XNect: Real-time Multi-Person 3D Motion Capture with a Single RGB Camera

DUSHYANT MEHTA[1,2], OLEKSANDR SOTNYCHENKO[1,2], FRANZISKA MUELLER[1,2], WEIPENG XU[1,2], MOHAMED ELGHARIB[1,2], PASCAL FUA[3], HANS-PETER SEIDEL[1,2], HELGE RHODIN[3,4], GERARD PONS-MOLL[1,2], CHRISTIAN THEOBALT[1,2]

[1]Max Planck Institute for Informatics, [2]Saarland Informatics Campus, [3]EPFL, [4]University of British Columbia

ACM Reference Format:
Dushyant Mehta, Oleksandr Sotnychenko, Franziska Mueller, Weipeng Xu, Mohamed Elgharib, Pascal Fua, Hans-Peter Seidel, Helge Rhodin, Gerard Pons-Moll, and Christian Theobalt . 2020. XNect: Real-time Multi-Person 3D Motion Capture with a Single RGB Camera. *ACM Trans. Graph.* 39, 4, Article 1 (July 2020), 7 pages. https://doi.org/10.1145/3386569.3392410

SUPPLEMENTAL DOCUMENT

Here we present additional qualitative results of our approach, ablation studies of design variations of SelecSLS architecture, further ablation studies of our proposed pose representation, and additional details about *Stage III* of our system. Refer to the main manuscript, the accompanying video, and the project website (http://gvv.mpi-inf.mpg.de/projects/XNect/) for further details and results.

1 *SELECSLS NET* DESIGN EVALUATION

Figure 1 shows variants of the overall architecture of the proposed *SelecSLS Net* that were considered. The architecture is parameterized by the type of module (*SelecSLS* concatenation-skip *CS* vs addition-skip *AS*), the stride of the module (s), the intermediate features in the module (k), cross- module skip connectivity (previous module or first module in the level), and number of outputs of the module (n_o (B)ase case). With the aim to promote information flow in the network, we also consider (W)ider n_o at transitions in spatial resolution. All 3×3 convolutions with more than 96 outputs use a group size of 2, and those with more than 192 outputs use a group size of 4.

We experimentally determine the best network design by testing the *Stage I* network with a *SelecSLS Net* core on 2D multi-person pose estimation, i.e., only using the 2D branch, which plays an integral role in the overall pipeline. Our conclusions transfer to the full *Stage I* network, as further evidenced in Section 7.4 in the main manuscript.

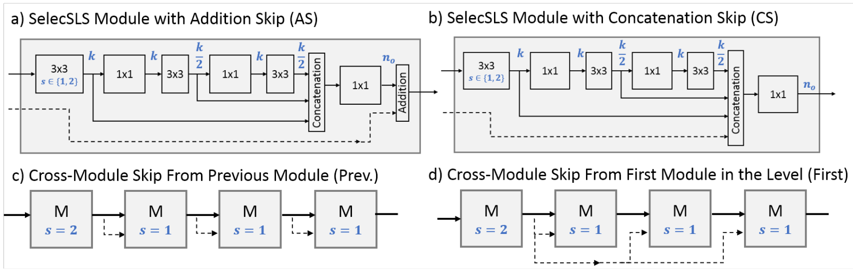
Table 1. Evaluation of possible baseline architecture choices for the core network. The networks are trained on MPI [2014] and LSP [2010; 2011] single person 2D pose datasets, and evaluated on LSP testset. The inference speed ratios are with respect to ResNet-50 forward pass time for 320×320 pixel images on an NVIDIA K80 GPU, using [Jolibrain Caffe Fork 2018] with optimized depthwise convolution implementation.

Core Network	PCK	FP Speed Ratio
MobileNetV2 1.0x [2018]	85	1.78
MobileNetV2 1.3x [2018]	86	1.51
Xception [2017]	81	0.67
InceptionV3 [2016]	88	0.96
ResNet-34 [2016]	89	1.27
ResNet-50 [2016]	89	1.00

