Reducing Footskate in Human Motion Reconstruction with Ground Contact Constraints

Yuliang Zou[1] Jimei Yang[2] Duygu Ceylan[2] Jianming Zhang[2] Federico Perazzi[2]
Jia-Bin Huang[1]
[1]Virginia Tech [2]Adobe Research

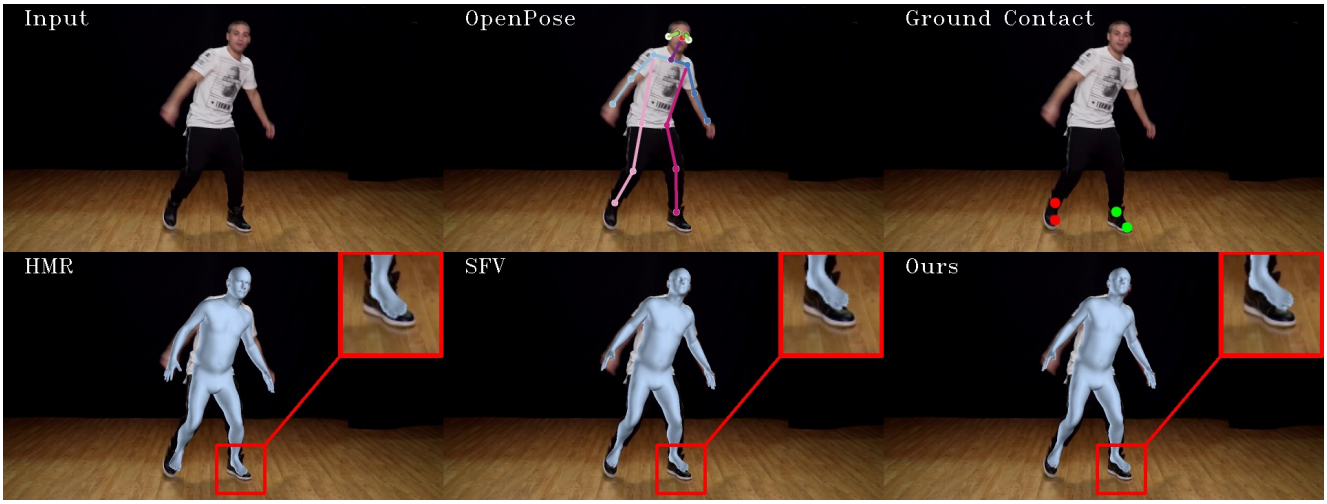

Figure 1. **Motion reconstruction.** Applying single-frame human shape and pose estimation method (HMR [16]) to a video frame-by-frame often results in unwanted flickering artifacts. Temporal smoothing algorithm (SFV [34]) reduces the jitters but still suffers from foot slippage artifact. With explicitly predicted ground contact constraint (marked in green), our method produces a more physically realistic motion trajectory with significantly less footskate artifacts . Animations can be viewed at https://yuliang.vision/.

Abstract

In this paper, we aim to reduce the footskate artifacts when reconstructing human dynamics from monocular RGB videos. Recent work has made substantial progress in improving the temporal smoothness of the reconstructed motion trajectories. Their results, however, still suffer from severe foot skating and slippage artifacts. To tackle this issue, we present a neural network based detector for localizing ground contact events of human feet and use it to impose a physical constraint for optimization of the whole human dynamics in a video. We present a detailed study on the proposed ground contact detector and demonstrate high-quality human motion reconstruction results in various videos.

1. Introduction

Human sensing from visual data has been an active research area in computer vision with many applications in augmented and mixed reality, animation, and re-enactment. Thanks to the success of deep learning methods and the availability of large-scale datasets, single-image-based 3D human pose and shape estimation has made significant progress in recent years [3, 16, 21, 25, 26, 29, 32, 40, 42, 45]. However, many applications in animation, motion re-targeting, and imitation learning pose additional requirements for a visual sensing system. First, in addition to recovering the pose in a human-centric coordinate system (which is the common practice for single-view reconstruction methods), it is desired to recover the motion trajectory of the person as well. Second, reconstructed motion sequences not only need to be temporally smooth and coherent but also need to be visually plausible. Recent works [17, 34, 36] take temporal context into account to generate more smooth and accurate

motion trajectories. However, without explicit constraints on the physical plausibility, these methods often suffer from common artifacts such as foot skating.

In this paper, we propose a method to reduce footskate artifacts in human motion reconstruction, which can be readily used for animation and retargeting tasks. Since foot skating is crucial for the realism of a motion [18], we focus on explicitly modeling the physical constraints between human feet and the ground plane. Our core idea is that when feet are in contact with the ground, zero velocity constraints can be exerted to corresponding joints (toes or heels). Optimizing for motion trajectories with this additional constraint results in visually satisfying motion, as illustrated in Figure 1, Figure 7, and the accompanying supplementary video.

Our approach consists of three main steps. First, given the input video, we use a state-of-the-art single-view human pose and shape estimation method [16] to obtain local estimates. We then utilize the 2D pose information in a novel method to predict ground contact events in the input video. Finally, we present a novel optimization strategy to recover a motion sequence that explores both temporal and ground contact constraints. This results in a smooth and visually satisfying motion along with the motion trajectory.

Our contributions are summarized below.

