PhysPT: Physics-aware Pretrained Transformer for Estimating Human Dynamics from Monocular Videos

Supplementary Material

In this supplementary material, we first provide additional details of our proposed approach:

- Section A: Global Trajectory Estimation
- Section B: Creation of Phys-SMPL, which includes (1) the Calculation of Body Part Volume, Mass, and Inertia Tensor; and (2) the Derivation of the Physical Parameters in the Euler-Lagrange Equations
- Section C: Selection of Body Contact Regions

We then present additional evaluation results:

- Section D: Improvements over Different Kinematics-based 3D Body Reconstruction Models
- Section E: Evaluation on Global Motion Recovery
- Section F: Quality of the Generated Force Labels
- Section G: Action-wise Recognition Performance

A. Global Trajectory Estimation

Traditional image or video-based 3D human body reconstruction models provide estimates of 3D body configuration in the body frame. Additionally, they estimate a root rotation that transforms the estimated 3D configuration from the body frame to the camera frame. To effectively model human dynamics, a human body motion trajectory represented in the world frame is needed. Besides local body movements, the global motion trajectory further involves the relative rotation between the camera frame and the world frame, and the body translation in the world frame. Inspired by the framework proposed by [94], we estimate these information through a global trajectory predictor with the model architecture illustrated in Figure 6. The input of the model are 3D body joint positions represented in the body frame $\{\mathbf{J}_t\}_{t=1}^{T}$. A Spatial-Temporal Graph Convolutional Networks (ST-GCN) [86] are then employed to extract spatial-temporal features for every time frame as:

$$\{\mathbf{h}_t\}_{t=1}^{T} = \mathtt{GCN}(\{\mathbf{J}_t\}_{t=1}^{T}), \tag{10}$$

where $\mathbf{h}^t \in \mathbb{R}^{256}$. Typically, the camera pose remains constant throughout a video. The model thus estimates a single rotation matrix for all frames. Specifically, the extracted features $\{\mathbf{h}_t\}_{t=1}^{T}$ are concatenated together and then input to a multi-layer perceptrons (MLP). The MLP module includes three fully connected layers with the ReLU activation functions [54] to predict the rotation transformation between the camera and world frame as:

$$\mathbf{R}_c = \mathtt{MLP}(\{\mathbf{h}_t\}_{t=1}^{T}). \tag{11}$$

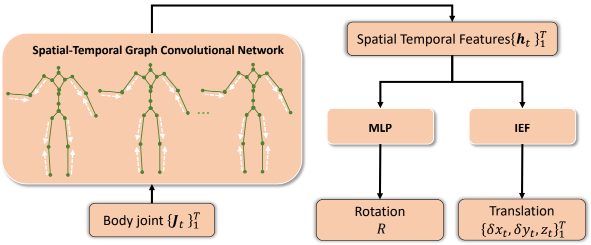
Spatial-Temporal Graph Convolutional Network
Spatial Temporal Features$\{\boldsymbol{h}_t\}_1^T$
MLP
IEF
Body joint $\{\boldsymbol{J}_t\}_1^T$
Rotation
R
Translation
$\{\delta x_t, \delta y_t, z_t\}_1^T$

Figure 6. **Global Trajectory Estimation Model.** "MLP" and "IEF" represents Multi-layer Perceptrons and Iterative Error Feedback regression model [7], respectively.

Combining the estimated $\mathbf{R}_c$ with the root rotation generated by the kinematics-based 3D human body reconstruction model at each frame leads to the global rotation $\{\mathbf{R}_t\}_{t=1}^{T}$ defined in Eq. 1. Note that the prediction of the camera rotation can be readily extended to predicting rotations for every frame considering a moving camera. Furthermore, to estimate the global translation, we employ a regression model with Iterative Error Feedback (IEF) [7] with 3 iterations. The employed regression model takes as input the exacted spatial-temporal features and outputs the 3D root joint positions represented in the world frame as:

$$\delta_{x,t}, \delta_{y,t}, z_t = \mathtt{IEF}(\mathbf{h}_t), \tag{12}$$

where $\delta_{x,t}$ and $\delta_{y,t}$ are position changes in the horizontal directions at time frame t, and z_t is the corresponding vertical position relative to the ground plane. Adding the position changes $(\delta_{x,t}, \delta_{y,t})$ over time and combining them with the vertical position z_t at different time frames, we obtain the final global translation $\{\mathbf{T}_t\}_{t=1}^{T}$ required to specify the generalized positions defined in Eq. 1.