As mentioned in Section 6.2 in the main manuscript, and shown in Table 1, for GPU-based deployment ResNet architectures provide a better or comparable speed–accuracy tradeoff to various parameter-efficient depthwise-convolution based designs. Thus, we compare against ResNet-50 and ResNet-34 architectures as core networks to establish the appropriate baselines. For ResNet, we keep the network until the first residual module in level-5 and remove striding from level-5. We evaluate on a held-out 1000 frame subset of the MS-COCO validation set, and report the Average Precision (AP) and Recall (AR), as well as inference time on different hardware in Table 2. Using the *AS* module with *Prev* connectivity and n_o(B) outputs for modules, the performance as well as the inference time on an Nvidia K80 GPU is close to that of ResNet-34. Using *CS* instead of addition-skip significantly improves the average precision from 47.0 to 47.6, and the average recall from 51.7 to 52.6. Switching the number of module outputs to the wider n_o(W) scheme leads to further improvement in AP and AR, at a slight increase in inference time. Using *First* connectivity further improves performance, namely to 48.6 AP and 53.3 AR, reaching close to ResNet-50 in AP (48.8) and performing slightly better with regard to AR (53.2). Still our new design has a 1.4-1.8× faster inference time across all devices. We also evaluate the publicly available model of [Cao et al. 2017] on the same validation subset. Their multi-stage network is 11 percentage points better on AP and AR than our network, at the price of being $10 - 20\times$ slower. The follow-up versions [Cao et al. 2019] are $\approx 2\times$ faster on the GPU and $\approx 5\times$ slower on the CPU, and $4-5$ percentage points better on AP than the original ([Cao et al. 2017]), though it still remains 5× slower than our network on a GPU, and $\approx 80\times$ slower than our network on a CPU.

Thus, of the different possible designs of the *SelecSLS* module, and the inter-module skip connectivity choices, the best design for *SelecSLS Net* is the one with concatenation-skip modules, cross-module skip connectivity to the first module in the level, and n_o(W) scheme for module outputs. Refer to Section 7.4 in the main manuscript for further comparisons of our architecture against ResNet-50 and ResNet-34 baselines on single-person and multi-person 3D pose benchmarks.

Authors' addresses: Dushyant Mehta, Oleksandr Sotnychenko, Franziska Mueller, Weipeng Xu, Mohamed Elgharib, Hans-Peter Seidel, Gerard Pons-Moll, and Christian Theobalt, MPI For Informatics, Saarland Informatics Campus; Pascal Fua, École Polytechnique Fédérale de Lausanne; Helge Rhodin, University of British Columbia.



Level	Output	*SelecSLS*	Stride		Cross-Module	n_o	
	Resolution	Module	s	k	Skip Conn.	(B)	(W)
L0	w/2 x h/2	Conv. 3x3	2	-	-	32	32
L1	w/4 x h/4	CS/AS	2	64	No	64	64
	w/4 x h/4	CS/AS	1	64	Prev/First	64	128
L2	w/8 x h/8	CS/AS	2	128	No	128	128
	w/8 x h/8	CS/AS	1	128	Prev/First	128	128
	w/8 x h/8	CS/AS	1	128	Prev/First	128	288
L3	w/16 x h/16	CS/AS	2	288	No	288	288
	w/16 x h/16	CS/AS	1	288	Prev/First	288	288
	w/16 x h/16	CS/AS	1	288	Prev/First	288	288
	w/16 x h/16	CS/As	1	288	Prev/First	416	416

Fig. 1. Variants of *SelecSLS* module design (a) and (b). Both share a common design comprised of interleaved 1×1 and 3×3 convolutions, with different ways of handling cross-module skip connections internally: (a) as additive-skip connections, or (b) as concatenative-skip connections. The cross module skip connections can themselves come either from the previous module (c) or from the first module which outputs features at a particular spatial resolution (d). In addition to the different skip connectivity choices, our design is parameterized by module stride (s), the number of intermediate features (k), and the number of module ouputs n_o. The table on the right shows the network levels, overall number of modules, number of intermediate features k, and the spatial resolution of features of the network designs we evaluate in Section 1. The design choices evaluated are the type of module (additive skip *AS* vs concatenation skip *CS*), the type of cross module skip connectivity (From previous module (*Prev*) or first module (*First* in the level), and the scheme for the number of outputs of modules n_o ((B)ase or (W)ide).

Table 2. Evaluation of design decisions for first stage of our system. We evaluate different core networks with the 2D pose branch on a subset of validation frames of MS COCO dataset. Also reported are the forward pass timings of the core network and the 2D pose branch on different GPUs (K80, TitanX (Pascal)) as well as Xeon E5-1607 CPU on 512×320 pixel input. We also evaluate the publicly available model of [Cao et al. 2017] on the same subset of validation frames.