- We propose a novel network architecture to detect ground contact events from 2D keypoint estimations. Using such intermediate representations instead of raw pixels as input simplifies the annotation of ground truth contact events. Thus, we present a semi-automatic algorithm to collect a dataset using motion capture data.
- We model ground contact events with a simple zero-velocity constraint in our motion reconstruction optimization. Together with the smoothness objective, this significantly reduces the footskate artifacts.
- In addition to quantitative evaluations, we also verify the effectiveness of our method in various real-world videos. All the code and data will be made publicly available to facilitate future research.

2. Related Work

3D pose and shape estimation from a single image. Single-image 3D human pose estimation, often formulated as locating major 3D joints of the human body in a human-centric coordinate space, is a fundamental problem in computer vision. Mainstream methods predict 3D pose either from a single RGB image [21, 31, 40, 45] or from intermediate representations such as 2D joint detection [7, 25, 29]. There are also recent methods [26, 27, 37, 39] that integrate 2D pose estimation to the 3D pose prediction pipeline. Note that most of the methods above only estimate the joint locations, which is not sufficient for many applications in AR and VR that require information about the body as well. Methods combining UV maps [2] or parametric body models [3, 16, 32, 42] are thus proposed to provide more fine-grained information. Built on the state-of-the-art 3D shape estimation method HMR [16], our method also utilizes a parametric body model (SMPL [24]) to produce motion reconstructions with fine-grained information. Unlike most of the above work, however, we also focus on recovering a physically plausible motion trajectory of the person.

3D pose and shape estimation from monocular videos. Due to the lack of temporal cues, directly applying single-image 3D reconstruction models on videos frame-by-frame results in non-smooth trajectories and suffer from artifacts such as jitter. Prior works [28, 34, 36, 46, 48] formulate a constrained optimization problem to obtain smooth motion trajectories by taking advantage of 2D cues. Rhodin et al. [36] use human silhouettes to optimize 3D poses and shapes in videos. Mehta et al. [28] propose an online skeleton tracking algorithm to obtain smooth pose sequences in real-time. Zanfir et al. [46] integrate scene constraints to the optimization problem for multi-person settings. Recent methods also learn to predict smooth 3D pose sequences directly from video input, either by LSTMs [20, 35] or temporal CNNs [8, 33], which can be used to simplify the optimization. Our method is built upon the 2D-3D optimization proposed by Peng et al. [34] as it is flexible to integrate additional nonlinear terms that are important for reducing footskate artifacts .

Modeling physical contact. Physical contact plays an important role in modeling both human and humanoid robot motions, such as walking [13]. Generating artificial motion is usually formulated as an optimal control problem under contact and other physical constraints [9, 38], being widely studied in the robotics community. Due to the non-smoothness introduced by the active/inactive status of contact points, jointly optimizing for contact states and the motion trajectory is challenging [44]. A common solution is to estimate the sequence of contact states first and use it in a subsequent optimization as a fixed variable [10, 19, 41]. The pioneering work by Brubaker *et al*. [4] tackles the 3D pose tracking and contact dynamics estimation simultaneously using a continuous contact model. This method can generate a physically plausible motion trajectory, but it requires MoCap data or multi-view videos as input, while our method only takes monocular RGB videos as input. Livne *et al*. [23] also tackles a similar problem but takes 3D point cloud as input instead. Wei *et al*. [43] aims to recover physically realistic human motions from monocular video sequences. However, they require manual annotations of contact events, while our system automatically detects those ground contact events. In a recent work, Zanfir *et al*. [46] jointly optimize for both human body shape and a ground plane where the humans are

assumed to stand on. However, they estimate foot contacts based on simple thresholding of the distance between the ankle joints and the ground plane. This simple strategy is not sufficient to avoid the foot slipping artifact that is commonly observed in motion sequences recovered by 3D pose and shape estimation methods. Funk *et al*. [11] focuses on estimating foot pressure from 2D human pose extracted in an input video. We also utilize the 2D keypoints but instead predict the binary state of foot contacts and further utilize this information for optimizing the motion trajectory. The most relevant work to ours is the concurrent work of Li *et al*. [22], where they exploit the contact constraints to estimate human pose and contact forces. While both Li *et al*. [22] and our method adopt the two-step approach of first identifying contact points, we have a few distinctions. First, they utilize RGB image patches to train a separate model for predicting the contact states of each keypoint, while we use a single network taking as input the 2D keypoints, to estimate the ground contact status. Second, building on top of the work of Peng*et al*. [34], we perform the motion trajectory optimization in the latent space while Li *et al*. [22] optimize for the motion variables explicitly. Our approach ensures to stay in the learned motion manifold while ensuring physical plausibility. Finally, we extensively evaluate both the robustness of our contact detection module as well as the quality of the reconstructed motion trajectories.

3. Method

Our goal is to reduce footskate artifacts in human motion reconstruction, which takes as input monocular RGB videos. Our system can handle both static and moving cameras. For moving cameras, structure-from-motion (SfM) is first applied to estimate the camera motion. For simplicity, we assume the camera is fixed over the whole video sequence in the following discussion. Our system consists of three stages: 1) pose estimation, 2) ground contact prediction, and 3) motion reconstruction. Figure 2 shows a high-level sketch of our system. Given a video clip, we first run OpenPose [5] and HMR [16] to estimate 2D and 3D human pose for each frame. We then feed the 2D pose data into a pre-trained ground contact prediction model to estimate the ground contact state of each foot joint for each frame. With the estimated ground contact states, we solve an optimization problem with explicit ground contact constraints to generate a motion trajectory with less footskate artifacts . In this section, we first describe how we model the ground contact events and the network design. We then provide background for human pose and shape estimation using HMR [16] and how temporal smoothing works. Lastly, we introduce the proposed zero velocity constraint based on ground contact events.

