Supplementary Material

Ziniu Wan[1*] Zhengjia Li[2*] Maoqing Tian[3] Jianbo Liu[4] Shuai Yi[3] Hongsheng Li[4]
[1] Carnegie Mellon University [2] Tongji University
[3] SenseTime Research [4] Chinese University of Hong Kong
ziniuwan@andrew.cmu.edu zjli1997@tongji.edu.cn tianmaoqing@sensetime.com
liujianbo@link.cuhk.edu.hk yishuai@sensetime.com hsli@ee.cuhk.edu.hk

1. Further Training Details

Sampling strategy. For all video datasets, we first sample 128 consecutive frames. From these 128 frames, we start from a random position and sample a 16-frame clip at a equal interval of 8 frames, that we take as a video training instance.

Hybrid training of image and video. In the second training stage, we use both video datasets and image datasets for training. Specifically, for every iteration, we first feed the model with video training instance without updating the model weights, and then feed the model with image training instance of the same batch size. We use the accumulated gradients of these two forward propagation to update the model weights.

CNN backbone. All the input images of CNN backbone are resized to size 224×224. Following ViT [3], we make three modifications to ResNet-50 [4]: 1. Replace Batch Normalization with Group Batch Normalization [15]. 2. Remove the fourth stage and increase the number of blocks in the third stage to 9. As a result, the number of blocks per stage changes from [3, 4, 6, 3] to [3, 4, 9]. 3. Remove the global pooling layer.

STE. Following ViT [3], the resolution of the output feature of CNN backbone is 14×14, which are then flattened to a sequence of length 196 to fed into STE. The feature dimension of STE is 768. Moreover, 6 STE Parallel Blocks are stacked, and each block has 12 heads.

KTD. Following SPIN [10], we map the dimension of output feature of STE from 768 to 1024 (hidden dim) using a fully connected layer $W_{\text{hid}} \in \mathbb{R}^{1024\times768}$.

Loss weights. We split L_{SMPL} mentioned in Section 3.2 into two parts: L_{shape} and L_{pose}. Then the training loss is formulated as following:

$$L = L_{3D} + L_{2D} + L_{shape} + L_{pose} + L_{NORM} \tag{1}$$

where each term is calculated as:

$$\begin{aligned} L_{3D} &= \sum_{k=1}^{K} \left\| J_{3d}^{k} - J_{3dgt}^{k} \right\|_2, \\ L_{2D} &= \sum_{k=1}^{K} \left\| J_{2d}^{k} - J_{2dgt}^{k} \right\|_2 \\ L_{shape} &= \|\vec{\beta} - \vec{\beta}_{gt}\|_2 \\ L_{pose} &= \|\vec{\theta} - \vec{\theta}_{gt}\|_2 \\ L_{NORM} &= \|\vec{\beta}\|_2 + \|\vec{\theta}\|_2 \end{aligned} \tag{2}$$

We use different weight coefficients for each term in the loss. The coefficients of L_{3D}, L_{2D}, L_{shape}, L_{pose} and L_{NORM} are 600.0, 300.0, 0.06, 60.0 and 1.0 respectively.

2. Ablation Study of Datasets

The datasets used in our methods differ from other methods, such as SPIN [10], MEVA [12] and VIBE [9], whose settings are described in Table 1. In this section, we use the exactly same datasets as each of them and report the performances on 3DPW [14] in Table 2. $\text{MAED}_{\text{spin}}$/$\text{MAED}_{\text{meva}}$/$\text{MAED}_{\text{vibe}}$ represents MAED with the same datasets as SPIN/MEVA/VIBE. It is worth noting that VIBE and MEVA adopt the pretrained CNN from SPIN [10] and HMR [7] respectively. In order to achieve fair comparison, we also use the same datasets as SPIN and HMR to pretrain MAED in the first training stage. We can see that our method still outperforms them even with the same datasets, which demonstrates the robustness of our method.