	FP Time								
Core Network	**K80**	**TitanX**	**CPU**	**AP**	**$\text{AP}_{0.5}$**	**$\text{AP}_{0.75}$**	**AR**	**$\text{AR}_{0.5}$**	**$\text{AR}_{0.75}$**
ResNet-50	35.7ms	9.6ms	349ms	48.8	74.6	52.1	53.2	76.8	56.3
ResNet-34	25.7ms	5.7ms	269ms	46.4	72.7	47.3	51.3	75.2	52.8
Ours									
Add-Skip Prev. (B)	24.5ms	6.5ms	167ms	47.0	73.4	49.7	51.7	75.6	54.5
Conc.-Skip Prev. (B)	24.3ms	6.3ms	172ms	47.6	73.3	50.7	52.6	76.1	55.6
Conc.-Skip Prev. (W)	25.0ms	6.7ms	184ms	48.3	74.4	51.1	52.9	76.5	55.7
Conc.-Skip First (W)	25.0ms	6.7ms	184ms	48.6	74.2	52.2	53.3	76.6	56.7
[Cao et al. 2017]	243ms	73.4ms	3660ms	58.0	79.5	62.9	62.1	81.2	66.5

Table 3. Results of SelecSLSNet on image classification on the ImageNet dataset. The top-1 and top-5 accuracy on the Imagenet validation set is shown, as well the maximum batch size of 224×224 pixel images that can be run in inference mode on an Nvidia V100 16GB card, with FP16 compute. Also shown is the peak throughput obtained with each network, and the batch size of peak throughput (in brackets). SelecSLSNet achieves comparable performance to ResNet-50 [He et al. 2016], while being 1.3−1.4× faster, and with a much smaller memory footprint.

	Speed	**Maximum**	**Accuracy**	
	(Images / sec)	**Batch Size**	**top-1**	**top-5**
ResNet-50	2700	1200 (1024)	78.5	94.3
SelecSLSNet	3900	2000 (1800)	78.4	98.1

2 SELECSLSNET ON IMAGE CLASSIFICATION

To demonstrate the efficacy of our proposed architecture on tasks beyond 2D and 3D body pose estimation, we train a variant of SelecSLSNet on ImageNet [Russakovsky et al. 2015], a large scale image classification dataset. The network architecture is shown in Figure 2, and the results are shown in Table 3.

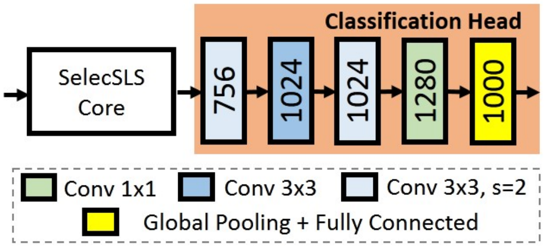


Fig. 2. For experiments on the image classification task on ImageNet [Russakovsky et al. 2015], we use the same core architecture design as SelecSLSNet for the multi-person task. Group convolutions are not used in the core network for this task. Inplace of the '2D Branch' and '3D Branch' described in Fig. 2 in the main manuscript, we use a 5 layer network as the classification head. As with the various 2D and 3D pose estimation tasks described previously, the network matches the accuracy of ResNet-50 on ImageNet as well, while being 1.3 − 1.4× faster.

3 ABLATION OF INPUT TO *STAGE II*

We evaluate variants of *Stage II* network taking different subsets of outputs from *Stage I* as input. We compare the *Stage II* output, without *Stage III* on MPI-INF-3DHP single person benchmark.On the single person benchmark (Table 4), using only the 2D pose from the 2D branch as input to Stage II, without having trained the 3D branch for Stage I, results in a 3DPCK of 76.0. When using 2D pose from a network with a 3D branch, trained additionally on MuCo-3DHP dataset, we see a minor performance decrease to 75.5 3DPCK. Though it comes with a performance improvement on challenging pose classes such as 'Sitting' and 'On The Floor' which are under-represrented in MSCOCO. Adding other components on top of 2D pose, such as the joint detection confidences C_k, and output features from the 3D branch $\{l_{j,k}\}_{j=1}^{J}$ (as described in Section 4.1.2 in the main manuscript) leads to consistent improvement as more components are subsequently used as input to *Stage II.* Using joint detection confidences C_k with 2D pose increases the accuracy to 77.2 3DPCK, and incorporating 3D pose features $\{l_{j,k}\}_{j=1}^{J}$ increases

Table 4. Evaluation of the impact of the different components from *Stage I* that form the input to *Stage II*. The method is trained for multi-person pose estimation and evaluated on the MPI-INF-3DHP [2017a] single person 3D pose benchmark. The components evaluated are the 2D pose predictions P_k^{2D}, the body joint confidences C_k, and the set of extracted 3D pose encodings $\{l_{j,k}\}_{j=1}^{J}$. Metrics used are: 3D percentage of correct keypoints (**3DPCK**, higher is better), area under the curve (**AUC**, higher is better) and mean 3D joint position error (**MJPE**, lower is better). Also shown are the results with channel-dense supervision of 3D pose encodings $\{l_{j,k}\}_{j=1}^{J}$, as well as evaluation of *Stage III* output.