3.1. Ground contact detection

We aim to recognize the occurrence of the physical event of human feet contacting the ground. We model each foot with a rigid skeleton of three connected joints: ankle, toe, and heel, as illustrated in Figure 2. We assume only toe and heel can contact the ground and use a binary label to indicate whether each one of the four joints (left toe, left heel, right toe, and right heel) contacts the ground. Thus, a ground contact event at a particular frame can be represented by a four-bit vector. We assume general ground surfaces are not slippery, so zero velocity should be observed in the video when the contact happens for a particular foot joint.

Dataset construction. There are no existing datasets with ground-contact annotations available, and thus we choose to construct a new dataset for this purpose. Manually annotating contact status for videos can be tedious, so we develop a semi-automatic method to obtain ground truth contact labels. We first collect multi-view videos from the walking motion category in the Human3.6M dataset [15] as well as dancing and sports sequences in the MADS dataset [47]. We run OpenPose [5] on all these videos to extract 2D keypoints on feet (left big toe, left heel, right big toe, and right heel). Based on the zero velocity assumption, we decide the ground contact status of each keypoint by measuring the distance between two consecutive frames. If it is smaller than a threshold, we consider it is contacting the ground. We manually annotate a small portion of frames to tune the threshold and use it for the whole dataset. The human action sequences are captured by multiple RGB cameras, so we accept the label only if the status predictions from four views are consistent. Now that we have a large training set with noisy labels, we screen all the frames to examine their annotations and filter out the bad ones to correct manually. In total, we have around 60,000 frames for the training set and around 15,000 frames for the validation set.

Learning. Since our training data comes from MoCap videos in lab environments, training our model using raw pixels would not have a good generalization ability to real-world indoor/outdoor scenes. We thus choose to use other features as our input. Intuitively, 2D keypoint locations within a local temporal window contain the motion cues. However, raw 2D keypoint locations from OpenPose [5] has two issues. First, OpenPose [5] produces missed keypoints over the frames due to detection failures or occlusions. Second, OpenPose [5] estimates joint positions frame by frame, where temporal inconsistency could cause high-frequency jitters. To address these issues, we first perform linear interpolation, which also provides estimates of the locations of those missed keypoints. We then apply the OneEuro Filter [6] to reduce high-frequency noise, suggested by Mehta *et al*. [28].

To detect the ground contact events for each foot keypoint,

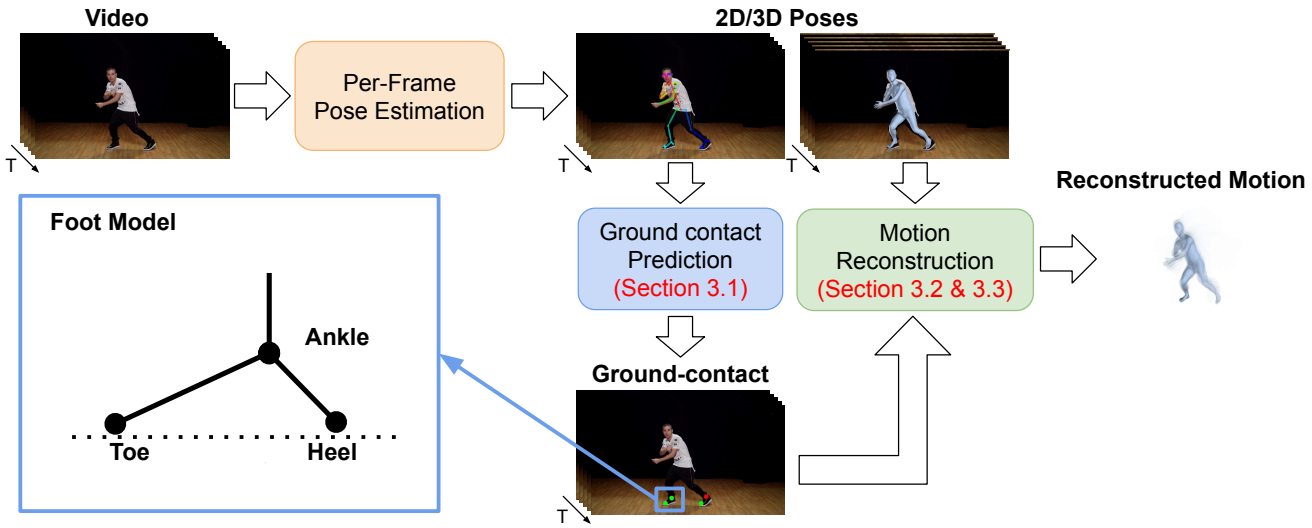

Figure 2. **Overview.** Our system consists of three stages: pose estimation, ground-contact prediction, and motion reconstruction. The foot skeleton model is illustrated in bottom left.

we utilize a temporal convolution network with residual connections, inspired by Pavllo *et al*. [33]. Figure 3 shows the network architecture. The input is a $B \times T \times K \times C$ tensor, where B is the batch size, T is the temporal window size, K is the number of keypoints, and C is the number of features. The kernel size of all the temporal convolutional layers is 3, except for the last one, which we set it to the size of the temporal window to reduce the temporal size to one. Before all the temporal convolutional layers except for the last one, we perform replicate padding to deal with the boundaries. The last layer consists of four sigmoid units for generating contact labels.

