RESEARCH-ARTICLE

SMPL-IK: Learned Morphology-Aware Inverse Kinematics for AI Driven Artistic Workflows

VIKRAM VOLETI, Montreal Institute for Learning Algorithms, Montreal, QC, Canada

BORIS NIKOLAI ORESHKIN, Unity Technologies, San Francisco, CA, United States

FLORENT BOCQUELET, Unity Technologies, San Francisco, CA, United States

FÉLIX G HARVEY, Unity Technologies, San Francisco, CA, United States

LOUIS SIMON MÉNARD, Unity Technologies, San Francisco, CA, United States

CHRISTOPHER J PAL, Montreal Institute for Learning Algorithms, Montreal, QC, Canada

.

SMPL-IK: Learned Morphology-Aware Inverse Kinematics for AI Driven Artistic Workflows

Vikram Voleti[*]
vikram.voleti@umontreal.ca
Mila, University of Montreal
Canada

Boris N. Oreshkin[*]
boris.oreshkin@gmail.com
Unity Technologies
Canada

Florent Bocquelet[*]
florent.bocquelet@unity3d.com
Unity Technologies
Canada

Félix G. Harvey
felix.harvey@unity3d.com
Unity Technologies
Canada

Louis-Simon Ménard
louissimon.menard@unity3d.com
Unity Technologies
Canada

Christopher Pal
christopher.pal@polymtl.ca
Mila, Polytechnique Montreal, CIFAR
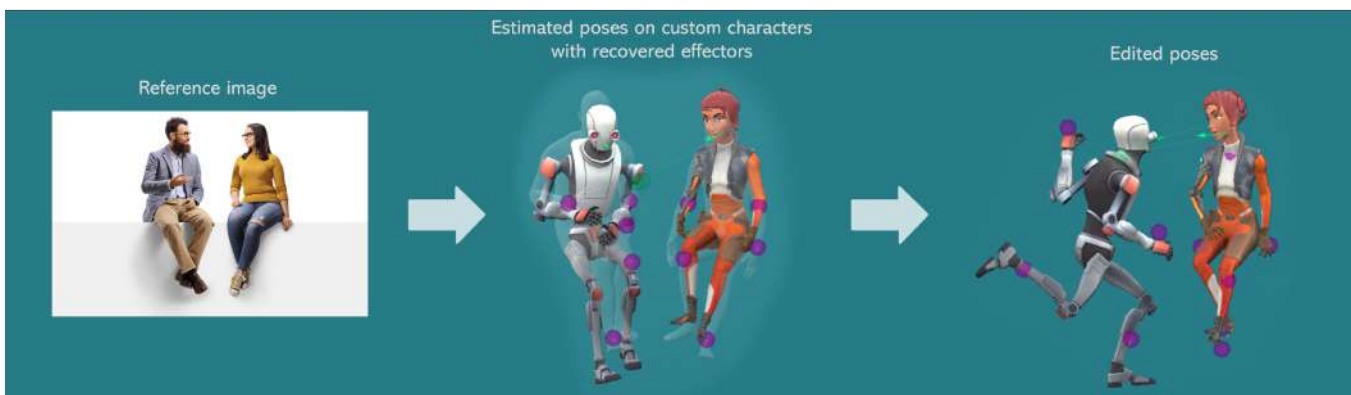Chair, ServiceNow
Canada

Figure 1: Our approach unlocks novel artistic workflows such as the one depicted above. Animator grabs one of the many pictures available on the web and uses it to initialize an editable 3D scene. Thus a multi-person 3D scene acquired from a RGB picture is populated with custom user-defined characters whose 3D poses are further edited with the state-of-the-art machine learning inverse kinematics tool integrated in a real-time 3D development software.

ABSTRACT

Inverse Kinematics (IK) systems are often rigid with respect to their input character, thus requiring user intervention to be adapted to new skeletons. In this paper we aim at creating a flexible, learned IK solver applicable to a wide variety of human morphologies. We extend a state-of-the-art machine learning IK solver to operate on the well known Skinned Multi-Person Linear model (SMPL). We call our model SMPL-IK, and show that when integrated into real-time 3D software, this extended system opens up opportunities for defining novel AI-assisted animation workflows. For example, when chained with existing pose estimation algorithms, SMPL-IK accelerates posing by allowing users to bootstrap 3D scenes from 2D images while allowing for further editing. Additionally, we propose a novel SMPL Shape Inversion mechanism (SMPL-SI) to map arbitrary humanoid characters to the SMPL space, allowing artists to leverage SMPL-IK on custom characters. In addition to qualitative demos showing proposed tools, we present quantitative SMPL-IK baselines on the H36M and AMASS datasets. Our code is publicly available https://github.com/boreshkinai/smpl-ik.