	3DPCK				
Stage II	**Stand**		**On The**	**Total**	
Input	**/Walk**	**Sitt.**	**Floor**	**3DPCK**	**AUC**
P_k^{2D} (2D Branch Only)	86.4	76.3	44.9	76.0	42.1
P_k^{2D}	79.8	78.4	58.5	75.5	41.3
$P_k^{2D} + C_k$	85.9	79.4	58.7	77.2	42.2
$P_k^{2D} + C_k + \{l_{j,k}\}_{j=1}^{J}$	**88.4**	**85.8**	**70.7**	**82.8**	**45.3**
Channel-Dense $\{l_{j,k}\}_{j=1}^{J}$ Supervision					
$P_k^{2D} + C_k + \{l_{j,k}\}_{j=1}^{J}$	87.0	83.6	61.5	80.1	43.3

the accuracy to 82.8 3DPCK, and both lead to improvements in AUC and MPJPE as well as improvements for both simpler poses such as upright 'Standing/walking' as well as more difficult poses such as 'Sitting' and 'On the Floor'

This shows the limitations of 2D-3D 'lifing' approaches, and demonstrates that incorporating additional information, such as the joint detection confidences, and our proposed 3D pose encoding that uses local kinematic context (channel-sparse supervision) improve the pose performance, leads to significant improvements in 3D pose accuracy.

4 SEQUENTIAL MOTION CAPTURE (*STAGE III*): ADDITIONAL DETAILS

Absolute Height Calibration. As mentioned in the main document, to allow more accurate camera relative localization, we can optionally utilize the ground plane as reference geometry. First, we determine the camera relative position of a person by shooting a ray from the camera origin through the person's foot detection in 2D and computing its intersection with the ground plane. The subject height, h_k, is then the distance from the ground plane to the intersection point of a virtual billboard placed at the determined foot position and the view ray through the detected head position. Because we want to capture dynamic motions such as jumping, running, and partial (self-)occlusions, we cannot assume that the ankle is visible and touches the ground at every frame. Instead, we use this strategy only once when the person appears. As shown in the accompanying video, such a height calibration strategy allows reliable camera-relative localization of subjects in the scene even when they are not in contact with the ground plane.

In practice, we compute intrinsic and extrinsic camera parameters once prior to recording using checkerboard calibration. Other object-free calibration approaches would be feasible alternatives [Yang and Zhou 2018; Zanfir et al. 2018].

Inverse Kinematics Tracking Error Recovery: Since we use gradient descent for optimizing the fitting energy, we can monitor the gradients of E_{3D} and E_{lim} terms in $\mathcal{E}(\theta_1[t], \cdots, \theta_K[t])$ to identify when tracking has failed, either due to a failure to correctly match subjects to tracks because of similar appearance and pose, or when the fitting gets stuck in a local minimum. When the gradients associated with these terms exceed a certain threshold for a subject for 30 frames, the identity and pose track of the subject is re-initialized.

