3D Human Motion Estimation via Motion Compression and Refinement Supplementary Material

Zhengyi Luo, S. Alireza Golestaneh, and Kris M. Kitani

Carnegie Mellon University, Pittsburgh, PA 15213, USA
{zluo2,sgolesta,kkitani}@cs.cmu.edu

This supplementary material is organized as follows. In Section 1, we provide qualitative results for our proposed method (*MEVA*). In Section 2, we provide complementary ablation studies. In Section 3, we will discuss the failure modes of our method. In the last section, we include the details of our implementation.

1 Qualitative Results

To best view our motion estimation and compare it with state-of-the-art, please refer to the supplementary video.

Specifically, in the supplementary video, we first show a visual demonstration of our two-stage decomposition of coarse and fine motion from a given video sequence. Next, we demonstrate the qualitative comparison between our algorithm and the prior state-of-the-art (VIBE[1]) and show that our method achieves smoother, more natural, and accurate motion estimation. Finally, we will discuss the implementation details of our method.

2 Additional Ablation Studies

2.1 Comparison with Average Filtering

While our method has significantly reduced the acceleration error and achieves state-of-the-art accuracy, one can still apply postprocessing to existing sequences to further improve the prediction. To best study its effects, here we implement a simple average filter using spherical linear interpolation (slerp) in quaternion. Specifically, for each joint rotation in SMPL q^i at timestep t, we apply slerp with a ratio of 0.5: $q_t^i = slerp(q_t^i, q_{t+1}^i, 0.5)$. Table 1 shows the result of applying averaging filtering as postprocessing on both VIBE [1] and *MEVA*. From the results, it is clear that average filtering can help reduce the acceleration error of both VIBE and *MEVA* while slightly affecting accuracy. It is also conceivable that more sophisticated methods such as solving a constrained optimization problem [2,3,4] can further improve results. Nonetheless, in the paper we only compare with feed-forward methods without any postprocessing, since postprocessing approaches are complementary to feed-forward methods and would be beneficial to all of them.

Table 1. Ablation study on average filtering. Here we show the result of applying the average filter on the output from VIBE [1] and *MEVA*.

	3DPW		
	MPJPE ↓	PA-MPJPE ↓	ACC-ERR ↓
VIBE (w/o H3.6M SMPL) [1]	91.9	57.6	25.4
VIBE (w/o H3.6M SMPL) [1] + Average Filtering	91.6	57.8	13.5
MEVA (w/o H3.6M SMPL) (ours)	**86.9**	**54.7**	11.6
MEVA (w/o H3.6M SMPL) + Average Filtering (ours)	87.6	55.5	**8.2**

2.2 Effects of a long temporal window

MEVA uses a significantly longer temporal window (90 frames) than prior art (HMMR [5]: 20 frames, VIBE[1]: 16 frames). To show that our *MEVA* framework benefits more from this setting, we retrain VIBE with a 90 frames temporal window. As shown in Table 2, using the same size temporal window, *MEVA* produces better results on all three metrics and maintains a significant advantage in acceleration error. Notice that VIBE trained with a longer temporal window shows a slight improvement against the ones that use a shorter window, validating our intuition that a longer temporal window provides a more substantial context for motion estimation. Nonetheless, our two-stage decomposition method is more effective in utilizing a longer temporal window due to its separate motion compression and refinement stages.

Table 2. Ablation study on temporal window size. Here we show the results of using different temporal windows in VIBE [1] and *MEVA*

	3DPW		
	MPJPE ↓	PA-MPJPE ↓	ACC-ERR ↓
VIBE (w/o H3.6M SMPL) + 16 frames [1]	91.9	57.6	25.4
VIBE (w/o H3.6M SMPL) + 90 frames [1]	88.1	56.6	21.2
MEVA (w/o H3.6M SMPL) + 90 frames (ours)	**86.9**	**54.7**	**11.6**

2.3 Effects of STE and VME on *MEVA*

Here we take a further look into the effects of different components (STE, VME, and MRR) of our proposed method. Notice that without the Variational Motion Estimator (VME), our method will collapse into a single-stage estimator that only relies on the SMPL regressor, which has been studied extensively in prior art. Thus, here we only study the effects of Spatial Temporal Feature Extractor (STE) and Motion Refinement Regressor (MRR). Table 3 shows the results of our framework trained without STE or MRR. Without the STE, *MEVA* obtains high accuracy but suffers from high acceleration error. This indicates that STE produces correlated features that impart the necessary temporal consistency information to MRR. We reason that without STE, even although initialized with coarse estimation from VME, MRR will be biased by the input visual features and produce a temporally inconsistent refinement pose that negatively affects the overall estimation. On the other hand, without MRR, our method

is reduced to one stage and only estimates the coarse motion. As shown in the result, using only VME will lead to an overly smoothed motion estimation and result in a higher acceleration error (underestimating movement also leads to high acceleration error).

Table 3. Ablation of MEVA components. Here we show *MEVA* trained without STE (with both VME and MRR) and without MRR (with both STE and VME).

	3DPW		
	MPJPE ↓	PA-MPJPE ↓	ACC-ERR ↓
MEVA w/o STE	89.7	55.4	29.0
MEVA w/o MRR	118.1	73.7	15.4
MEVA	**86.9**	**54.7**	**11.6**

3 Failure Modes

Although *MEVA* shows promising results in producing smooth and accurate human motion, there is still room for improvement.