For training of the global trajectory predictor, we use AMASS [51]. The training loss consists of the mean square errors between the predicted and the ground truth rotation and translation. The 3D motion sequences in AMASS are 3D trajectories represented in the world frame and only vary in the body translation. We hence introduce random rotation changes to the input to allow the model to predict the rotation changes. Meanwhile, we add random Gaussian noise to the input 3D joint positions to improve the model robustness. During training, we utilize the Adam optimizer [30] with a weight decay of 10^{-4}. The initial learning rate is 10^{-3} and decreases to its 0.95 after every 15,000 steps. The total number of training epochs is 20. Note that the training of the global trajectory predictor is independent to certain 3D human body reconstruction models. Once the model is trained, we directly combine it with the 3D reconstruction model to generate the initial generalize positions $\{\hat{\mathbf{q}}_t\}_{t=1}^{T}$.

B. Creation of Phys-SMPL

Phys-SMPL characterizes the necessary physical properties of the human body required in modeling human dynamics through the Euler-Lagrange equations. Specifically, these physical information includes body mass and inertia of different body parts and is utilized in computing the physical terms in the Euler-Lagrange equations. In this section, we first introduce the way of calculating the physics information based on the geometry information provided by SMPL. Then, we present the analytical equations for computing the physical parameters in the Euler-Lagrange Equations.

Calculation of Body Part Volume, Mass, and Inertia Tensor. The original SMPL builds upon 3D triangle mesh models. The 3D triangle mesh model captures the body geometry information but not physics. To effectively model physical properties from the geometry information, we first compute the volume of different body parts. Specifically, we build a close mesh for each body part by closing the mesh along the boundary. Then, each mesh triangle in a body part combined with the body part center can form a 3D tetrahedron, the volume of which can be easily computed. Meanwhile, for each body part, its volume can be computed as the sum of the volumes of all the 3D tetrahedra belonging to that body part. Defining the origin of a body part as its geometric center, the volume of body part i can thus be computed as:

$$V_i = \sum_{j=1}^{n_j} |\det(\mathbf{P}_{i,j,1}, \mathbf{P}_{i,j,2}, \mathbf{P}_{i,j,3})|, \tag{13}$$

where n_j is the total number of mesh triangles included in the i^{th} body part, $\mathbf{P}_{i,j,1}$, $\mathbf{P}_{i,j,2}$, and $\mathbf{P}_{i,j,3}$ are the 3D vertex position of the j^{th} triangle.

Using the computed volume, we can then determine the mass of each body part. Specifically, for the mean shape of SMPL, we consider it has a total body mass of 70 kg following the typical setting [84]. We compute the mass of each body part by distributing the total body mass based on the average human body weight distribution [56]. For subjects with different body shapes, we calculate their body mass based on the proportion of their body part volume relative to the mean shape. As human tissue exhibits varying mass density, instead of scaling the mass solely based on the body part volume, we consider the density differences between bone, muscle, and fat to introduce additional predefined scaling factors for different body parts following the study shown in [57].