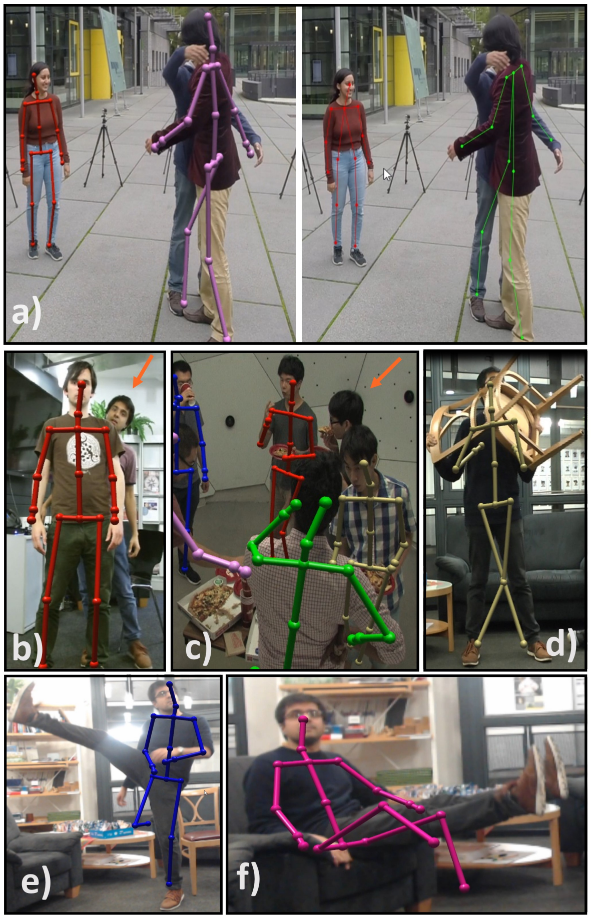


Fig. 3. **Limitations:** a)3D pose inaccuracy due to 2D pose limb confusion, b),c) Person not detected due to neck occlusion and extreme occlusion, d) Body orientation confusion due to occluded face e),f) Unreliable pose estimates for poses drastically different from the training data.

5 MORE QUALITATIVE RESULTS

Limitations: Figure 3 shows visual examples of the limitations of our approach, as discussed in Section 8 in the main manuscript. Our system mispredicts when the underlying 2D pose prediction mispredicts limb associations across subject. When the neck of the subject is occluded, we treat the subject as not present, even when the rest of the body is visible. This could be handled by using multiple reference joints on the body, instead of just the neck. Also,

