This supplement presents addition results (§1), a description of the datasets used (§2) together with a description of the usage of data with human subjects (§2.2), and additional details of the simulation setup (§3). Please refer to our video for qualitative results at tiny.cc/diffphy.

1. Additional Results

Tab. 1 presents an ablation on window size performed using mocap data as initialization and reference trajectory rather than using the kinematic initialization. In this case, we note that a smaller window size of 480 outperforms the larger window size of 960 used in the main paper. We hypothesize that when the reference signal lacks noise, a smaller window is easier to optimize since the dimension of the problem is reduced. However, with noisy observations, a larger window is required for the method to be robust to missing or poor kinematic reconstructions.

| Window | MPJPE-G | MPJPE | MPJPE-PA |
|--------|---------|-------|---------|
| 240    | 112.8   | 75.9  | 40.1    |
| 480    | 39.4    | 33.4  | 21.9    |
| 720    | 46.1    | 42.1  | 29.4    |
| 960    | 77.8    | 68.4  | 44.9    |

Table 1. Ablation study of the optimization window size. Experiments were carried out on motion capture rather than using the kinematic initialization as input. The experiment was performed on the same Human3.6M sequences as in the ablation in the main paper. Note that when using mocap rather than noisy observations, a smaller window size is better (480 vs. 960 in main paper).

2. Datasets

We evaluate our method on the two established datasets Human3.6M [3] and AIST [7]. In addition, we evaluate our method on "real-world" internet videos.

Human3.6M. When comparing to the state-of-the-art methods, we evaluate on the Human3.6M Protocol P2 sequences while excluding the same sequences as by Xie et al. [8]. That leaves the sequences: Directions, Discussions, Greeting, Posing, Purchases, Taking Photos, Waiting, Walking, Walking Dog and Walking Together. We evaluate the motions using only camera 60457274. Similar to [8], we downsampled the Human3.6M data from 50 FPS to 25 FPS.

The ablation studies were performed on a smaller subset of four-second clips (frames 400-599) from a random camera, see tab. 2.

AIST. AIST provides dynamic dance motions not present
in Human3.6M. We evaluate our method using the pseudo-ground-truth provided by [4]. We use the first four seconds (120 frames) using a randomly selected camera from the sequences in tab. 3.

**Internet Videos.** Finally, we perform qualitative evaluation of our method on internet videos made public under creative common licences.

### 2.1. Metrics

**Total variation.** We compute the total variation of the 3d joint acceleration as a measurement of the jitter in motion. This is given as

\[
\frac{1}{T} \sum_{t \in T} \sum_{k \in K} |\dddot{x}_t^k - \dddot{x}_{t+1}^k|,
\]

where \(\dddot{x}_t^k\) is the 3d joint acceleration of joint \(k\) at time \(t\). We estimate the acceleration through finite differences.

**Foot skating.** We track unnatural foot skating artifacts by measuring the percentage of frames where either foot is “skating” along the ground. Our formulation doesn’t rely on foot contact annotations but instead heuristically detect when foot contacts occur by measuring the distance between the foot mesh and the ground-plane. A contact is defined as \(N = 10\) foot mesh vertices being within \(d\) mm of the ground-plane. For kinematics we use \(d = 5\) mm and for dynamics \(d = 1\) mm to account for the capsule approximation being smaller than the foot mesh. We define skating as a foot moving \(\geq 2\) cm between two frames while being in contact with the ground.

### 2.2. Usage of data with human subjects

In this work, we employ two established pose benchmarks that are commonly used in the field of human pose estimation. Human3.6M [3] was recorded in a laboratory setting with the permission of the actors, and AIST [7] contains “a shared database containing original street dance videos with copyright-cleared dance music. This is the first large-scale shared database focusing on street dances to promote academic research regarding Dance Information Processing”\(^1\). As for the “in-the-wild” videos, these were released under creative commons licenses granting express permission to “copy and redistribute the material in any medium or format” and “remix, transform, and build upon the material for any purpose, even commercially”. Finally, we do not intend to release these videos as part of a dataset. Instead we only use them to demonstrate our method on videos with poses and motion uncommon in laboratory captured datasets.