For the inertia tensor required to specify the Euler-Lagrange equations, we need to compute the inertia tensor for each body part relative to its root joint represented in the body part frame. In the following, we present the analytical equations to compute the inertia tensor $\mathbf{I}_i$ of the i^{th} body part without loss of the generality. Firstly, we consider a body part is a rigid solid body that has a uniform mass density. The entries in the inertia tensor can be denoted as

$$\mathbf{I}_i = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix}. \tag{14}$$

Following the standard way to compute the inertia tensor [70], an entry in $\mathbf{I}_i$ is computed as

$$\frac{m_i}{V_i} \iiint_{(x,y,z)\in\mathbf{S}_i} f(x,y,z)dxdydz, \tag{15}$$

where m_i, V_i, and $\mathbf{S}_i$ denotes the mass, volume, 3D integral region of the body part, respectively. Then,

$$f(x,y,z) = \begin{cases} y^2 + z^2 & \text{for } I_{xx} \\ x^2 + z^2 & \text{for } I_{yy} \\ x^2 + y^2 & \text{for } I_{zz} \\ -xy & \text{for } I_{xy} \text{ and } I_{yx} \\ -xz & \text{for } I_{xz} \text{ and } I_{xz} \\ -yz & \text{for } I_{yz} \text{ and } I_{yz} \end{cases}. \tag{16}$$

As a body part is represented as a triangulate mesh, the integral can be computed as the sum of the integral computed within each tetrahedron of the body part as

$$\frac{m_i}{V_i} \sum_{j=1}^{n_i} \iiint_{(x,y,z)\in\mathbf{S}_{i,j}} f(x,y,z)dxdydz, \tag{17}$$

where $\mathbf{S}_{i,j}$ is the 3D integral region of the j^{th} tetrahedron, and n_j is the total number of mesh triangles of the i body part. Given that each body part is assumed to have a uniform mass density, the Center of Mass (COM) coincides with the centroid of that body part. As shown in [27], the integral over the j^{th} tetrahedron of the i^{th} body part can be computed as

$$\tfrac{v_{i,j}}{20}(f(\mathbf{P}_{i,j,1}) + f(\mathbf{P}_{i,j,2}) + f(\mathbf{P}_{i,j,3}) + f(\mathbf{P}_{i,j,1} + \mathbf{P}_{i,j,2} + \mathbf{P}_{i,j,3})), \tag{18}$$

where $\mathbf{P}_{i,j,1}$, $\mathbf{P}_{i,j,2}$, and $\mathbf{P}_{i,j,3}$ represent the 3D vertex positions of the j^{th} tetrahedron in the body frame with regards to the COM.

As introduced above, the calculation of the body mass and inertia tensor is based on the study of human anatomy and is accomplished using SMPL's geometry information. This approach avoids the use of unrealistic proxy bodies and establishes a direct mapping between the physics information and the shape parameters.

Derivation of the Physical Parameters in the Euler-Lagrange Equations. The physical parameters in the Euler-Lagrange equations are functions of the generalized positions, as well as the body mass and inertia. In this section, we present analytical equations to compute these physical parameters, including the contact Jacobian $\mathbf{J}_C$, the generalized mass matrix $\mathbf{M}\big(\mathbf{q};\ \mathbf{m}, \mathbf{I}\big)$, the gravitational force $\mathbf{g}\big(\mathbf{q}; \mathbf{m}, \mathbf{I}\big)$, and the generalized bias force $\mathbf{C}\big(\mathbf{q}, \dot{\mathbf{q}};\ \mathbf{m}, \mathbf{I}\big)$.

For the contact Jacobian, it is the Jacobian matrix that maps a contact force represented in the Cartesian coordinates to the generalized coordinates. The contact Jacobian matrix only relates to the position of a contact point. For the human body, it can receive multiple contact forces. Below we derive the contact Jacobian matrix applied to one contact position C without loss of generality.