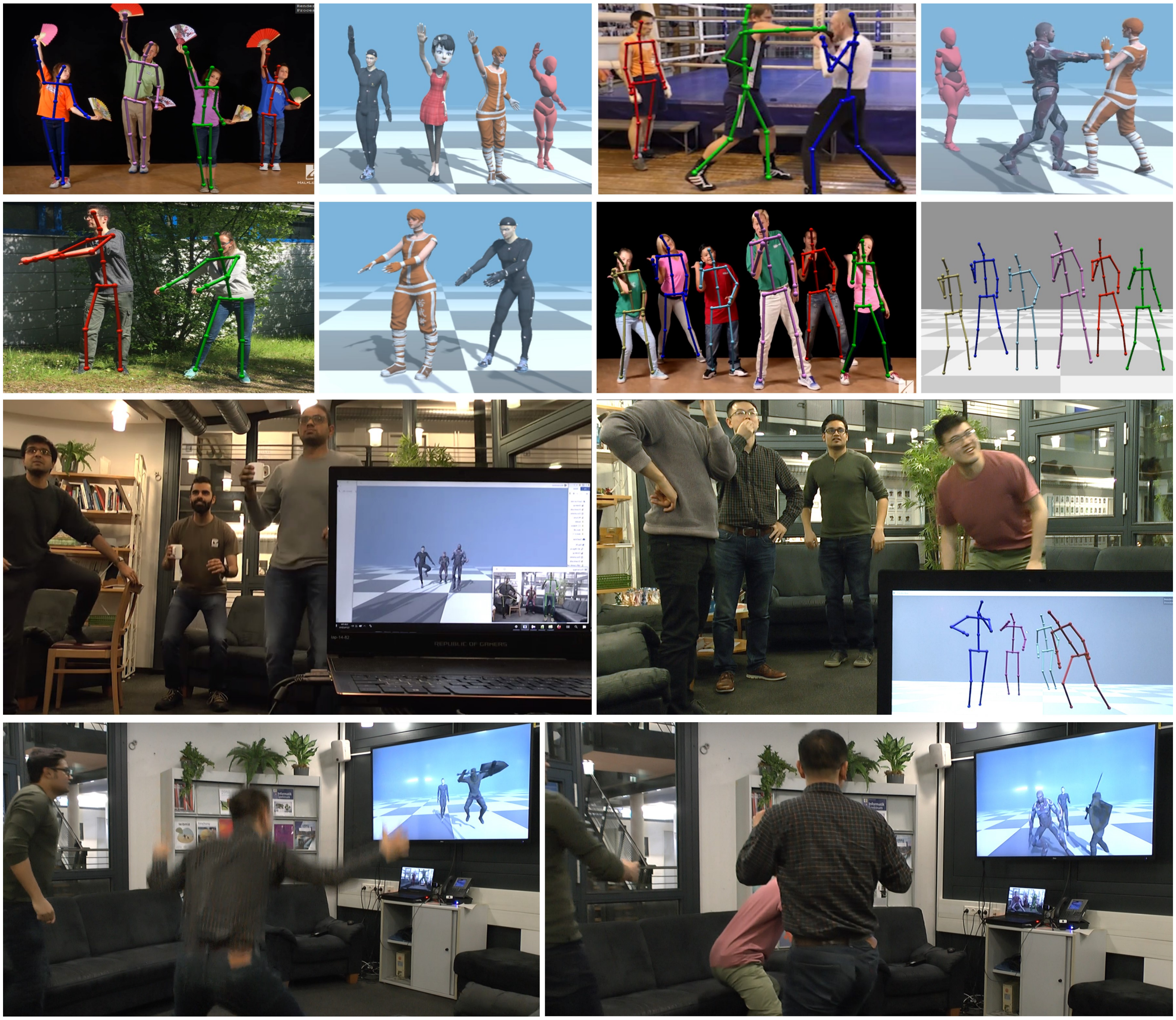


Fig. 4. **Live Interaction and Virtual Character Control:** The temporally smooth joint angle predictions from *Stage III* can be readily employed for driving virtual characters in real-time. The top shows our system driving virtual skeletons and characters in real-time with the captured motion. On the bottom, our system is set up as a Kinect-like game controller, allowing subjects to interact with their virtual avatars live. Some images courtesy Boxing School Alexei Frolov (https://youtu.be/dbuz9Q05bsM), and Music Express Magazine (https://youtu.be/kX6xMYlEwLA, https://youtu.be/lv-h4WNnw0g). Please refer to the supplemental video for more results.

as our approach is a learning based approach, it mispredicts when the presented pose is outside the training distribution.

Comparisons With Prior Work: Figure 5 shows visual comparisons of our approach to prior single-person and multi-person approaches. Also refer to the accompanying video for further comparisons. Results of our real-time system are comparable or better in quality than both, single-person and multi-person approaches, many of which run at offline [Kanazawa et al. 2018, 2019; Mehta et al. 2018; Moon et al. 2019] and interactive [Dabral et al. 2019; Rogez et al. 2019] frame-rates. As shown in the video, the temporal stability of our approach is comparable to real-time [Mehta et al. 2017b] and offline [Kanazawa et al. 2019] temporally consistent single-person approaches. Our approach differs from much of recent multi-person approaches in that ours is a bottom-up approach, while others employ a top-down formulation [Dabral et al. 2019; Moon et al. 2019; Rogez et al. 2017, 2019] inspired by work on object