3. Influence of Layer Number of STE

Figure 1 summarizes the results of MEAD with different layer numbers of STE. We can observe that STE stably obtains higher performance gains from more stacked layers. However, when the layer number of the model exceeds 6, the performance improvements brought by increased layer number will be minor. Considering the trade-off between speed and accuracy, we empirically choose 6 layers.

*Equal Contribution.

Method	InstaVariety [8]	PoseTrack [1]	PennAction [16]	3DPW [14]	MPI-INF-3DHP [13]	Human3.6M [5]	COCO [11]	LSP-Extended [6]	MPII [2]
HMR [7]					✓	✓	✓	✓	✓
SPIN [10]					✓	✓	✓	✓	✓
MEVA [12]	✓		✓	✓	✓				
VIBE [9]	✓	✓	✓	✓	✓	✓			
MAED	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 1: Dataset settings of different methods.

Method	PA-MPJPE	MPJPE
HMR [7]	81.3	130.0
SPIN [10]	59.2	96.9
MEVA [12]	54.7	86.9
VIBE [9]	51.9	82.9
$\text{MAED}_{\text{spin}}$	51.9	90.7
$\text{MAED}_{\text{meva}}$	46.1	77.5
$\text{MAED}_{\text{vibe}}$	46.8	80.2
MAED	45.7	79.1

Table 2: MAED with different dataset settings.

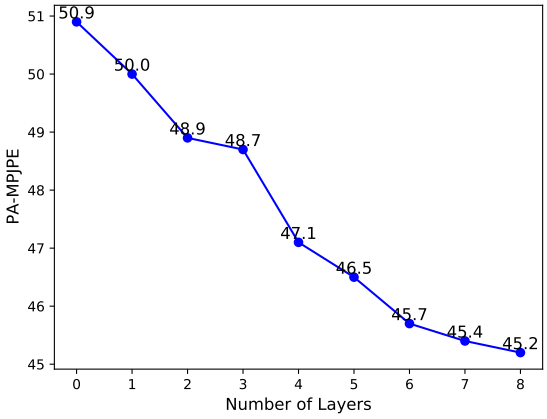


Figure 1: Analytical experiment results with different layer numbers.

4. Computation Overhead

MAED significantly improves the accuracy of 3D pose estimation but also incurs non-negligible computation overhead. Compared to VIBE [9], our method achieves an improvement in PA-MPJPE even with only 1 STE block (from 51.9 to 50.0). However, with 1 STE block, MAED contains 91.1 GFLOPS, which is a non-negligible overhead compared to VIBE's 67.6 GFLOPS. Furthermore, despite yielding stable improvement with increasing STE blocks, each STE block brings about extra 22 GFLOPS, resulting in 201.1 GFLOPS in total for the 6-block MAED, which is almost 3x that of VIBE. Therefore, we propose to feed only part of a input video sequence into the model and interpolate the rest to reduce the computation overhead for a single forward propagation. Specifically, for a 16-frame input sequence, we sample 8 frames at equal intervals as input, and obtain the estimation of the remaining 8 frames through interpolation. This reduces the GFLOPS of a single forward by half without significantly reducing accuracy, as is shown in Table 3.

Method	PA-MPJPE	GFLOPS	input
VIBE [9]	51.9	67.6	16f
$\text{MAED}_{\text{1-block}}$	50.0	91.1	16f
$\text{MAED}_{\text{6-block}}$	48.2	100.4	8f
$\text{MAED}_{\text{6-block}}$	45.7	201.1	16f

Table 3: Computation overhead comparison with VIBE [9].

5. More Visualization Comparison

We compare our results with VIBE's [9] under some challenging scenario, as is shown in Figure 2.