As shown in [35], the contact Jacobian can be computed as,

$$\mathbf{J}_C = \begin{bmatrix} \mathbf{e}_x & \mathbf{e}_y & \mathbf{e}_z & \boldsymbol{\xi}_i & \cdots \end{bmatrix}, \tag{19}$$

where $\mathbf{e}_x$, $\mathbf{e}_y$, and $\mathbf{e}_z$ are the unit vectors along the x, y, and z directions, respectively. $\mathbf{e}_x$, $\mathbf{e}_y$, and $\mathbf{e}_z$ correspond to the global translation defined in the generalized position $\mathbf{q}$. For the other columns corresponding to joint rotations, they are computed as

$$\boldsymbol{\xi}_i = \begin{cases} {}_{\mathcal{A}}\mathbf{h}_i \times {}_{\mathcal{A}}\mathbf{r}_{iC}, & \text{if } i \text{ is a parent joint,} \\ \mathbf{0}, & \text{otherwise.} \end{cases} \tag{20}$$

where ${}_{\mathcal{A}}\mathbf{h}_i$ is the rotation axis of $\mathbf{q}_i$ represented in the world frame $\mathcal{A}$, and is computed as

$${}_{\mathcal{A}}\mathbf{h}_i = \mathbf{R}_{0i-1}{}_{\mathcal{B}}\mathbf{h}_i. \tag{21}$$

$\mathbf{R}_{0i-1}$ is the rotation matrix describing the transformation from the body part frame to the world frame. Moreover, ${}_{\mathcal{A}}\mathbf{r}_{iC}$ is the 3D position of the contact point relative to its root joint i represented in the world frame. ${}_{\mathcal{A}}\mathbf{r}_{iC}$ is computed as

$${}_{\mathcal{A}}\mathbf{r}_{iC} = \mathbf{r}_C - \mathbf{P}_i, \tag{22}$$

where $\mathbf{r}_C$ and $\mathbf{P}_i$ are 3D positions of the contact point and the root joint represented in the world frame, respectively.

For the generalized mass matrix $\mathbf{M}\big(\mathbf{q};\ \mathbf{m}, \mathbf{I}\big)$, it is a function of the body part mass $\mathbf{m}$, inertia $\mathbf{I}$, and the generalized position $\mathbf{q}$. The generalized mass matrix can be computed as the sum of the generalized mass of each individual body part as:

$$\mathbf{M}\big(\mathbf{q};\ \mathbf{m}, \mathbf{I}\big) = \sum_{n=1}^{24} \mathbf{J}_{S,n}^T m_n \mathbf{J}_{S,n} + \mathbf{J}_{R,n}^T \mathbf{R}_{0,n} \mathbf{I}_n \mathbf{R}_{0,n}^T \mathbf{J}_{R,n} \tag{23}$$

where $\mathbf{J}_{S,n}$ is the Jacobian matrix computed in the world frame. The calculation of $\mathbf{J}_{S,n}$ is similar to the contact Jacobian matrix but considers the root joint of a body part as the target position to computed the Jacobian matrix. Moreover, $\mathbf{R}_{0,n}$ is the pose of the n^{th} body part, and $\mathbf{J}_{R,n}$ is the angular Jacobian computed as

$$\mathbf{J}_{R,n} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{\xi}_i & \cdots & \mathbf{0} \end{bmatrix}, \tag{24}$$

where

$$\boldsymbol{\xi}_i = \begin{cases} {}_{\mathcal{A}}\mathbf{h}_i, & \text{if } i \text{ is a parent joint,} \\ \mathbf{0}, & \text{otherwise.} \end{cases} \tag{25}$$

For the gravitational force term, it is computed as

$$\mathbf{g}\big(\mathbf{q};\ \mathbf{m}, \mathbf{I}\big) = -\sum_{n=1}^{24} \mathbf{J}_{S,n}^T m_n \mathbf{g}, \tag{26}$$

where $\mathbf{J}_{S,n}$ is the Jacobian matrix of the n^{th} body part and $\mathbf{g}$ is the gravitational acceleration.