3.2. Single-frame estimation and temporal smoothing

Notations. We adopt the recent state-of-the-art method HMR [16] to initialize the pose and shape parameters, which uses the SMPL body model [24] as the representation. SMPL is a generative model factoring out human 3D mesh into shape parameters, $\beta \in \mathbf{R}^{10}$, and pose parameters, $\theta \in \mathbf{R}^{69}$. Shape parameters control the body proportion, height, and weight, while pose parameters are bone rotation angles, controlling the body deformation. Given an image as input, HMR first encodes it into an embedding space $z \in \mathbf{R}^{2048}$, then decodes the embedding to predict shape parameters β, pose parameters θ, and camera parameters Π. The local 3D keypoint positions can then be represented as $x_{3D} = \text{SMPL}(\beta, \theta)$, whereas the local 2D keypoint positions are denoted as $x_{2D} = \Pi(x_{3D})$.

Optimization objective. When running on an input video, HMR predicts shape and pose parameters on each frame independently, without considering any temporal coherency. Building on top of HMR, SFV [34] solves the following optimization problem to ensure temporal smoothness:

$$L_{\text{SFV}} = w_{\text{2D}}L_{\text{2D}} + w_{\text{3D}}L_{\text{3D}} + w_{\text{shape}}L_{\text{shape}} + w_{\text{smooth}}L_{\text{smooth}} + w_{\text{cam}}L_{\text{cam}} \quad (1)$$

The 2D consistency loss L_{2D} minimizes the reprojection error between the projection of predicted 3D joint locations and the OpenPose [5] output. The error for each joint is weighted by the detection confidence from OpenPose [5]. Note that HMR predicts 3D shape and local pose parameters, which are used to obtain local 3D keypoint locations relative to the root, i.e., hips. However, HMR works on fixed-size input images where the images have been scaled such that the height of the person is roughly half the height of the image. In order to ensure the OpenPose predictions and the projection of predicted 3D keypoints are at the same coordinate system, we convert the global 2D keypoint positions from OpenPose $x_{\text{OP, global}}$ to local coordinates $x_{\text{OP}} \in [-1, 1]$:

$$x_{\text{OP}} = 2\frac{S(x_{\text{OP, global}} - P)}{H} - 1 \quad (2)$$

where S is a scale factor to normalize the person height, P is the starting point from where we crop out the patch, H is a pre-defined size of the cropped patch. Thus, the 2D consistency loss is $L_{2D} = \sum_t \|x_{2D}^{(t)} - x_{\text{OP}}^{(t)}\|$. The 3D consistency loss L_{3D} enforces the optimized 3D pose parameters to stay close to the initial 3D pose parameters predicted by HMR. The shape consistency loss L_{shape} minimizes the variance of the predicted shape parameters, enforcing the shape of the person does not change rapidly over time. The smoothness loss L_{smooth} minimizes the difference of 3D joint locations

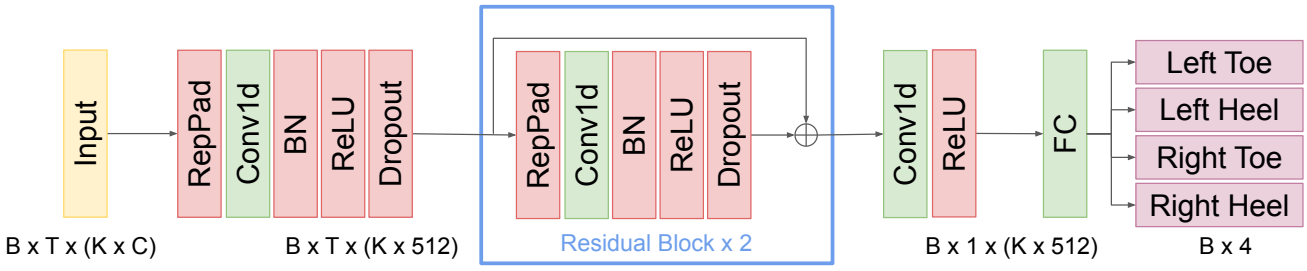

Figure 3. **Network architecture of the ground contact detection model.**

between adjacent frames enforcing temporal smoothness. Lastly, a camera consistency loss is added to minimize the camera parameter difference between adjacent frames. Note that SFV [34] does not optimize for pose, shape, and camera parameters directly, instead it optimizes for the latent code z in the embedding space. For more details, we refer the reader to the original SFV paper [34].

3.3. Motion reconstruction with ground contact constraint

As shown in Figure 1, compared to the single frame baseline (HMR), temporal smoothing (SFV) reduces the jittering and generates a more temporal coherent motion trajectory. However, if we have a closer look at the results, we observe a severe foot sliding phenomenon when the feet are supposed to be planted on the ground. With the estimated ground contact states, a natural way is to set the velocity of those keypoints that contact the ground to zero. Thus, we formulate this constraint on the position of the global 2D joint locations. We first recover the global position of the projections of the predicted 3D keypoints as

$$x_{2D,\text{global}} = \frac{H(x_{2D}+1)}{2S} + P \tag{3}$$

We assume $x^{(k,t)}_{2D,global}$ denotes the global position of the 2D projection of the keypoint k in frame t, S_{foot} denotes the set of foot keypoints which we use in our ground contact model [1], and $y^{(k,t)}$ is the binary label of ground contact status for the keypoint k at the frame t. The proposed zero velocity constraint can then be represented as

$$L_{zv} = \sum_{t=2}^{T} \sum_{k \in S_{\text{foot}}} y^{(k,t-1)} y^{(k,t)} \left|\left| x^{(k,t-1)}_{2D,global} - x^{(k,t)}_{2D,global} \right|\right|^2 \tag{4}$$

The overall objective can thus be written as

$$L_{overall} = L_{SFV} + w_{zv} L_{zv} \tag{5}$$

[1]The keypoint definition of SMPL does not contain left or right heel, we get the keypoint location from mesh representation with corresponding vertex IDs.

where w_{zv} is the trade-off weight. However, we found that optimizing (5) is very challenging as L_{zv} introduces non-smoothness at ground contacts. Therefore, we propose to first optimize L_{SFV} to its convergence and then optimize $L_{overall}$. In Sec. 4.2, we compare the loss curves of each term with and without L_{zv}.

