OpenCapBench: Supplementary materials

Finetuning details

All SynthPose models were finetuned using the mmpose framework [1], for 30 epochs each. The fine-tuning process utilized the Adam optimizer [3] and a multi-step learning rate scheduler. The learning rate started at 5×10^{-3} and was reduced by a factor of 10 at epochs 15 and 20. The length of an epoch was determined by the size of the largest dataset. Throughout finetuning, we cycled through datasets using random index permutations for each cycle, ensuring an equal distribution of training samples from each dataset within each epoch and each batch when possible. For example, if a training batch is of size 4 and the aggregated dataset contains 4 datasets, each batch contains one sample from each dataset.

Importance of pretraining

Although we briefly discussed the importance of utilizing weights from models trained on popular datasets like COCO to finetune SynthPose models in the main paper, we present a quantitative ablation study below.

	PCK Precision(↑)		
	@0.05	@0.10	@0.20
SynthPose trained from scratch			
HRNet-W48 [5]	0.843	0.929	0.967
ViTPose-B [4]	0.738	0.851	0.912
ViTPose-H [4]	0.644	0.769	0.853
SynthPose finetuned from COCO pretrained weights			
HRNet-W48 [5]	0.892	0.958	0.982
ViTPose-B [4]	0.859	0.941	0.971
ViTPose-H [4]	**0.903**	**0.966**	**0.985**

Table 1. Ablation study on the RICH [2] test set, illustrating the importance of leveraging pretrained weights to fine-tune SynthPose models. Additionally, we observe that the significance of using pretrained weights increases with the number of parameters in a model.

Visualizations

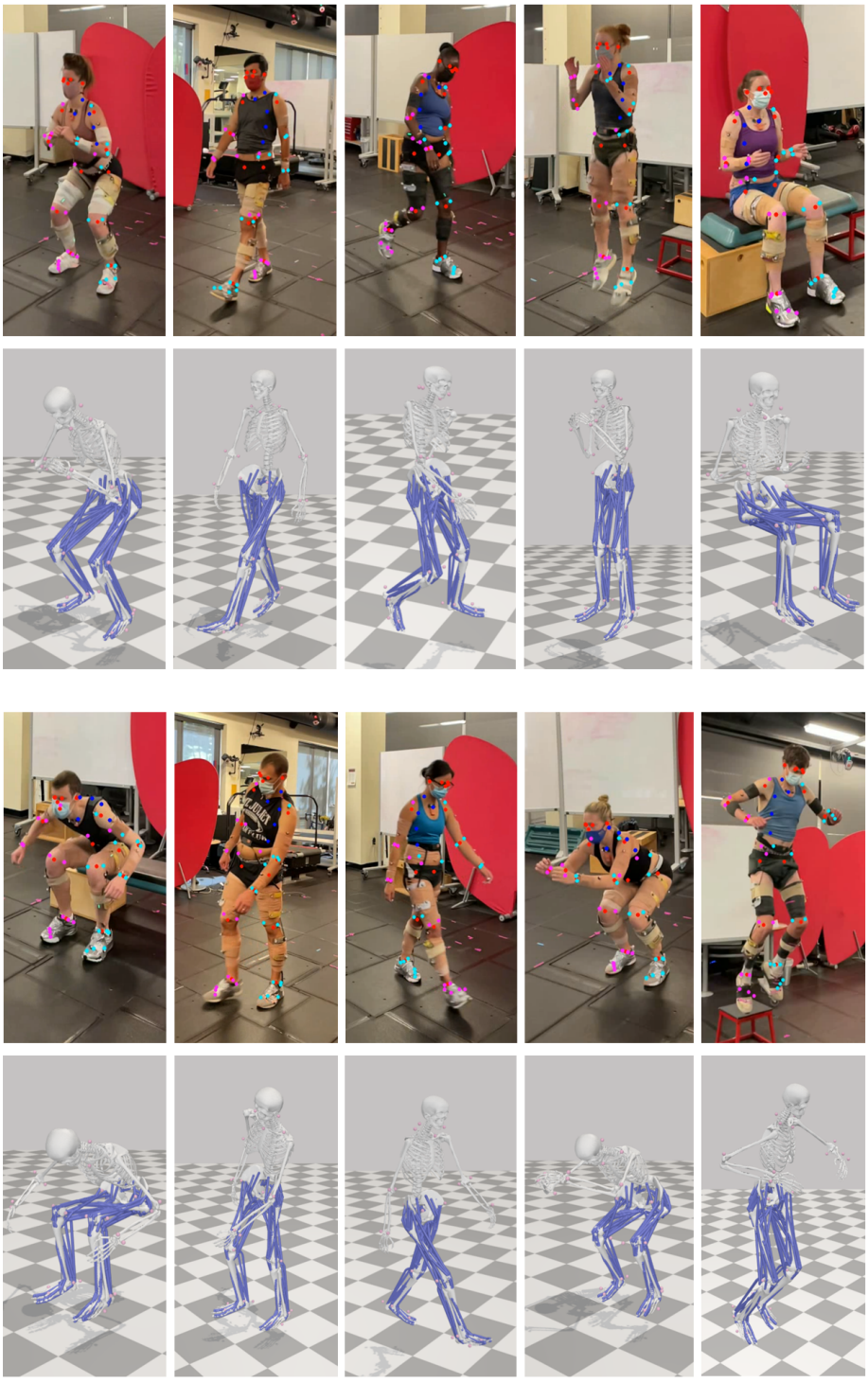

Figure 1. Samples of Synthpose MoCap marker predictions on each subject of the OpenCap dataset, with their corresponding OpenSim kinematics output. The pink markers represent anatomical markers on the right side of the body, the cyan markers represent those on the left side, the blue markers represent markers in the center of the body, and the red markers represent markers in the COCO format. The SynthPose model used is HRNet48 fine-tuned on the entire aggregated dataset to predict the subset of markers defined in the main submission.

References

[1] MMPose Contributors. Openmmlab pose estimation toolbox and benchmark. https://github.com/open-mmlab/mmpose, 2020. 1

[2] Chun-Hao P. Huang, Hongwei Yi, Markus Höschle, Matvey Safroshkin, Tsvetelina Alexiadis, Senya Polikovsky, Daniel Scharstein, and Michael J. Black. Capturing and inferring dense full-body human-scene contact. In *Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 13274–13285, June 2022. 1

[3] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. 1

[4] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. ViTPose: Simple vision transformer baselines for human pose estimation. In *Advances in Neural Information Processing Systems*, 2022. 1

[5] Feng Zhang, Xiatian Zhu, Hanbin Dai, Mao Ye, and Ce Zhu. Distribution-aware coordinate representation for human pose estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1