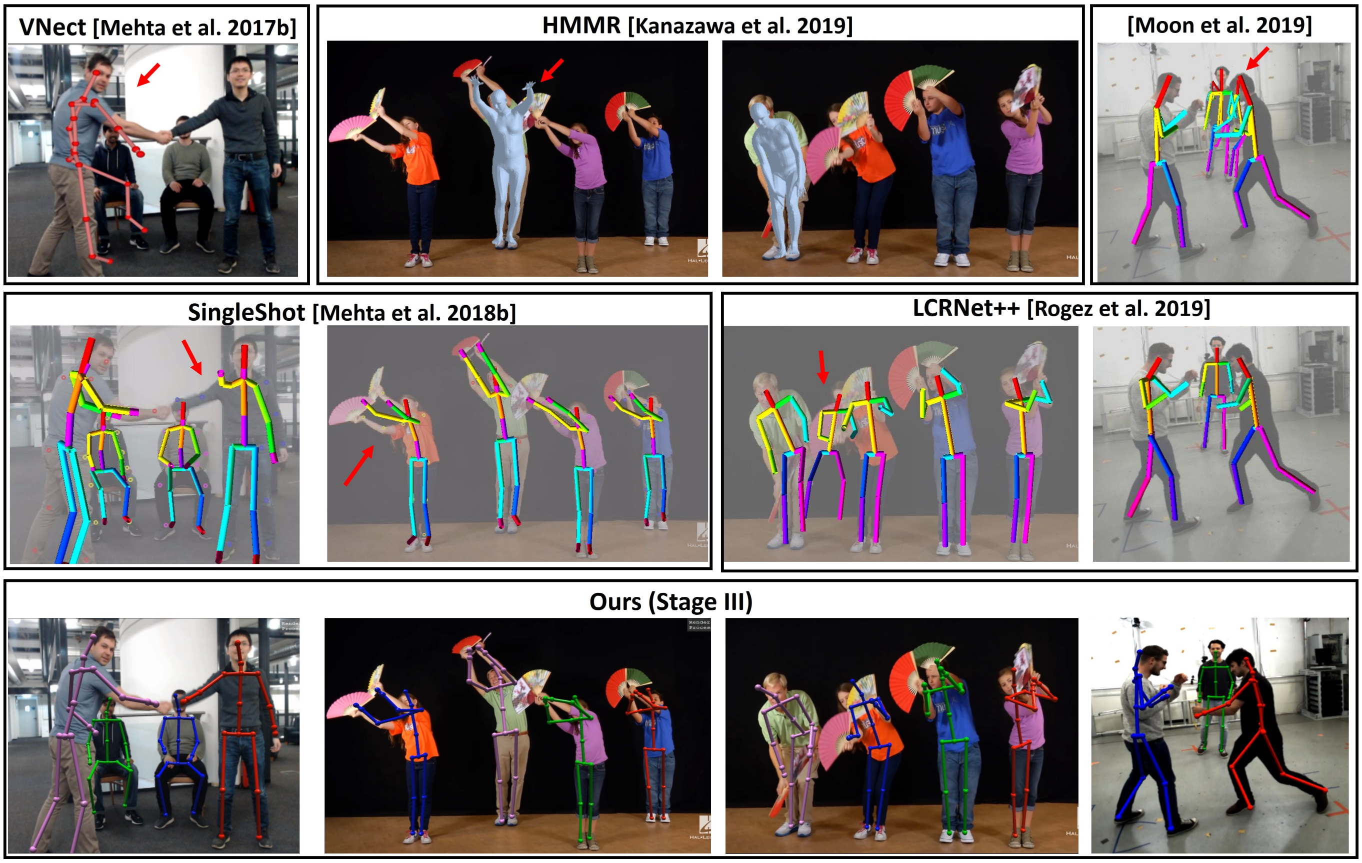

VNect [Mehta et al. 2017b]
HMMR [Kanazawa et al. 2019]
[Moon et al. 2019]
SingleShot [Mehta et al. 2018b]
LCRNet++ [Rogez et al. 2019]
Ours (Stage III)

Fig. 5. Real-time 3D Pose approaches such as VNect [2017b] only work in single person scenarios, and not designed to be occlusion robust or to handle other subjects in close proximity to the tracked subject. LCRNet++ [2019] is able to handle multi-person scenarios, but works at interactive frame rates, and requires post processing to be able to fuse the multiple pose proposals generated per subject. The post-processing is not always successful at fusing all proposals, leading to ghost predictions. The offline single-person approach HMMR [2019] uses 2D multi-person pose estimation as a pre-processing step and is thus able to handle unoccluded subjects in multi-person scenes in a top-down way. However, the approach fails under occlusion, and the run-time scales linearly with the number of subjects in the scene. The multi-person approach of [Mehta et al. 2018] jointly handles multiple subjects in the scene, however shows failures in cases of inter-personal proximity. The multi-person approach of [Moon et al. 2019] works offline, and similar to LCRNet++ it often produces spurious predictions due to the difficulty of filtering multiple proposals from top-down approaches. Here for our bottom-up approach (bottom), we show the 3D skeleton from *Stage III* reprojected on the image. Some images courtesy Music Express Magazine (https://youtu.be/kX6xMYlEwLA).

detection. Top-down approaches produce multiple predictions (proposals) per subject in the scene, and require a post-processing step to filter. Even when carefully tuned, this filtering step can either suppress valid predictions (two subjects with similar poses in close proximity) or fail to suppress invalid predictions (ghost predictions where there is no subject in the scene).

Live Interaction and Character Control: Figure 4 shows additional examples of live character control with our real-time monocular motion capture approach. Also refer to the accompanying video for more character control examples. Our system can act as a drop-in replacement for typical depth sensing based game controllers, allowing subjects to interact with their live avatars.

Diverse Pose and Scene Settings: Figure 6 shows the 3D capture results from our system (*Stage III*) overlaid on input images from diverse and challenging scenarios. See the accompanying video for additional results. Our approach can handle a wide range of poses, in a wide variety of scenes with different lighting conditions, background, and person density.