4. Experimental Results

We evaluate our system from two perspectives, ground contact estimation performance and the quality of our reconstructed motion.

4.1. Ground contact prediction

For the test set, we select a few sequences from Human3.6M [15], MADS [47], and real-world videos used in Peng *et al*. [34]. We then carefully annotate the ground contact labels for them, resulting in around 1,500 frames. Note that they are non-overlapping with the training set. Sample videos can be found in the supplementary material.

Implementation details. We set the batch size B to 512, temporal window size T to 7. We set the number of keypoints K to 16 for the network training, by selecting the lower body joints from OpenPose output. The intermediate feature channel is 512. The initial learning rate is 1e-4, and it decreases under the exponential schedule for each epoch, with a decay rate of $\gamma = 0.99$. The max number of epochs is 80. We implement the ground contact predictor using PyTorch [30]. It takes less than two hours to finish training a single model on a GTX1080 GPU. The best network for each model discussed below is selected according to its performance on the validation set.

Comparing different input features. Since the 2D keypoint detection might be inaccurate, we also explore other features such as optical flow vectors. Instead of using a dense flow field from the whole image, we crop a 5×5 window around each keypoint and compute the median value as the flow representation of that keypoint. Empirically, we found FlowNet2 [14] generates accurate flow for our data. We also try to include the OpenPose detection score as an additional input channel.

Table 1. **Ground contact prediction results**. The best performance of each column is in **bold**.

	Left Toe	Left Heel	Right Toe	Right Heel	mean AP
Keypoint (w/o training)	0.9418	0.8314	0.9437	0.7876	0.8761
Flow (w/o training)	0.9169	0.8003	0.9426	0.7881	0.8620
Flow	0.9670	0.8559	0.9422	0.8284	0.8984
Keypoint	**0.9755**	**0.8960**	**0.9662**	**0.8789**	**0.9292**
Keypoint + Detection score	0.9686	0.8783	0.9588	0.8762	0.9205
Keypoint + Flow	0.9725	0.8846	0.9634	0.8700	0.9226

Baselines. Using the input features discussed above, we use two learning-free heuristics as our baselines: 1) When the distance between the same keypoint in two consecutive frames is less than a threshold, we label the keypoint in the first frame as contacting the ground; 2) When the flow feature of a keypoint is smaller than a threshold, we label it as contacting the ground.

Quantitative evaluation. To make comparisons among different methods, we choose to use average precision (AP) as our evaluation metric. We report the average precision for each foot keypoint, and also the mean average precision (mAP). Table 1 shows the performance of each method on our ground contact test set. The learning-based method outperforms the two simple baseline methods by a large margin, which validates the necessity of training a model for ground contact detection. And we notice that the model with keypoint as input achieves a better mAP than the model with flow as input. Also, adding the detection score from OpenPose as additional input decreases the performance. We conjecture that it increases the difficulty of network learning since the detection score does not provide direct information about ground contact state.

Design choices. To validate our design choices for the ground contact detection model, we conduct two more experiments. First, we change the size of the temporal window and evaluate how this affects the final performance. As we can see in Figure 4, the best performance is achieved with a temporal window of size $T = 7$. Second, we verify the effectiveness of the OpenPose keypoint clean up step and find it improves the mean AP from 0.91 to 0.9292 and the AP of the left heel by 3%.

Table 2. **With and without keypoint clean-up.**

	Left Toe	Left Heel	Right Toe	Right Heel	mean AP
w/o clean-up	0.9515	0.8630	0.9610	0.8645	0.9100
w/ clean-up	**0.9755**	**0.8960**	**0.9662**	**0.8789**	**0.9292**

Qualitative result. Figure 5 shows qualitative results from our trained ground contact model. It contains both indoor and outdoor scenes, and the motions range from walking to dancing and sports, showing that our model can generalize to different motion patterns not seen in the training set.

Figure 4. **Different temporal window size.**

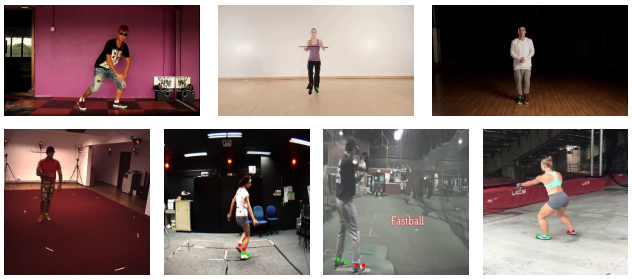

Figure 5. **Qualitative results of ground contact estimation.** When the prediction for a keypoint is in contact with the ground, we plot it in green, otherwise we use red. Our ground contact model generates accurate detections from indoor to outdoor scenes, from walking sequence to challenging dancing and sports sequences, validating the generalization ability of our model. Animations can be viewed at https://yuliang.vision/.

4.2. Motion reconstruction

Qualitative result. In Figure 7, We compare our reconstructed motions with HMR [16] and SFV [34]. More videos can be found in supplementary materials.