CCS CONCEPTS

• **Computing methodologies → Animation**.

KEYWORDS

SMPL, learned inverse kinematics, 3D animation, pose authoring

[*]Authors contributed equally to this research.

1 INTRODUCTION

Inverse Kinematics (IK) is the problem of estimating 3D positions and rotations of body joints given some end-effector locations [Aristidou et al. 2018; Kawato et al. 1993]. IK is an ill-posed nonlinear problem with multiple solutions. For example, given the 3D location of the right hand, what is a realistic human pose? It has been shown recently that machine learning IK model can be integrated with 3D content authoring user interface to produce a very effective pose authoring tool [Bocquelet et al. 2022; Oreshkin et al. 2021]. Using this tool, an animator provides a terse pose definition via a limited set of positional and angular constraints. The computer tool fills in the rest of the pose, minimizing pose authoring overhead.

The Skinned Multi-Person Linear (SMPL) model is a principled and popular way of jointly modelling human mesh, skeleton and pose [Loper et al. 2015]. It would seem natural to extend this model with inverse kinematics capabilities: making both human shape/mesh and pose editable using independent parameters. Additionally, many computer vision pose estimation algorithms naturally operate in the SMPL space making them natively compatible with a hypothetical SMPL IK model. This extension would open new content authoring opportunities. However, to date SMPL models have not been integrated with advanced machine learning IK tools, and this represents a clear research gap.

In our work we close this gap, exploring and solving two inverse problems in the context of the SMPL human mesh representation: SMPL-IK, an Inverse Kinematics model, and SMPL-SI, a Shape Inversion model. We show how these new components can be used to create new artistic workflows driven by AI algorithms. For example, we demonstrate the tool integrating SMPL-IK and SMPL-SI with an off-the-shelf image-to-pose model, initializing a multi-person 3D scene editable via flexible and easy-to-use user controls.

2 BACKGROUND

2.1 Inverse Kinematics

Inverse Kinematics is a prominent problem in robotics and animation, which is traditionally solved by analytical or iterative optimization methods comprehensively reviewed by Aristidou et al. [2018]. Solving IK using machine learning techniques has consistently attracted attention [Bócsi et al. 2011; De Angulo and Torras 2008; D'Souza et al. 2001], with more work focusing on neural networks based methods [Bensadoun et al. 2022; El-Sherbiny et al. 2018; Mourot et al. 2022]. In the animation space, the current neural IK state-of-the-art is ProtoRes [Oreshkin et al. 2021]. It takes a variable set of effector positions, rotations or look-at targets as inputs, and performs IK to reconstruct all joint locations and rotations. Its effectiveness in editing complex 3D character poses has been demonstrated in a real-time live demo [Bocquelet et al. 2022]. One limitation of ProtoRes is that, being trained on a fixed skeleton, it does not explicitly include any learnt body shape prior. In this work we relax this limitation by integrating ProtoRes with SMPL.

2.2 SMPL

SMPL is a realistic 3D human body model parameterized by the body's shape and pose based on skinning and blend shapes [Loper et al. 2015]. SMPL realistically represents a wide range of human body shapes controlled by shape parameters, as well as natural pose-dependent deformations controlled by pose parameters. SMPL remains a widely accepted model to represent realistic human body pose and is widely used for 3D pose estimation of humans in images and video [Bogo et al. 2016; Li et al. 2021; Luo et al. 2020; Rajasegaran et al. 2021; Sun et al. 2021].

2.3 Retargeting

Retargeting is the task of transferring the pose of a source character to a target character with a different morphology (bone lengths) and possibly a different topology (number of joints, connectivity, etc.) [Gleicher 1998]. Retargeting is a ubiquitous task in animation, and procedural tools exist for retargeting between skeletons of different morphologies and topologies [3D 2022].

2.4 SMPL and IK

SMPL and IK There is very little prior work that attempts to use the IK-enabled SMPL model for 3D character animation. Bebko et al. [2021] pose SMPL characters in the Unity platform, but do not perform any IK. Zhou [2020] performs IK on SMPL parameters using standard optimization, but only in the full pose context, which has very limited applicability for artistic pose editing. VPoser [Pavlakos et al. 2019] trains a Variational Auto-Encoder (VAE) to work as a prior on 3D human pose obtained from SMPL. This VAE is used as an iterative IK solver for a pose defined via keypoints. However, the VPoser architecture only works with relatively dense *positional* inputs (no ability to handle sparse heterogeneous effector scenarios has been demonstrated). It also requires on-line L-BFGS optimization, making it too rigid and computationally expensive for pose authoring. There is a clear gap between SMPL and IK: existing IK models suitable for artistic pose editing do not support SMPL, and existing SMPL-based models have insufficient IK cababilities.