3.1 Sliding window processing

MEVA processes videos using a sliding window: input video sequences are splited into chunks of 90 frames for processing. Due to the natural of this sliding window approach, inconsistency can sometimes be observed at a 3 second interval (videos are assumed to be at 30 fps). The explanation is as follows: the coarse motion estimated by VME can be quite different between each temporal window and MRR sometimes is unable to make enough adjustments to account for a smooth transition. Each temporal window also has their own STE, so the features from each window are longer correlated. Fig. 1 shows an instance of this behavior. For visual inspection, please refer to our supplement video.

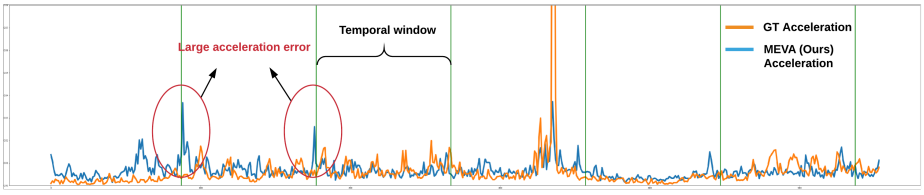

Fig. 1. Sliding window failure mode. This plot shows that at the intersection of temporal windows, *MEVA* can result in a inaccurate transition and bring a large acceleration error. Each green line in the plot marks a temporal window, and there are large spikes of acceleration error at the first two intersections.

3.2 Occluded body parts

Occluded body parts can still be challenging for *MEVA*. During occlusion, the lack of visual indicators will compel *MEVA* to rely on coarse motion estimation over the whole sequence and leads to a misscapture of detailed motion. Please refer to the supplementary video for an example.

3.3 Missing hands and face movement

Since the original SMPL[6] model does not contain joints for the hand and face, all methods using SMPL do not capture hand movements and facial expressions. Moreover, there is not enough high quality 3D data that provides hand and face annotations. A recent work [7] develops an enhanced SMPL model that jointly models body pose, hands, and face, but this model has not gained significant traction in the pose estimation community. We believe that capturing hands and face movement in motion estimation is an essential direction for future work.

4 Implementation Details

4.1 Human Motion VAE

The motion VAE's encoder, E_{vae}, is a bidirectional Gated Recurrent Unit (bi-GRU) with average pooling to obtain the temporal encoding h of the overall input motion sequence $M_W \in R^{W\times 144}$. We pass the temporal encoding h into a multilayer perceptron (MLP) with two hidden layers (1024, 512) and two heads to obtain the mean μ and variance σ for the latent code z. For the decoder D_{vae}, a forward GRU is used to decode the output motion sequence. At each time step, the GRU takes in the previous step estimation θ_{t-1} and the current latent code $z \in R^{1\times S_z}$ to output a 512 latent feature. The feature is then passed through an MLP with two hidden layers (1024, 512) to generate the reconstructed pose $\theta \in R^{1\times 144}$

4.2 Spatio-Temporal Feature Extractor

For a video input, we first preprocess the video frames using a pretrained ResNet-59 network [8]. The feature extractor outputs $f_i \in R^{2048}$ for each frame. The extracted features within the same temporal window W (we choose $W = 90$) are stacked together $[f_t]_{t=1}^{90} \in R^{90\times 2048}$ and are encoded by STE into a sequence of temporally correlated features $[f'_t]_{t=1}^{90} \in R^{90\times 2048}$. STE is a 2 layer bi-GRU with hidden size 1024 that outputs a feature encoding at each timestep. E_{motion} that shares the same architecture as E_{feat} except for the final average pooling step to come up with a latent code $z \in R^{1\times 512}$ that represents the whole motion sequence.

4.3 Motion Residual Regressor

The MRR consists of 2 fully connected layers, each with 1024 neurons. It takes in per-frame features and a set of initializing parameters (pose, shape, and camera) and iterative refines its predictions (pose, shape, and camera) for k iterations.

References

1. Kocabas, M., Athanasiou, N., Black, M.J.: Vibe: Video inference for human body pose and shape estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2020) 5253–5263
2. Huang, Y.: Towards accurate marker-less human shape and pose estimation over time. 2017 International Conference on 3D Vision (3DV) (2017) 421–430
3. Peng, X.B., Kanazawa, A., Malik, J., Abbeel, P., Levine, S.: SFV: Reinforcement learning of physical skills from videos. SIGGRAPH Asia 2018 Technical Papers, SIGGRAPH Asia 2018 **37** (2018)
4. Mehta, D., Sridhar, S., Sotnychenko, O., Rhodin, H., Shafiei, M.M.h.M.M.h.M.M.h., Seidel, H.p., Casas, D., Theobalt, C., Shafiei, M.M.h.M.M.h.M.M.h., Seidel, H.p., Xu, W.: VNect Real-time 3D Human Pose Estimation with a Single RGB Camera ACM Reference format VNect Real-time 3D Human Po.pdf. Tog **36** (2017) 1–13
5. Kanazawa, A., Zhang, J.Y., Felsen, P., Malik, J.: Learning 3d human dynamics from video. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019) 5607–5616
6. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: Smpl: a skinned multi-person linear model. ACM Trans. Graph. **34** (2015) 248:1–248:16
7. Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A.A.A., Tzionas, D., Black, M.J.: Expressive body capture: 3d hands, face, and body from a single image. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019) 10967–10977
8. Kanazawa, A., Black, M.J., Jacobs, D.W., Malik, J.: End-to-end recovery of human shape and pose. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (2018) 7122–7131