Implementation details. We implement the motion reconstruction optimization based on SFV [34], using TensorFlow [1] and the Adam solver. For the first stage of

optimization, we use the default setting of SFV [34]. In the second stage of optimization, we set $w_{zv} = 0.5$. The learning rate of the second stage is set to 1e-4, which is one-tenth of the initial learning rate used in the first stage. The number of optimization iterations is set to 100, since empirically we found the whole objective converges within 100 iterations.

3D pose estimation. As a byproduct of the optimization, our motion reconstruction achieves better pose estimation results. We evaluate pose estimation performance on walking sequences of the Human3.6M dataset [15]. We evaluate on the front-view camera. The performance is measured by mean per-joint position error (MPJPE) after rigid alignment [12]. We show the results in Table 3.

Table 3. **3D Pose estimation results** on Human3.6M [15]. We evaluate mean per-joint position error (MPJPE) on the front-view camera after robust alignment.

	WalkDog	Walking	WalkTogether	Average
HMR [16]	75.05	64.82	73.29	71.05
SFV [34]	72.68	66.63	72.21	70.42
Ours	**72.26**	**65.61**	**71.27**	**69.63**

Why is two-stage optimization necessary? As we mentioned previously, directly optimizing (5) is very challenging as L_{zv} introduces non-smoothness at ground contacts. To validate this claim, we apply the single-stage optimization strategy on a demo video (shown in Figure 1) with two different objectives, L_{SFV} and L_{overall}. Figure 6 shows the loss curves of each constraint. We can see that the zero velocity loss conflicts with the other constraints if we directly optimize all the loss terms simultaneously in one stage.

5. Conclusions

We have presented a method to reduce footskate artifacts in 3D human motion reconstruction by explicitly modeling the ground contact events. We proposed a zero velocity constraint at contact points to optimize the whole sequence jointly to remove foot skating. To learn when and where the ground contacts happen, we collected a dataset and developed a neural network based foot contact detector using 2D keypoints as input, which we also found more robust than non-learning heuristic-based detection methods. We presented motion reconstruction results on various lab and real-world videos and demonstrated the improvement of our method made over the state-of-the-art.

Our work handles foot contacts with the ground plane, and it will be interesting to generalize it to more contact events of other parts of the body and of other objects and people in the environment.

Acknowledgement. This work was supported in part by NSF under Grant No. 1755785 and Adobe Gift. We thank the support of NVIDIA Corporation with the GPU donation.

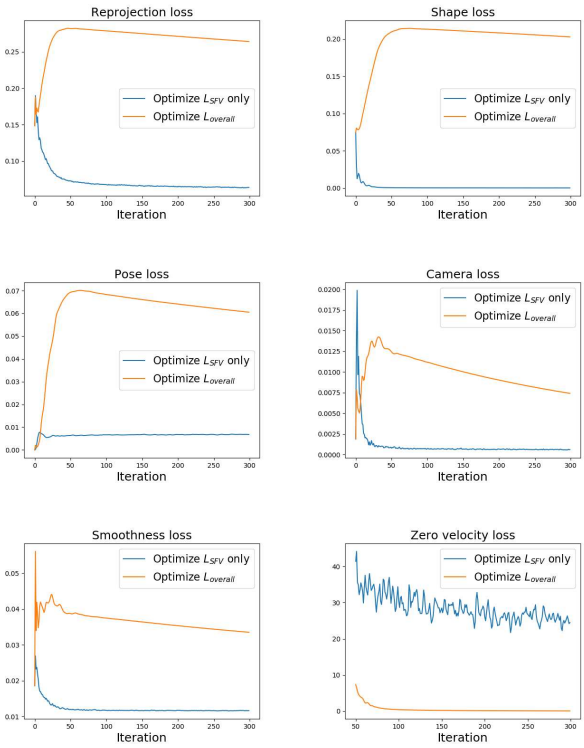

Figure 6. **Zero velocity constraint for optimization.** To validate the necessity of the two-stage optimization strategy, we conduct the single-stage optimization experiments with different objectives, L_{SFV} only and L_{overall}. The loss curves show that the zero velocity loss conflicts with other loss terms if we try to optimize them simultaneously in one single stage.