3 SMPL-IK

We propose SMPL-IK, a learned morphology-aware inverse kinematics module that accounts for SMPL shape and gender information to compute the full pose including the root joint position and 3D rotations of all SMPL joints based on a partially-defined pose specified by SMPL β-parameters (body shape), gender flag, and a few input effectors (positions, rotations, or look-at targets). SMPL-IK supports effector combinations of arbitrary number and type. SMPL-IK extends the learned inverse kinematics model ProtoRes [Oreshkin et al. 2021]. ProtoRes only deals with a fixed morphology scenario in which an ML-based IK model is trained on a fixed skeleton. We remove this limitation by conditioning the ProtoRes computation on the SMPL β-parameters and gender (see Appendix F.1 for technical details). This results in an IK model that can operate on the wide range of morphologies incorporated in the expansive dataset used to create the SMPL model itself.

There are multiple advantages of this extension, including the following. First, rich public datasets can be used to train a learned IK model, in our case we train on the large AMASS dataset [Mahmood et al. 2019]. Second, an animator can now edit both the pose and the body shape of a flexible SMPL-based puppet using a state-of-the-art learned IK tool, which we demonstrate in Appendix A. Third,
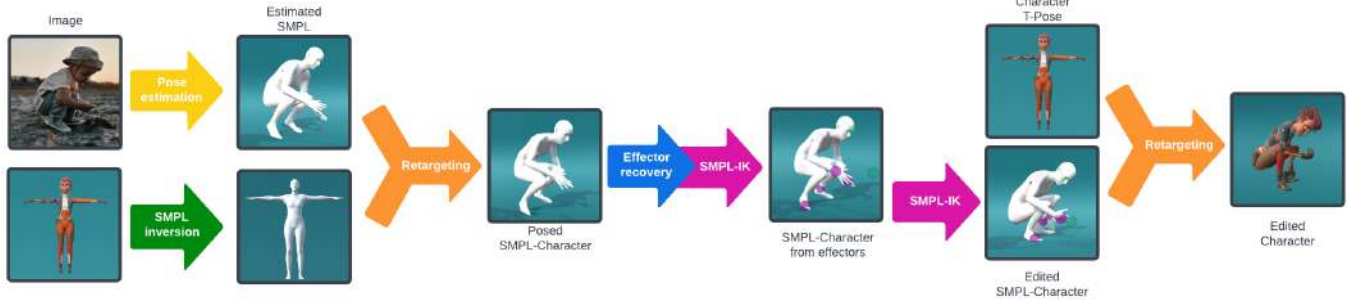
Figure 2: Pipeline for pose estimation and editing from a 2D image on a custom humanoid character.

training IK in SMPL space unlocks a seamless interface with off-the-shelf AI algorithms operating in a standardized SMPL space, such as computer vision pose estimation backbones.

4 SMPL-SI

SMPL-SI maps arbitrary humanoid skeletons onto their SMPL approximations by learning a mapping from skeleton features to the corresponding SMPL β-parameters (solving the inverse shape problem). Therefore, it can be used to map arbitrary user supplied skeletons in the SMPL representation and hence integrate SMPL-IK described above with custom user skeletons. Recall that the SMPL model implements the following forward equation:

$$\mathbf{p} = \text{SMPL}(\beta, \theta), \tag{1}$$

mapping shape parameters $\beta \in \mathbb{R}^{10}$ and pose angles $\theta \in \mathbb{R}^{22 \times 3}$ into positions $\mathbf{p} \in \mathbb{R}^{24 \times 3}$ of SMPL joints. Datasets such as H36M contain multiple tuples $(\mathbf{p}_i, \beta_i, \theta_i)$. In principle, the pairs of H36M's skeleton features $\mathbf{f}_i$ extracted from $(\mathbf{p}_i, \theta_i)$ and corresponding labels β_i, could be used for training a shape inversion model:

$$\hat{\beta} = \text{SMPL-SI}(\mathbf{f}). \tag{2}$$

However, the H36M training set contains only 6 subjects, meaning that the entire dataset contains only 6 distinct vectors β_i, which is unlikely to be sufficient for learning any meaningful SMPL-SI.