For the generalized bias force $\mathbf{C}\big(\mathbf{q}, \dot{\mathbf{q}};\ \mathbf{m}, \mathbf{I}\big)$, it is computed as

$$\begin{aligned} \mathbf{C}\big(\mathbf{q}, \dot{\mathbf{q}};\ \mathbf{m}, \mathbf{I}\big) = \sum_{n=1}^{24} & \mathbf{J}_{S,n}^T m_n \dot{\mathbf{J}}_{S,n} \dot{\mathbf{q}} && (27) \\ & + \mathbf{J}_{R,n}^T (\mathbf{R}_{0,n} \mathbf{I}_n \mathbf{R}_{0,n}^T \dot{\mathbf{J}}_{R,n} \dot{\mathbf{q}} && (28) \\ & + \mathbf{J}_{R,n} \dot{\mathbf{q}} \times \mathbf{R}_{0,n} \mathbf{I}_n \mathbf{R}_{0,n}^T \mathbf{J}_{R,n} \dot{\mathbf{q}}), && (29) \end{aligned}$$

where $\dot{\mathbf{J}}_{S,n}$ and $\dot{\mathbf{J}}_{R,n}$ are the time derivatives of the linear and angular Jacobian. Specifically,

$$\dot{\mathbf{J}}_{S,n} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \dot{\boldsymbol{\xi}}_i & \cdots \end{bmatrix}. \tag{30}$$

For the none-zero terms,

$$\begin{aligned} \dot{\boldsymbol{\xi}}_i = {} & \dot{\mathbf{R}}_{0i-1}{}_{\mathcal{B}}\mathbf{h}_i \times {}_{\mathcal{A}}\mathbf{r}_{iC} && (31) \\ & + \mathbf{R}_{0i-1}{}_{\mathcal{B}}\mathbf{h}_i \times {}_{\mathcal{A}}\dot{\mathbf{r}}_{iC} && (32) \end{aligned}$$

where $\dot{\mathbf{r}}_{iC}$ can be computed through finite difference. The time derivative of the rotation matrix is computed as

$$\dot{\mathbf{R}}_{0i-1} = (\mathbf{J}_{R,i-1}\dot{\mathbf{q}})^{\times} \mathbf{R}_{0i-1} \tag{33}$$

On the other hand, the time derivatives of the angular Jacobian can be computed similarly as

$$\dot{\mathbf{J}}_{R,n} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \dot{\mathbf{R}}_{0i-1}{}_{\mathcal{B}}\mathbf{h}_i & \cdots \end{bmatrix}. \tag{34}$$

In summary, Phys-SMPL computes the body mass and inertia information directly from SMPL specified by its shape parameters. The calculation of the physical terms in the Euler-Lagrange equations is also fully-differentiable, facilitating the seamless integration with physics and deep learning models. Additional background of the Euler-Lagrange equations can be found in [35].

C. Selection of Body Contact Regions

We consider that the human body receives contact forces primarily from the ground. To effectively and efficiently model the contact behavior, we first investigate the body regions that frequently contact with the ground. We employ the motion sequence in AMASS and count the frequency of

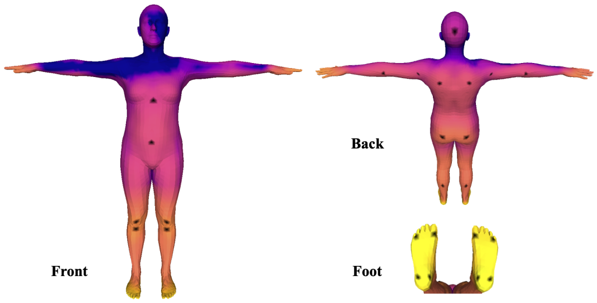

Figure 7. **Contact Map.** The lighter color (yellow) of a vertex indicates the higher frequency of contact with the ground. The modeled contact vertices are marked in black.

Method	*Rec. Error*		*Phys. Plausibility*				
	MJE	P-MJE	ACCL	VEL	FS	GP	BOS
SPIN [33]	66.2	40.8	18.1	8.2	8.8	12.5	23.7
+PoseBert [2]	64.3	41.8	5.3	4.1	9.2	15.7	26.1
+PhysPT (Ours)	**60.7**	**40.3**	**2.5**	**3.6**	**3.0**	**2.1**	**31.4**
IPMAN [76]	63.1	41.0	17.2	7.8	8.6	11.9	28.6
+PoseBert [2]	62.2	41.5	5.3	4.2	9.1	11.9	29.8
+PhysPT (Ours)	**59.4**	**40.2**	**2.5**	**3.6**	**3.0**	**2.3**	**36.1**