References

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org. 6

[2] Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *CVPR*, 2018. 2

[3] 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*, 2016. 1, 2

[4] Marcus A Brubaker, Leonid Sigal, and David J Fleet. Esti-

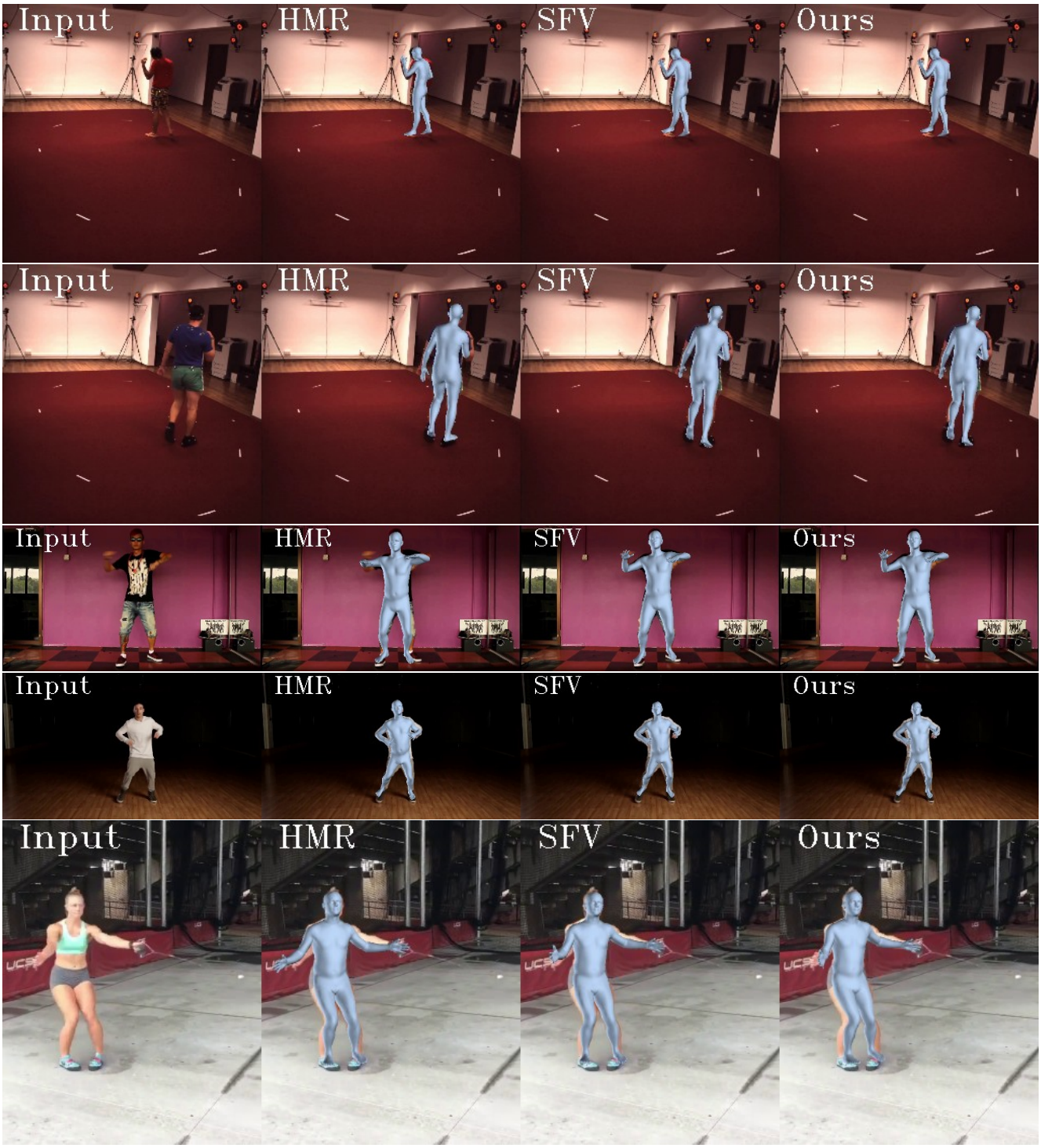

Figure 7. **Qualitative results of motion reconstruction.** With accurate ground contact detections, our method effectively alleviate the foot skating and slipping issues, reconstructing more physically plausible motion trajectories. Animations can be viewed at https://yuliang.vision/.

mating contact dynamics. In *ICCV*, 2009. 2

[5] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. In *arXiv preprint arXiv:1812.08008*, 2018. 3, 4

[6] Géry Casiez, Nicolas Roussel, and Daniel Vogel. 1 filter: a

simple speed-based low-pass filter for noisy input in interactive systems. In *SIGCHI*, 2012. 3

[7] Yu-Wei Chao, Jimei Yang, Brian Price, Scott Cohen, and Jia Deng. Forecasting human dynamics from static images. In *CVPR*, 2017. 2

[8] Rishabh Dabral, Anurag Mundhada, Uday Kusupati, Safeer Afaque, Abhishek Sharma, and Arjun Jain. Learning 3d human pose from structure and motion. In *ECCV*, 2018. 2

[9] Moritz Diehl, Hans Georg Bock, Holger Diedam, and P-B Wieber. Fast direct multiple shooting algorithms for optimal robot control. In *Fast motions in biomechanics and robotics*. 2006. 2

[10] Xiaoxiao Du, Ram Vasudevan, and Matthew Johnson-Roberson. Bio-lstm: A biomechanically inspired recurrent neural network for 3-d pedestrian pose and gait prediction. *IEEE Robotics and Automation Letters*, 4(2):1501–1508, 2019. 2

[11] Christopher Funk, Savinay Nagendra, Jesse Scott, John H Challis, Robert T Collins, and Yanxi Liu. Learning dynamics from kinematics: Estimating foot pressure from video. *arXiv preprint arXiv:1811.12607*, 2018. 3

[12] John C Gower. Generalized procrustes analysis. *Psychometrika*, 40(1):33–51, 1975. 7

[13] Andrei Herdt, Nicolas Perrin, and Pierre-Brice Wieber. Walking without thinking about it. In *IROS*, 2010. 2

[14] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *CVPR*, 2017. 5

[15] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *TPAMI*, 36(7):1325–1339, Jul. 2014. 3, 5, 7

[16] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *CVPR*, 2018. 1, 2, 3, 4, 6, 7

[17] Angjoo Kanazawa, Jason Y. Zhang, Panna Felsen, and Jitendra Malik. Learning 3d human dynamics from video. In *CVPR*, 2019. 1

[18] Lucas Kovar, John Schreiner, and Michael Gleicher. Footskate cleanup for motion capture editing. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '02, pages 97–104, New York, NY, USA, 2002. ACM. 2

[19] James Kuffner, Koichi Nishiwaki, Satoshi Kagami, Masayuki Inaba, and Hirochika Inoue. Motion planning for humanoid robots. In *Robotics Research. The Eleventh International Symposium*, pages 365–374. Springer, 2005. 2