Taking this into account, we propose to train the SMPL-SI model as follows. We randomly sample 20k tuples $(\mathbf{p}_i, \beta_i)$ with $\widetilde{\beta}_i = [\epsilon_i, s_i]$, where $\epsilon_i \in \mathbb{R}^{10}$ is a sample from uniform distribution $\mathcal{U}(-5, 5)$ and $s_i \in \mathbb{R}$ is the scale sampled from $\mathcal{U}(0.2, 2)$. Scale s_i accounts for the fact that the actual user supplied characters may have much smaller or much larger overall scale than the standard SMPL model. Furthermore, we use the SMPL forward equation (1) to compute joint positions $\widetilde{\mathbf{p}}_i$ corresponding to $\widetilde{\beta}_i$ and θ_i set to the T-pose. Finally, we compute skeleton features $\widetilde{\mathbf{f}}_i$ for each $\widetilde{\mathbf{p}}_i$ as distances between the following pairs of joints: (right hip, right knee), (right knee, right ankle), (head, right ankle), (head, right wrist), (right shoulder, right elbow), (right elbow, right wrist). Given the 20k samples from the SMPL model and the features of the user skeleton, $\mathbf{f}$, we implement the kernel density estimator for the shape parameters of the SMPL model approximating the user supplied skeleton:

$$\hat{\beta} = \sum_i \frac{\widetilde{\beta}_i w_i}{\sum_j w_j}, \quad w_i = \kappa((\mathbf{f} - \widetilde{\mathbf{f}}_i)/h). \tag{3}$$

Here κ is the Gaussian kernel of width $h = 0.02$. The theory behind this implementation is that in general, for each skeleton, characterized *e.g.* by its bone lengths, there exist multiple equally plausible β's (not surprisingly SMPL-SI is an ill-defined problem, like many other inverse problems). Therefore, a point solution of the inverse problem is likely to be degenerate. To resolve this, we formulate the general solution in probabilistic Bayesian terms, based on $p(\widetilde{\beta}, \mathbf{f})$, the joint generative distribution of skeleton shape and features. The corresponding posterior distribution of β parameters given features gives rise to the following Bayesian β-estimator:

$$\hat{\beta} = \int \widetilde{\beta} p(\widetilde{\beta}|\mathbf{f}) d\widetilde{\beta}, \tag{4}$$

Note that $\hat{\beta}$ mixes a few likely values of $\widetilde{\beta}$ corresponding to posterior distribution modes. Decomposing $p(\widetilde{\beta}, \mathbf{f}) = p(\mathbf{f}|\widetilde{\beta})p(\widetilde{\beta})$ we get:

$$\hat{\beta} = \int \frac{\widetilde{\beta} p(\mathbf{f}|\widetilde{\beta}) p(\widetilde{\beta}) d\widetilde{\beta}}{\int p(\mathbf{f}|\widetilde{\beta}) p(\widetilde{\beta}) d\widetilde{\beta}}. \tag{5}$$

Since the joint distribution $p(\widetilde{\beta}, \mathbf{f})$ is unknown, we approximate it using a combination of kernel density estimation and Monte-Carlo sampling. Assuming conservative uniform prior for $p(\widetilde{\beta})$, we sample β as described above and we use a kernel density estimator $p(\mathbf{f}|\widetilde{\beta}) \approx \frac{1}{hN} \sum_i \kappa(\frac{\mathbf{f} - \widetilde{\mathbf{f}}_i}{h})$. Using this in (5) together with Monte-Carlo sampling from $p(\widetilde{\beta})$, results in (3).

5 PROPOSED AI DRIVEN WORKFLOW

Figure 1 presents a high-level summary of the proposed artistic workflow for 3D scene authoring from an image, while Fig. 2 provides the detailed overview of how it is implemented for a user-defined humanoid character. Appendices A, C and B depict simpler workflows for authoring SMPL poses, image labeling in the SMPL space and authoring poses on custom characters from scratch. These were implemented in the 3D real-time Unity engine for validation. These workflows leverage SMPL-IK and SMPL-SI building blocks as well as some others described in the rest of this section.