Table 5. **Evaluation on Human3.6M using Different Kinematics-based 3D Reconstruction Models.** The results of other works are from their officially released models. BOS is measured in percentages, with larger values indicating better performance, while the other metrics prefer smaller values.

the vertices in contact with the ground. We visualize the results in Figure 7. As expected, not all vertices have frequent contacts with the ground. On the other hand, modeling the contact for all vertices can result in significant computational overhead. Following existing approaches, we model a subset of vertices for each body part that frequently come into contact with the ground. The chosen vertices are highlighted by black colors in Figure 7.

D. Improvements over Different Kinematics-based 3D Body Reconstruction Models

PhysPT can be seamlessly applied to different kinematics-based models to enhance their motion estimates. To demonstrate this, besides employing CLIFF (evaluation is discussed in the main manuscript), we here report the improvements over two recent image-based 3D human body reconstruction methods, SPIN and IPMAN. SPIN and IPMAN are both model-based 3D human body reconstruction models that directly predict the 3D body pose and shape parameters from input images. IPMAN can produce more stable 3D body pose estimates than SPIN due to the incorporation of an intuitive-physics loss during training. To further demonstrate the effectiveness of PhysPT and compare with IPMAN, we follow IPMAN and compute the Base of Support (BOS) metric to evaluate the physical plausibility in terms of pose stability. We present the results in Table 5 and compare the improvements achieved by our approach with those produced by PoseBert. As demonstrated, when integrated with both SPIN and IPMAN, PhysPT effectively improves the reconstruction accuracy and substantially enhances the physical plausibility. For instance, when evaluating the reconstruction accuracy, adding PhysPT on top of SPIN shows a reduction in MJE from 66.2mm to 60.7mm, and on top of IPMAN, it experiences a decrease from 63.1mm to 59.4mm. In terms of physical plausibility, incorporating PhysPT with SPIN results in a reduction of 15.6 mm/frame^2 in ACCL, and with IPMAN, it exhibits a reduction of 14.7 mm/frame^2. The improvements achieved by PhysPT are also more pronounced than PoseBert. Furthermore, when assessing stability, SPIN initially exhibits poor performance with a lower BOS than IPMAN (23.7% vs. 28.6%). By integrating with PhysPT, SPIN can generate a higher BOS than IPMAN (31.4% over 28.6%). The improvements in stability are also evident when incorporating PhysPT with IPMAN. PhysPT significantly enhances the motion estimates by effectively leveraging the Transformer model and integrating physics principles and it applies to various kinematics-based 3D reconstruction models.

Method	Training on Human3.6M	G-MPJPE (↓)	G-MPVPE (↓)
D&D [39]	Yes	525.3	533.9
PhysPT (Ours)	**No**	**335.7**	**343.5**

Table 6. **Evaluation on Global Motion Recovery.** The evaluation is on the test set of Human3.6M. The units of G-MPJPE and G-MPVPE are in mm.

E. Evaluation on Global Motion Recovery

This section demonstrates the accurate global motion recovery achieved by our approach. We compute the Mean Per-Joint Position Errors and Mean Per-Vertex Position Errors in the world frame (G-MPJPE and G-MPVPE). Following the typical evaluation protocol [39, 94], we use a 10-second sliding window with the root translation aligned with the ground truth at the starting frame of each window. We report the results in Table 6 with comparison to the physics-based SOTA. As shown, our approach produces much lower errors than the SOTA. For example, our approach achieves 335.7 mm of G-MPJPE, reducing D&D's 525.3 by nearly 36%. Moreover, it's worth noting that our method achieves

