stefano Soatto. Relevant feature selection for human pose estimation and localization in cluttered images. In *ECCV*, pages 434–445. Springer, 2008. 2
- [38] Sarah Ostadabbas, Maziyar Baran Pouyan, Mehrdad Nourani, and Nasser Kehtarnavaz. In-bed posture classification and limb identification. In *BioCAS*, pages 133–136. IEEE, 2014. 3
- [39] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. Osman, D. Tzionas, and M. J. Black. Expressive body capture: 3d hands, face, and body from a single image. In *CVPR*, pages 10975–10985. IEEE, 2019. 2
- [40] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G. Derpanis, and Kostas Daniilidis. Coarse-to-fine volumetric prediction for single-image 3d human pose. In *CVPR*. IEEE, 2017. 2
- [41] Shahram Payandeh and Naoufel Azouz. Finite elements, mass-spring-damper systems and haptic rendering. In *ICRA*, pages 224–229. IEEE. 5
- [42] Zachary Pezzementi, Erica Jantho, Lucas Estrade, and Gregory D. Hager. Characterization and simulation of tactile sensors. In *IEEE Haptics Symposium*, pages 199–205, 2017. 3
- [43] Anurag Ranjan, Javier Romero, and Michael J. Black. Learning human optical flow. In *British Machine Vision Conference*. BMVA Press, 2018. 3
- [44] Asbjørn Roaas and Gunnar BJ Andersson. Normal range of motion of the hip, knee and ankle joints in male subjects, 3040 years of age. *Acta Orthopaedica Scandinavica*, 53(2):205–208, 1982. 3
- [45] Nikolaos Sarafianos, Bogdan Boteanu, Catalin Ionescu, and Ioannis A. Kakadiaris. 3d human pose estimation: A review of the literature and analysis of covariates. *Computer Vision and Image Understanding*, (152):1–20, 2016. 2
- [46] Jamie Shotton, Andrew Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, and Andrew Blake. Real-time human pose recognition in parts from single depth images. In *CVPR*, pages 1297–1304, 2011. 2
- [47] J. M. Soucie, C. Wang, A. Forsyth, S. Funk, M. Denny, K. E. Roach, D. Boone, and Hemophilia Treatment Center Network. Range of motion measurements: reference values and a database for comparison studies. *Haemophilia*, 17(3):500–507, 2011. 3
- [48] Jonathan J. Tompson, Arjun Jain, Yann LeCun, and Christoph Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In *Advances in neural information processing systems*, pages 1799–1807, 2014. 2
- [49] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *CVPR*, pages 1653–1660, 2014. 2, 3
- [50] Aydin Tözeren. *Human Body Dynamics: Classical Mechanics and Human Movement*. Springer, 1999. 5
- [51] Gul Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J. Black, Ivan Laptev, and Cordelia Schmid. Learning from synthetic humans. In *CVPR*, pages 109–117, 2017. 3
- [52] Tao Yu, Zerong Zheng, Yuan Zhong, Jianhui Zhao, Qionghai Dai, Gerard Pons-Moll, and Yebin Liu. Simulcap: Single-view human performance capture with cloth simulation. In *CVPR*, pages 5499–5509. IEEE, 2019. 3
- [53] Xingyi Zhou, Xiao Sun, Wei Zhang, Shuang Liang, and Yichen Wei. Deep kinematic pose regression. In *ECCV 2016 Workshops*, pages 186–201, 2016. 2