5.1 3D Scene From Image

We propose to process a monocular RGB image to initialize an editable 3D scene as shown in Fig. 1. A few methods exist for pose estimation from RGB inputs, most recent of which include ROMP [Sun et al. 2021] and HybrIK [Li et al. 2021]. In our approach,

we use a pre-trained ROMP model that predicts shape, 3D joint rotations and 3D root joint location for each human instance in the image. The outputs of pose estimation can be directly used to edit the estimated 3D SMPL mesh using SMPL-IK, leading to advanced 3D labelling tools that can be used to refine pose estimation and augmented reality datasets, as described in Appendix B. Alternatively, pose estimation results can be retargeted to user-supplied 3D characters. In which case, the 3D scene with retargeted characters is further edited through the combination of SMPL-IK and SMPL-SI as explained below.

5.2 Custom characters

Pose estimation algorithms output pose in the standardized SMPL space, whereas users may wish to repurpose the pose towards their own custom character. We use SMPL-SI to find the best approximation of the custom character by estimating its corresponding SMPL β parameters from the custom skeleton features (e.g. certain bone lengths). The SMPL character created using SMPL-SI provides a good approximation of the user character hence providing for its smooth integration with SMPL-IK, operating in the standard SMPL space.

5.3 Retargeting

In Figure 2, procedural retargeting first retargets the initial pose estimation result onto the SMPL approximation of the user-defined character obtained via SMPL-SI. Second, it retargets the pose edited by the animator with SMPL-IK back on the user character. On both occasions, SMPL-SI makes the job of procedural retargeting easier. First, it aligns the topology of user character with the SMPL space. Second, the SMPL character derived via SMPL-SI is a close approximation of the user character, simplifying the transfer of the pose edited with SMPL-IK back onto the user character.

5.4 Effector Recovery

Pose estimation outputs a full pose (24 3D joint angles and 3D root joint location) of each human in the scene. The pose editing process constrained by this information would be very tedious. SMPL-IK makes pose authoring efficient using very sparse constraints (e.g. using 5-6 effectors). Therefore, we propose to extract only a few effectors to create an editable initial pose. We call this *Effector Recovery*, which proceeds starting from an empty set of effectors, given the full pose provided by the computer vision backbone, in an iterative greedy fashion. Out of the remaining effectors, we add one at a time, run a new candidate effector configuration through SMPL-IK, and obtain the pose reconstructed from this configuration. We then choose a new effector configuration by retaining the candidate effector set that minimizes the L2 joint reconstruction error in the character space. We repeat this process until either the maximum number of allowed effectors is reached, or the reconstruction error falls below a fixed threshold. We find this greedy algorithm very effective in producing a minimalistic set of effectors most useful in retaining the initial pose, which is shown in supplementary video discussed in Appendix D.

Table 1: SMPL-IK baseline for the new benchmark following the randomized effector scheme [Oreshkin et al. 2021] implemented on AMASS and H36M datasets, based on MPJPE (Mean Per Joint Position Error), PA-MPJPE (Procrustes-Aligned MPJPE), and GE (Geodesic Error) metrics.

AMASS			H36M		
MPJPE	PA-MPJPE	GE	MPJPE	PA-MPJPE	GE
59.3	52.5	0.1602	65.8	57.9	0.224

5.5 Pose Editing

Pose editing relies on the Unity UX integration of SMPL-IK similar to one of ProtoRes and augmented with the SMPL shape editing controls as well as pose estimation, SMPL-SI, retargeting and effector recovery integrations. Editing happens directly in the user character space following the WYSIWYG paradigm. A full pipeline demo is presented in Appendix E.

6 EMPIRICAL RESULTS

In Table 1, we report pose reconstruction errors of our SMPL-IK approach for two datasets : AMASS [Mahmood et al. 2019] and Human3.6M [Ionescu et al. 2014] (see Appendix F.2 for more details).

7 LIMITATIONS

SMPL-IK and SMPL-SI are most effective when dealing with realistic human shapes and poses, because they are trained on the SMPL model and realistic 3D pose data from the AMASS dataset [Mahmood et al. 2019]. Obviously, they perform worse when dealing with unrealistic and disproportionate human body types, such as those of certain cartoon characters. SMPL-SI relies on a set of joints to compute user character features. These joints are present in most characters, but without them its operation is not viable.

REFERENCES

Unity 3D. 2022. Retargeting of Humanoid animations. https://docs.unity3d.com/Manual/Retargeting.html. Accessed: 2022-08-01.

Andreas Aristidou, Joan Lasenby, Yiorgos Chrysanthou, and Ariel Shamir. 2018. Inverse kinematics techniques in computer graphics: A survey. In *Computer graphics forum*, Vol. 37. Wiley Online Library, 35–58.

A. O. Bebko, A. Thaler, and N. F. Troje. 2021. bmlSUP - A SMPL Unity Player. In *Proc. IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*.