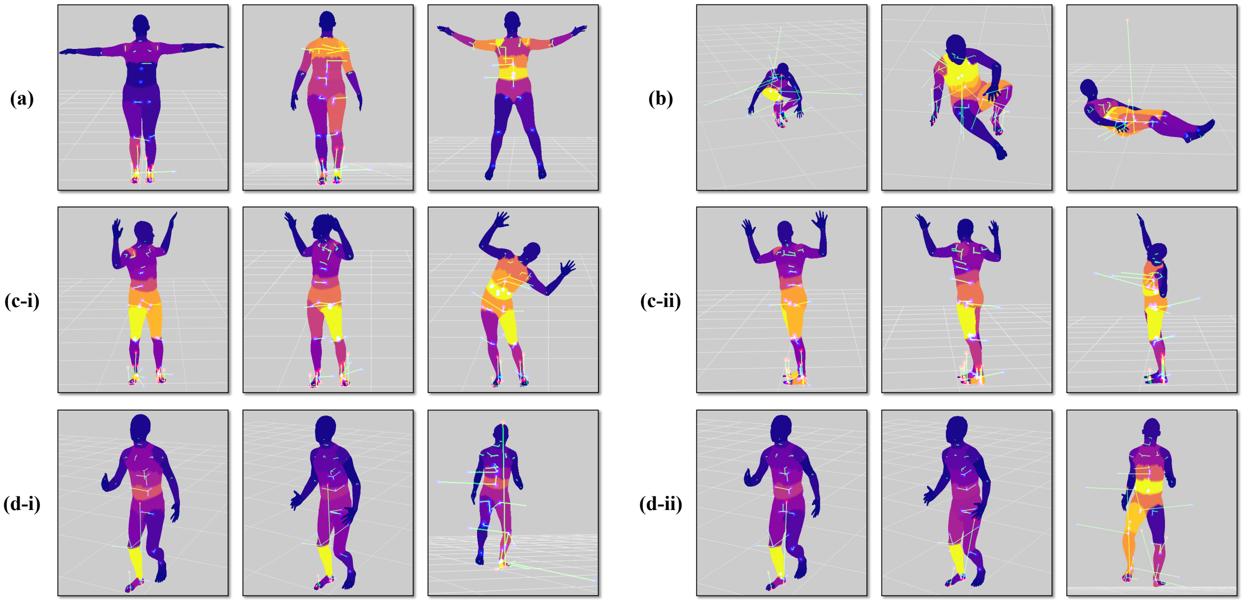

Figure 8. **Visualization of the Inferred Motion Forces.** The motion sequences (a), (b), (c), and (d) are from AMASS [51]. For sequence (c), (c-i) and (c-ii) are visualization of the same figure from different views. For sequence (d), (d-i) and (d-ii) are visualization of the forces generated with and without employing the continuous contact force model, respectively. In each image, the contact forces are visualized via a 3D vector on each contact point (green lines ended with red dots). The joint actuations are characterized by three vectors along the three Euler angles of a joint (green lines ended with blue dots). Meanwhile, the magnitudes of joint actuations are visualized by different colors at different body parts, with lighter colors indicating larger magnitudes.

better performance without utilizing any 3D data from Human3.6M during training.

F. Quality of the Inferred Motion Forces

The motion forces derived from the Euler-Lagrange equations using the physics-based body representation and the contact force model offer valuable insights into human dynamic behaviors. Utilizing them enables a more effective incorporation of the physics equations. We here discuss the quality of the inferred motion forces. In Figure 8, we present example forces generated from different motion sequences in AMASS. As illustrated, the inferred motion forces sensibly indicate the direction and magnitude of the underlying forces. For example, during normal standing (Figure 8-a, column 1), the contact forces are evenly distributed between both feet. When leaning to the left or right, the center of gravity shifts, and larger contact forces are displayed on the left or right foot accordingly (Figure 8-c, columns 1-2). Additionally, the contact forces applied to different body parts, such as those experienced on the feet and hips, are effectively modelled (Figure 8-b, columns 2-3). On the other hand, the inferred joint actuations clearly indicate the rotation direction and force magnitudes. For example, the spine joint actuation in the horizontal direction controls rotation along the horizontal axis. It changes direction when rotating the upper body from left to right (Figure 8-c, columns 1-2). Moreover, large forces are shown at the shoulder joints when extending the arms (Figure 8-a, columns 2-3), or at the spine joints when rotating the upper body (Figure 8-c, column 3).