References

[1] Mykhaylo Andriluka, Umar Iqbal, Eldar Insafutdinov, Leonid Pishchulin, Anton Milan, Juergen Gall, and Bernt Schiele. Posetrack: A benchmark for human pose estimation and tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5167–5176, 2018. 2

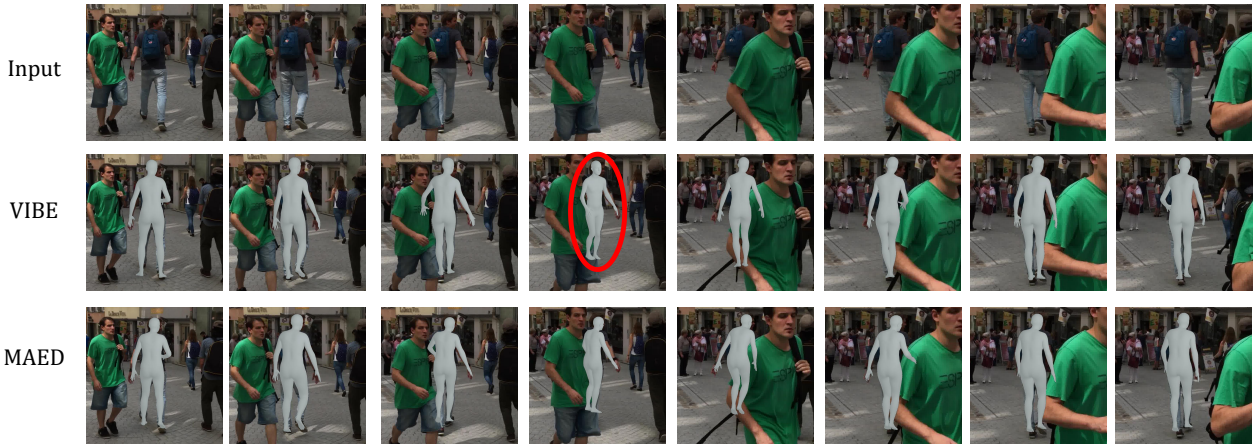
[2] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *Proceedings of the IEEE Conference on computer Vision and Pattern Recognition*, pages 3686–3693, 2014. 2

[3] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[5] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environ-

(a) In this challenging outdoor scenario, VIBE outputs suboptimal prediction in the first four frames, where a chair and a table in the foreground obscure most of the human body. The mesh largely shifts from the human body. While the results of MAED are cosistently accurate.



(a) Occluded by a passerby in this crowded scence, an unexpected jitter appears in the fourth frame of the prediction of VIBE. In comparison, our method utilizes the rich temporal information to infer current prediction, and hence outputs coherent meshs.

Figure 2: Visualization Comparison with VIBE [9].

ments. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1325–1339, 2013. 2

[6] Sam Johnson and Mark Everingham. Learning effective human pose estimation from inaccurate annotation. In *CVPR 2011*, pages 1465–1472. IEEE, 2011. 2

[7] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7122–7131, 2018. 1, 2

[8] Angjoo Kanazawa, Jason Y Zhang, Panna Felsen, and Jitendra Malik. Learning 3d human dynamics from video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5614–5623, 2019. 2

[9] Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. Vibe: Video inference for human body pose and shape estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5253–5263, 2020. 1, 2, 3

[10] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2252–2261, 2019. 1, 2

[11] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 2

[12] Zhengyi Luo, S Alireza Golestaneh, and Kris M Kitani. 3d human motion estimation via motion compression and refinement. In *Proceedings of the Asian Conference on Computer Vision*, 2020. 1, 2

[13] 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 *2017 international conference on 3D vision (3DV)*, pages 506–516. IEEE, 2017. 2

[14] Timo von Marcard, Roberto Henschel, Michael J Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 601–617, 2018. 1, 2

[15] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018. 1

[16] Weiyu Zhang, Menglong Zhu, and Konstantinos G Derpanis. From actemes to action: A strongly-supervised representation for detailed action understanding. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2248–2255, 2013. 2