Raphael Bensadoun, Shir Gur, Nitsan Blau, Tom Shenkar, and Liorn Wolf. 2022. Neural Inverse Kinematics. In *Proc. ICML*.

F. Bocquelet, B. Oreshkin, F. Harvey, L-S Ménard, D. Laflamme, B. Raitt, and J. Cowles. 2022. AI and Physics Assisted Character Pose Authoring. In *Proc. SIGGRAPH'22 Real-Time Live!* Association for Computing Machinery, Article 3, 2 pages.

Botond Bócsi, Duy Nguyen-Tuong, Lehel Csató, Bernhard Schoelkopf, and Jan Peters. 2011. Learning inverse kinematics with structured prediction. In *Proc. International Conference on Intelligent Robots and Systems*. IEEE, 698–703.

Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. 2016. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *Proc. ECCV*. Springer, 561–578.

Vicente Ruiz De Angulo and Carme Torras. 2008. Learning inverse kinematics: Reduced sampling through decomposition into virtual robots. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 38, 6 (2008), 1571–1577.

Aaron D'Souza, Sethu Vijayakumar, and Stefan Schaal. 2001. Learning inverse kinematics. In *Proc. International Conference on Intelligent Robots and Systems.*, Vol. 1. IEEE, 298–303.

Ahmed El-Sherbiny, Mostafa A Elhosseini, and Amira Y Haikal. 2018. A comparative study of soft computing methods to solve inverse kinematics problem. *Ain Shams Engineering Journal* 9, 4 (2018), 2535–2548.

Michael Gleicher. 1998. Retargetting motion to new characters. In *Proc. Annual conference on nomputer graphics and interactive techniques*. 33–42.

Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. 2014. Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments. *IEEE Trans. PAMI* 36, 7 (jul 2014), 1325–1339.

Mitsuo Kawato, Hiroaki Gomi, Masazumi Katayamat, and Yasuharu Koike. 1993. Supervised learning for coordinative motor control. *Computational learning & cognition* (1993), 126–161.

Jiefeng Li, Chao Xu, Zhicun Chen, Siyuan Bian, Lixin Yang, and Cewu Lu. 2021. Hybrik: A hybrid analytical-neural inverse kinematics solution for 3d human pose and shape estimation. In *Proc. CVPR*. 3383–3393.

M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M.J. Black. 2015. SMPL: A skinned multi-person linear model. *ACM TOG* 34, 6 (2015), 1–16.

Zhengyi Luo, S. Alireza Golestaneh, and Kris M. Kitani. 2020. 3D Human Motion Estimation via Motion Compression and Refinement. In *Proc. the Asian Conference on Computer Vision (ACCV)*.

N. Mahmood, N. Ghorbani, N.F. Troje, G. Pons-Moll, and M.J. Black. 2019. AMASS: Archive of motion capture as surface shapes. In *Proc. ICCV*. 5442–5451.

Lucas Mourot, Ludovic Hoyet, François Le Clerc, François Schnitzler, and Pierre Hellier. 2022. A Survey on Deep Learning for Skeleton-Based Human Animation. In *Computer Graphics Forum*, Vol. 41. Wiley Online Library, 122–157.

Boris N. Oreshkin, Florent Bocquelet, Felix G. Harvey, Bay Raitt, and Dominic Laflamme. 2021. ProtoRes: Proto-Residual Network for Pose Authoring via Learned Inverse Kinematics. In *Proc. ICLR*.

Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. 2019. Expressive body capture: 3d hands, face, and body from a single image. In *Proc. CVPR*. 10975–10985.

Jathushan Rajasegaran, Georgios Pavlakos, Angjoo Kanazawa, and Jitendra Malik. 2021. Tracking People with 3D Representations. *Proc. NIPS* 34 (2021), 23703–23713.

Seyed Sadegh Mohseni Salehi, Shadab Khan, Deniz Erdogmus, and Ali Gholipour. 2018. Real-time deep pose estimation with geodesic loss for image-to-template rigid registration. *IEEE Trans. Medical Imaging* 38, 2 (2018), 470–481.

Yu Sun, Qian Bao, Wu Liu, Yili Fu, Michael J Black, and Tao Mei. 2021. Monocular, one-stage, regression of multiple 3D people. In *Proc. ICCV*. 11179–11188.

Yuxiao Zhou. 2020. Minimal-IK : https://github.com/CalciferZh/Minimal-IK.

All demo videos are available here: https://drive.google.com/drive/u/1/folders/1bHwoZjAX9njFCGszzLpUtOFGXxs0sWKW.

