ProxyCap: Real-time Monocular Full-body Capture in World Space via Human-Centric Proxy-to-Motion Learning *Supplementary Material**

Anonymous CVPR submission

Paper ID *****

1. Implementation Details

1.1. Human-to-World Coordinate Transformation

We estimate the local human pose of each frame in our proxy-to-motion network, then we transform it into the global world space. In the first frame, the accumulated translation of human in x-z plane t_{xz} is set to zero. For the later human-space estimations, we firstly rotate the front axis of camera in human space to align the y-z plane by R_{front}. Denote the target parameters of camera in world space as R_W, T_W and the predicted parameters in human space as R_H, T_H. We have $R_W = R_H \cdot R_{front}$, $T_W = -R_W \cdot (R^T_{front} \cdot (-R^T_H \cdot T_H) + t_{xz})$. And the human orientation should also be rotate in the same time to maintain relative stillness: $\theta_W(root) = R^T_{front} \cdot \theta_H(root)$. The world translation can be calculated by $t_W = R^T_{front} \cdot (t_H + J_{root}) - J_{root} + t_{xz}$. Here J_{root} is the root joint of SMPL model in T-pose to eliminate the error resulted from the root-misalignment with the original point of SMPL model. Finally, the accumulated translation of human in x-z plane t_{xz} is updated by $t^{t+1}_{xz} = t^t_{xz} + R^T_{front} \cdot t_H$.

1.2. Partial Dilated Convolution

It should be noted that the hand area, being an extremity, is more prone to being affected by heavy motion blur and severe occlusions, resulting in missing detections. Simply setting the corrupted data to zero is not a viable solution as the original convolution kernel is unable to distinguish between normal data and corrupted data, leading to a significant reduction in performance as noise is propagated through the network layers.

To overcome this challenge, we employ partial convolution [1] to enhance our 1D dilated convolution framework. As illustrated in Fig. 1, rather than indiscriminately processing the input signals as in the original convolution operator, we utilize a mask-weighted partial convolution to selectively exclude corrupted data from the inputs. This enhances the robustness of hand recovery in scenarios involving fast movement and heavy occlusion. Specifically, the latent code X_0 is initially set as the concatenation of the (x, y) coordinates of the J joints for each frame $f \in 1, 2, ..., L$, while the mask M_0 is initialized as a binary variable with values of 0 or 1, corresponding to the detection confidence. Then we integrated the 2D partial convolution operation and mask update function from [1] into our 1D dilated convolution framework:

$$\begin{cases} M_{k+1} = I\left(sum(M_k) > 0\right) \\ X_{k+1} = M_{k+1}\left(W_k^\top \left(X_k \odot M_k\right) \frac{size(M_k)}{sum(M_k)} + b_k\right) \end{cases} \tag{1}$$

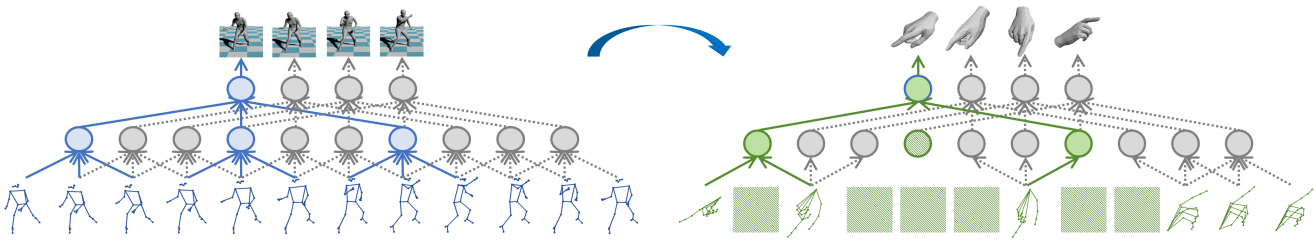
where W and b denotes the convolution filter weights and bias at layer k, and $\odot$ denotes the element-wise multiplication. Furthermore, in the training procedure, half of the sequential inputs are randomly masked to simulate detection failures that may occur in a deployment environment.

1.3. Training

We train our network using the Adam optimizer with a learning rate of 1e-4 and a batch size of 1024 in NVIDIA RTX 4090. We adopt a two-stage training scheme. Firstly, we train our proxy-to-motion initialization network (Sec. 4.2) for 50 epochs. Subsequently, we fix the weights of the motion recovery network and train the neural descent network (Sec. 4.3) for another 50 epochs.

1.4. Proxy Dataset

We conduct the training process on our synthetic proxy dataset. The 3D body rotational motions of the training set are sampled from AMASS [3]: [ACCAD, BMLmovi, BMLrub, CMU, CNRS, DFaust, EKUT, Eyes Japan Dataset, GRAB, HDM05, HumanEva, KIT, MoSh, PosePrior, SFU, SOMA, TCDHands, TotalCapture, WEIZMANN] and [SSM, Transitions] are for testing. Otherwise, for the generation of hand motions, we adopt the same dataset division of InterHand [4]. Then We animate the SMPL-X mesh and generate virtual cameras to obtain the



(a) original dilated conv **(b) partial dilated conv**

Figure 1. Illustration of human recovery. The left part (a) depicts the original dilated convolution backbone of [5], while the right part (b) illustrates our proposed partial dilated convolution architecture. Our approach selectively excludes corrupted data from input signals, allowing us to extract motion features more accurately and effectively. Specifically, detection failure (denoted as green mosaic squares) may occur during extremely fast motion or severe occlusion situations, while our architecture will cut off the connection from corrupted data to prevent disturbance transfer during network forward processing.

pseudo 2D labels.

2. Computational Complexity

In this section, we compare the inference speed of our method. Our real-time monocular full-body capture system can be implemented on a single Laptop (NVIDIA RTX 4060 GPU). Specifically, for the 2D pose estimator, we leverage the off-the-shelf Mediapipe [2] and MMPose and re-implement it on the NVIDIA TensorRT platform. We report the inference time of each module in Table. 1.

Table 1. Time costs of each module in our pipeline.

Network	Input	Speed
Body Crop Net	$224 \times 224 \times 3$	2ms
Body Landmark Net	$384 \times 288 \times 3$	5ms
Hand Crop Net	$2 \times 256 \times 256 \times 3$	1.5ms
Hand Landmark Net	$256 \times 256 \times 3$	2ms
Pose Initialization Net	$81 \times 67 \times 2$	3ms
Neural Descent Net	\	10ms

References

[1] Guilin Liu, Fitsum A. Reda, Kevin J. Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 1

[2] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, et al. Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*, 2019. 2

[3] Naureen Mahmood, Nima Ghorbani, Nikolaus F Troje, Gerard Pons-Moll, and Michael J Black. AMASS: Archive of motion capture as surface shapes. In *ICCV*, pages 5442–5451, 2019. 1

[4] Gyeongsik Moon, Shoou-I Yu, He Wen, Takaaki Shiratori, and Kyoung Mu Lee. InterHand2.6M: A dataset and baseline for 3D interacting hand pose estimation from a single RGB image. In *ECCV*, pages 548–564. Springer, 2020. 1

[5] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3D human pose estimation in video with temporal convolutions and semi-supervised training. pages 7753–7762, 2019. 2