[20] Kyoungoh Lee, Inwoong Lee, and Sanghoon Lee. Propagating lstm: 3d pose estimation based on joint interdependency. In *ECCV*, 2018. 2

[21] Sijin Li and Antoni B Chan. 3d human pose estimation from monocular images with deep convolutional neural network. In *ACCV*, 2014. 1, 2

[22] Zongmian Li, Jiri Sedlar, Justin Carpentier, Ivan Laptev, Nicolas Mansard, and Josef Sivic. Estimating 3d motion and forces of person-object interactions from monocular video. In *CVPR*, 2019. 3

[23] Micha Livne, Leonid Sigal, Marcus Brubaker, and David Fleet. Walking on thin air: Environment-free physics-based markerless motion capture. In *CRV*, 2018. 2

[24] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM TOG (Proc. SIGGRAPH)*, 34(6):248, 2015. 2, 4

[25] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3d human pose estimation. In *ICCV*, 2017. 1, 2

[26] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *3DV*, 2017. 1, 2

[27] Dushyant Mehta, Oleksandr Sotnychenko, Franziska Mueller, Weipeng Xu, Srinath Sridhar, Gerard Pons-Moll, and Christian Theobalt. Single-shot multi-person 3d pose estimation from monocular rgb. In *3DV*, 2018. 2

[28] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. Vnect: Real-time 3d human pose estimation with a single rgb camera. *ACM TOG (Proc. SIGGRAPH)*, 36(4):44, 2017. 2, 3

[29] Francesc Moreno-Noguer. 3d human pose estimation from a single image via distance matrix regression. In *CVPR*, 2017. 1, 2

[30] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch. In *NeurIPS Workshop*, 2017. 5

[31] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. Coarse-to-fine volumetric prediction for single-image 3d human pose. In *CVPR*, 2017. 2

[32] Georgios Pavlakos, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis. Learning to estimate 3d human pose and shape from a single color image. In *CVPR*, 2018. 1, 2

[33] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *CVPR*, 2019. 2, 4

[34] Xue Bin Peng, Angjoo Kanazawa, Jitendra Malik, Pieter Abbeel, and Sergey Levine. Sfv: Reinforcement learning of physical skills from videos. *ACM TOG (Proc. SIGGRAPH Asia)*, 37(6), Nov. 2018. 1, 2, 3, 4, 5, 6, 7

[35] Mir Rayat Imtiaz Hossain and James J Little. Exploiting temporal information for 3d human pose estimation. In *ECCV*, 2018. 2

[36] Helge Rhodin, Nadia Robertini, Dan Casas, Christian Richardt, Hans-Peter Seidel, and Christian Theobalt. General automatic human shape and motion capture using volumetric contour cues. In *ECCV*, 2016. 1, 2

[37] Gregory Rogez, Philippe Weinzaepfel, and Cordelia Schmid. Lcr-net++: Multi-person 2d and 3d pose detection in natural images. *TPAMI*, 2019. 2

[38] Gerrit Schultz and Katja Mombaur. Modeling and optimal control of human-like running. *IEEE/ASME Transactions on mechatronics*, 15(5):783–792, 2010. 2

[39] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *ECCV*, 2018. 2

[40] Bugra Tekin, Isinsu Katircioglu, Mathieu Salzmann, Vincent Lepetit, and Pascal Fua. Structured prediction of 3d human pose with deep neural networks. In *BMVC*, 2016. 1, 2

[41] Steve Tonneau, Andrea Del Prete, Julien Pettré, Chonhyon Park, Dinesh Manocha, and Nicolas Mansard. An efficient acyclic contact planner for multiped robots. *IEEE Transactions on Robotics*, 34(3):586–601, 2018. 2

[42] Gul Varol, Duygu Ceylan, Bryan Russell, Jimei Yang, Ersin Yumer, Ivan Laptev, and Cordelia Schmid. Bodynet: Volumetric inference of 3d human body shapes. In *ECCV*, 2018. 1, 2

[43] Xiaolin Wei and Jinxiang Chai. Videomocap: Modeling physically realistic human motion from monocular video sequences. *ACM TOG (Proc. SIGGRAPH)*, 29(4):42, 2010. 2

[44] Eric R Westervelt, Jessy W Grizzle, and Daniel E Koditschek. Hybrid zero dynamics of planar biped walkers. *IEEE transactions on automatic control*, 48(1):42–56, 2003. 2

[45] Wei Yang, Wanli Ouyang, Xiaolong Wang, Jimmy Ren, Hongsheng Li, and Xiaogang Wang. 3d human pose estimation in the wild by adversarial learning. In *CVPR*, 2018. 1, 2

[46] Andrei Zanfir, Elisabeta Marinoiu, and Cristian Sminchisescu. Monocular 3d pose and shape estimation of multiple people in natural scenes-the importance of multiple scene constraints. In *CVPR*, 2018. 2

[47] Weichen Zhang, Zhiguang Liu, Liuyang Zhou, Howard Leung, and Antoni B Chan. Martial arts, dancing and sports dataset: A challenging stereo and multi-view dataset for 3d human pose estimation. *Image and Vision Computing*, 61:22–39, 2017. 3, 5

[48] Xiaowei Zhou, Menglong Zhu, Spyridon Leonardos, Konstantinos G Derpanis, and Kostas Daniilidis. Sparseness meets deepness: 3d human pose estimation from monocular video. In *CVPR*, 2016. 2