A   EDITING BOTH SHAPE AND POSE DEMO

Figure 3: Morphology-aware learned IK. Left: posing the average male SMPL character. Center: result of modifying only the SMPL gender parameter. Right: result of additionally modifying the SMPL β shape parameters.

This is shown in the supplementary video Demo_Pose_and_Shape_Editing.mp4. Compared to ProtoRes, SMPL-IK adds the additional flexibility of editing body shape together with pose. Figure 3 demonstrates the user interface of shape editing, including the gender setting and the controls for the 10 SMPL β parameters. In addition, the demo video shows how pose and shape of the SMPL character can be edited simultaneously. In this video, we demonstrate the benefit of SMPL-IK in pose authoring. First, we show how different effectors can be successfully manipulated using SMPL-IK leading to different realistic poses of the same body. Then, we show how changing body type, described by gender and scale, leads to different realistic versions of the same pose for different bodies. Finally, we show fine-grained modification of the body type by manipulating the SMPL shape parameters of the body. At every step, the corresponding pose is estimated using our SMPL-IK approach.
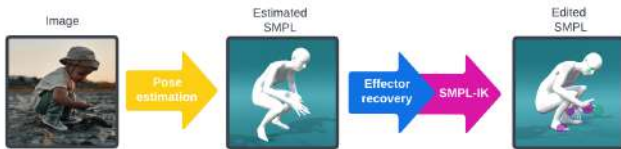
B   LABELING TOOL DEMO

Figure 4: Pipeline for 2D image labeling with accurate 3D pose.

This is shown in the supplementary video Demo_Labeling_Tool.mp4. One of the drawbacks of the current pose estimation datasets based on real data is that only 3D or 2D positions of joints are actually labeled. However, it was shown that rotations are very

important for representing a believable naturally looking pose [Oreshkin et al. 2021]. SMPL-IK can be used as a labeling tool to add the missing 3D-rotation information to existing datasets, elevating them to the next level with minimal human effort. Given an image of a human, our SMPL-IK approach (combined with an off-the-shelf image-to-pose estimator) provides an editable 3D SMPL model in a pose close to the one in the image (see Fig. 4). The labeling tool based on SMPL-IK and its integration with Unity can be used to correct the joint rotations and specify the correct lookat (head/eyes direction) that is most often estimated incorrectly by the current SOTA pose estimation algorithms due to the absence of this information in the current pose estimation datasets.

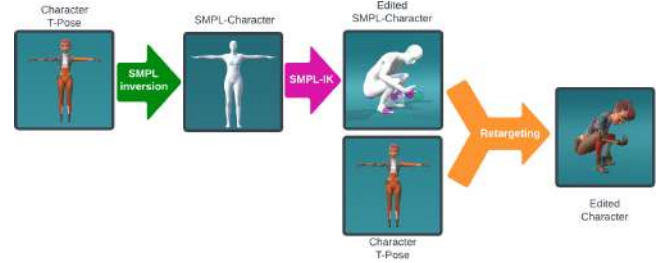## C   POSE AUTHORING ON A CUSTOM CHARACTER

Figure 5: Pipeline for pose authoring with a custom humanoid character via SMPL-IK and SMPL-SI.

Figure 5 depicts the simplified workflow that is used for authoring a pose for the custom user defined character using a combination of SMPL-IK and SMPL-SI. Supplementary videos Demo_Authoring_Pose_Child.mp4, Demo_Authoring_Pose_Child.mp4, Demo_Authoring_Pose_Female.mp4, Demo_Authoring_Pose_Male.mp4, Demo_Authoring_Pose_Strong.mp4 show how SMPL-SI can be used to manipulate 4 custom characters (child, female, male and strong male) with different proportions and morphologies.

D   EFFECTOR RECOVERY

Supplementary video Demo_Effector_Recovery.mp4 demonstrates the effector recovery mechanism in action. It shows the effect of changing the maximum number of effectors hyperparameter as well as the effect of changing number of recovered effectors on the initial pose extracted from image. It is clear that a relatively small number of effectors are sufficient to recover a good initialization for the editable pose.