To demonstrate the benefits of utilizing the continuous contact force model, we further compare the forces inferred with and without employing the contact force model. As illustrated in Figure 8-d, exploiting the contact force model results in a more stable estimation of the forces. Additionally, when not using the contact force model, a contact status must be determined beforehand in a heuristic manner. For example, a point is considered in contact if its distance to the ground is less than 3 cm, and its velocity is less than 1 m/s [90]. In contrast, utilizing the contact force model eliminates the need for estimating the contact status and directly infers the contact forces based on a spring-mass model. Utilizing the contact force model can avoid the problems caused by incorrect estimations of the contact status (Figure 8-d-ii, column 3).

G. Action-wise Recognition Performance

In the main manuscript, we demonstrate that combining motion and force estimates leads to the best model performance. In this section, we discuss the action-wise evalua-

Input	Baseball	Clean	Pull	Strum	Baseball	Golf	Push	Tennis	Bench	Jumping	Sit	Tennis	Bowling	Jump	Squats	All
	Pitch	&Jerk	Ups	Guitar	Swing	Swing	Ups	Forehand	Press	Jacks	Ups	Serve		Rope		
$\mathbf{J}_{phys}$	100.0	95.6	94.9	100.0	98.3	100.0	99.0	93.5	91.4	98.2	98.0	97.1	96.4	97.6	94.9	96.8
$\mathbf{F}$	100.0	93.3	97.0	97.8	96.6	97.4	98.1	81.8	94.3	96.4	90.0	92.9	88.1	97.6	95.9	94.4
$\mathbf{J}_{phys}$+$\mathbf{F}$	100.0	100.0	97.0	100.0	98.3	100.0	98.1	96.1	97.1	98.2	98.0	97.1	97.6	97.6	96.9	98.0

Table 7. **Action-wise Evaluation on PennAction Utilizing Different Model Inputs.** The numbers represent recognition accuracy in percentages. Actions that show higher accuracy when utilizing forces ($\mathbf{F}$) compared to utilizing the physics-based estimation of 3D body joint positions ($\mathbf{J}_{phys}$) are marked in green. The term “All” denotes the average accuracy over all actions.

tion results, providing further insights into the benefits of utilizing forces for understanding human behaviours.

Action-wise Evaluation. The action-wise evaluation results are summarized in Table 7. As shown, only using forces as model input produces a lower average recognition accuracy compared to utilizing the physics-based motion estimates. Nonetheless, utilizing forces yields higher accuracy for certain actions, such as “Pull Ups”, “Bench Press”, and “Squats”. These actions are distinctive particularly in their underlying motion forces. Specifically, ”Pull Ups” and “Bench Press” involve similar body movements, such as raising the arms with bending legs. However, their underlying motion forces are significantly different. “Pull Ups” involves body lifting, while “Bench Press” involves body lying on a bench. For these actions, the estimated forces provide additional insights towards the human dynamic behaviours, and when combined with the 3D position data, they can significantly improve the final recognition accuracy. For example, the recognition accuracy of “Bench Press” increases from 91.4% to 97.1% by further adding the forces as model input.

We here present the implementation details for reproducibility. **Implementation Details.** For our experiments on PennAction, we follow the established protocol that use the official split to divide the training and test sets. The skeleton graph is defined in the SMPL joint format. For the motion estimate input, they are 3D body joint positions. For the force input, they are estimated force values, where the contact forces are transformed into the generalized coordinates to align with the defined skeleton graph. When combining the motion and force estimates, we employ a decision-level fusion. The models are trained by minimizing the cross-entropy loss. We utilize the Adam optimizer with a weight decay of 10^{-4}. We train the models for 200 epochs with an initial learning rate of 10^{-4} and decreasing to its 0.8 after every 40 epochs. The batch size is 128.