---

\(^1\)https://aistancedb.ongaaccel.jp/

### 3. Differentiable Physics for Human Motion

Tiny Differentiable Simulator (TDS) [2] is a C++ simulator where the data type is templetized. In our experiments, we use the scalar from the automatic differentiation (AD) framework CppAD [1] to compute the simulation gradients. That is, we compute the gradients of the loss with respect to the input control variables at each time step:

\[
\frac{\partial L}{\partial \mathbf{q}_{1:T}} = \frac{\partial L}{\partial \dot{\mathbf{q}}_{1:T}} \frac{\partial \dot{\mathbf{q}}_{1:T}}{\partial \mathbf{q}_{1:T}},
\]

where \(L\) is objective function of the trajectory optimization, \(\mathbf{q}_{1:T}\) are the simulated body’s joint positions, and \(\dot{\mathbf{q}}_{1:T}\) are the per-timestep control signal to the PD controllers in the body joints.

To speed up the optimization we implement our simulation as a fixed computational graph of the simulation rollout for a fixed number of steps and then repeatedly use it to compute the values of the gradients in (2). This greatly speeds up the optimization since the automatic differentiation framework doesn’t need to setup the computational graph for each backward pass. To that end, we make the following adaptations to TDS to make it support a fixed graph.

**Differentiation and contact points.** Since at the time of graph construction it is not known in advance which contact points will be active for particular inputs we always include all contact points into the LCP formulation. This increases the graph size based on the number of contacts considered. The issue of large graph can be address by e.g. “checkpointing” the computation as described in [6].

**Dealing with exploding gradients.** As noted in [5], gradients from differentiable simulators may explode or vanishing when the window size is large. In this work, we experimentally found it possible to mitigate the issue by set-
ting the LCP solver iterations to $K = 1$ without noticeable
degradation of reconstruction quality.

**Implementation Details** In our experiments we run TDS
with a step size of $1$ms. This is partly due to the simpler PD
controller, which requires smaller simulation steps to allow
for stable control. We set the ground-plane friction to $0.8$
and the controller gains to $k_p = 200$ and $k_d = 5$. Evalu-
ating our loss function and computing the gradients for a
window of $960$ simulation steps takes approximately $\approx 5$
seconds on a standard desktop computer with only feet con-
tacts enabled. Enabling more contacts or simulating multi-
ple objects increases memory and computation time.

**References**

[1] B. Bell. Cppad: a package for c++ algorithmic differentiation,
2021. 2

[2] Eric Heiden, David Millard, Erwin Coumans, Yizhou Sheng,
and Gaurav S Sukhatme. NeuralSim: Augmenting differenti-
able simulators with neural networks. In *Proceedings of the*
*IEEE International Conference on Robotics and Automation
(ICRA)*, 2021. 2

[3] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian
Sminchisescu. Human3.6m: Large scale datasets and predic-
tive methods for 3d human sensing in natural environments.
*IEEE Transactions on Pattern Analysis and Machine Intelli-
gence*, 36(7):1325–1339, jul 2014. 1, 2

[4] Ruilong Li, Shan Yang, David A. Ross, and Angjoo
Kanazawa. Learn to dance with aist++: Music conditioned
3d dance generation, 2021. 2

[5] Luke Metz, C. Daniel Freeman, Samuel S. Schoenholz, and
Tal Kachman. Gradients are not all you need, 2021. 2

[6] Yi-Ling Qiao, Junbang Liang, Vladlen Koltun, and Ming C
Lin. Efficient differentiable simulation of articulated bod-
ies. In *International Conference on Machine Learning*, pages
8661–8671. PMLR, 2021. 2

[7] Shuhei Tsuchida, Satoru Fukayama, Masahiro Hamasaki, and
Masataka Goto. Aist dance video database: Multi-genre,
multi-dancer, and multi-camera database for dance informa-
tion processing. In *Proceedings of the 20th International
Society for Music Information Retrieval Conference, ISMIR
2019*, pages 501–510, Delft, Netherlands, Nov. 2019. 1, 2

[8] Kevin Xie, Tingwu Wang, Umar Iqbal, Yunrong Guo, Sanja
Fidler, and Florian Shkurti. Physics-based human motion
estimation and synthesis from videos. In *Proceedings of the
IEEE/CVF International Conference on Computer Vision
(ICCV)*, pages 11532–11541, October 2021. 1