E   FULL PIPELINE DEMO

E.1   Demo_Crouch_FineTuning.mp4

In this video, we show how to edit a pose in Unity using our approach. First, given a user-provided 3D character, SMPL-SI is used to estimate the SMPL body shape parameters that best fit the character. This SMPL-Character is shown in the video transparently along with the 3D character, and is also shown in the second image in Appendix C. Then, given an image of a human in a pose, such as

the crouched baby in Appendix B, an off-the-shelf image-to-pose estimator is used to obtain its SMPL pose parameters. Then, the SMPL-Character is retargeted onto the estimated pose. Next, for further editing of the character from the new pose, Effector Recovery is performed to recover the best effectors that describe that pose for that character. The effectors are shown in purple. These effectors can now be used to edit the pose as the user wishes. Optionally, more effectors could be activated for further fine-tuned editing, including both positional and rotational effectors.

E.2 Demo_Sitting_Editing.mp4

In this video, we demonstrate the case shown in the teaser image, with two humanoid 3D characters. As image of two people sitting is loaded, the poses of the two people are estimated using an off-the-shelf image-to-pose model, and the two 3D characters are retargeted to these estimated poses. Further, the pose of the 3D characters are then edited by manipulating the effectors. The video shows the various effectors and the effect of manipulating them. Every manipulation uses our SMPL-IK approach to estimate the realistic pose of that character.

F SMPL-IK DETAILS
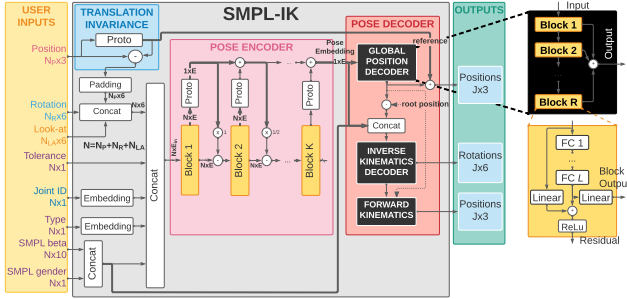
F.1 SMPL-IK Neural Network Diagram

Figure 6: SMPL-IK neural network diagram. Note input conditioning on β and gender inputs.

F.2 SMPL-IK Training and Evaluation Details

The training process overall follows that of ProtoRes [Oreshkin et al. 2021]. We found that naively training on AMASS works well, while for H36M, simply training on its 6 subjects results in the lack of generalization in the β subspace of inputs. To overcome this, we used the following β augmentation strategy. For each sample drawn from the H36M dataset we added white Gaussian noise with unit variance and recalculated joint positions based on the augmented value of β and the pose θ from the dataset. The model was trained on the augmented H36M dataset. We found that overall, the qualitative model quality was better when it was trained on the AMASS dataset, although the quality of the H36M model was also acceptable.

To measure quantitative generalization results, we used H36M train and test splits derived in ROMP [Sun et al. 2021], which in

tern follow H36M Protocol 2 (subjects S1, S5, S6, S7, S8 for training and S9, S11 for test, plus 1:10 subsampling of the training set) and AMASS train/validation/test splits from [Mahmood et al. 2019] (valdation datasets: HumanEva, MPI_HDM05, SFU, MPI_mosh; test datasets: Transitions_mocap, SSM_synced; training datasets: everything else).

To quantify SMPL-IK, we choose to use evaluation metrics commonly used in the context of H36M and AMASS datasets, MPJPE and PA-MPJPE plus the geodesic rotation error, which was shown to be important in quantifying the quality of realistic poses in [Oreshkin et al. 2021]. The metrics are defined as follows.

GE, geodesic error, between a rotation matrix $\mathbf{R}$ and its prediction $\widehat{\mathbf{R}}$, Salehi et al. [2018]:

$$\mathrm{GE}(\mathbf{R}, \widehat{\mathbf{R}}) = \arccos\left[(\mathrm{tr}(\widehat{\mathbf{R}}^T\mathbf{R}) - 1)/2\right]. \tag{6}$$

MPJPE, mean per joint position error, is computed by flattening all poses and joints in the batch into the leading dimension resulting in the ground truth tensor $\mathbf{p} \in \mathbb{R}^{N \times 3}$ and its prediction $\widehat{\mathbf{p}}$:

$$\mathrm{MPJPE}(\mathbf{p}, \widehat{\mathbf{p}}) = \frac{1}{N}\sum_{i=1}^{N}\|\mathbf{p}_i - \widehat{\mathbf{p}}_i\|_2. \tag{7}$$

PA-MPJPE, Procrustes aligned MPJPE, is MPJPE calculated after each estimated 3D pose in the batch is aligned to its respective ground truth by the Procrustes method, which is simply a similarity transformation.

All metrics in Table 1 are computed on test sets of AMASS and H36M using models trained on respective training sets using the randomized effector benchmark framework described in detail in [Oreshkin et al. 2021].