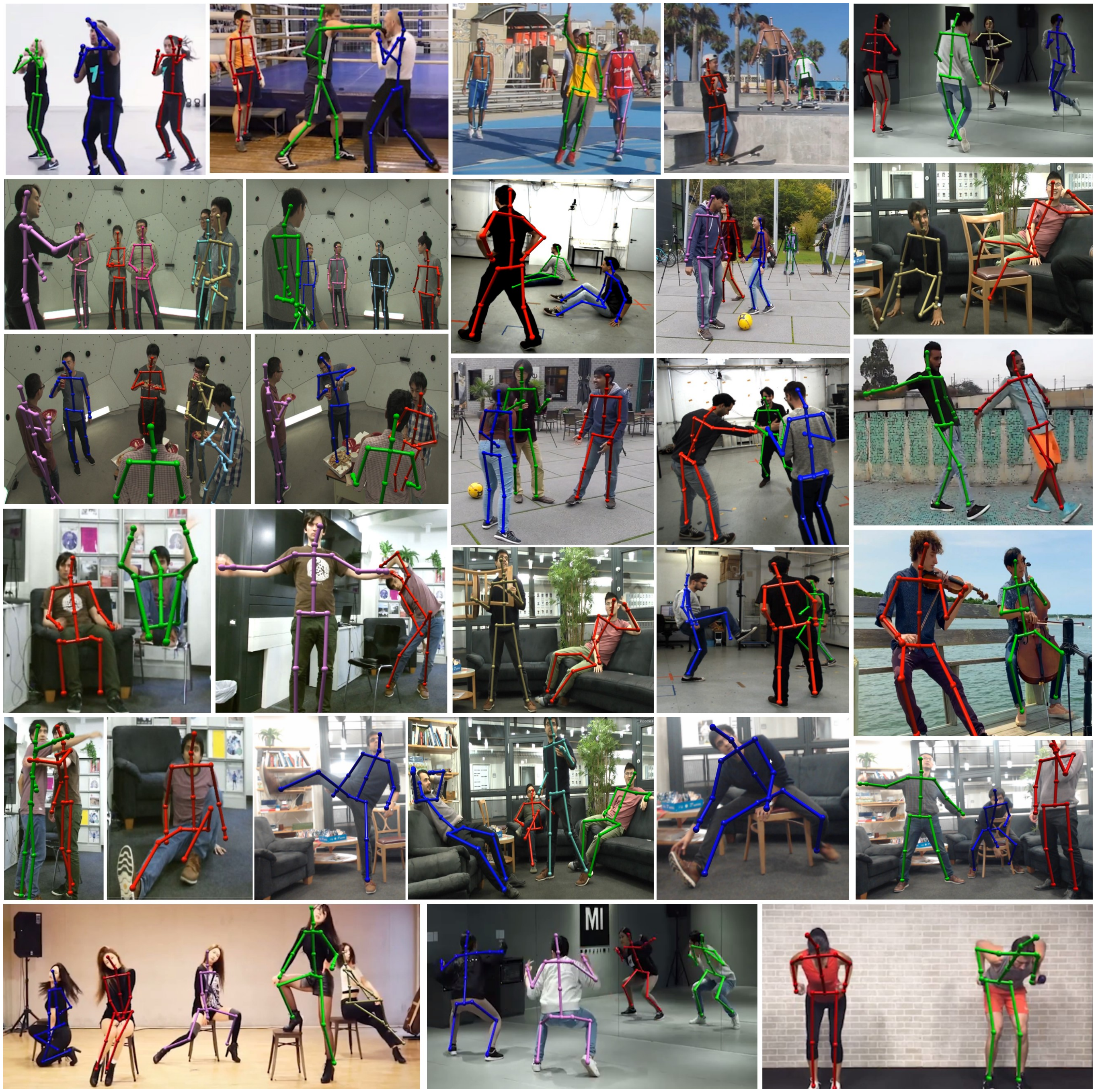


Fig. 6. Monocular 3D motion capture results from our full system (*Stage III*) on a wide variety of multi-person scenes, including Panoptic [2015] and MuPoTS-3D [2018] datasets. Our approach handles challenging motions and poses, including interactions and cases of self-occlusion. Some images courtesy KNG Music (https://youtu.be/_xCKmEhKQl4), 1MILLION TV (https://youtu.be/9HkVnFpmXAw), Indian dance world (https://youtu.be/PN6tRmj6xGU), 7 Minute Mornings (https://youtu.be/oVgG5ENXyVs), Crush Fitness (https://youtu.be/8qFwPKfllGI), Boxing School Alexei Frolov (https://youtu.be/dbuz9Q05bsM), and Brave Entertainment (https://youtu.be/ZhuDSdmby8k). Please refer to the accompanying video for more results.

REFERENCES

Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2014. 2D Human Pose Estimation: New Benchmark and State of the Art Analysis. In *CVPR*.

Zhe Cao, Gines Hidalgo Martinez, Tomas Simon, Shih-En Wei, and Yaser A Sheikh. 2019. OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2019).

Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. 2017. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. In *CVPR*.

François Chollet. 2017. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1251–1258.

Rishabh Dabral, Nitesh B Gundavarapu, Rahul Mitra, Abhishek Sharma, Ganesh Ramakrishnan, and Arjun Jain. 2019. Multi-Person 3D Human Pose Estimation from Monocular Images. In *2019 International Conference on 3D Vision (3DV)*. IEEE, 405–414.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *CVPR*.

Sam Johnson and Mark Everingham. 2010. Clustered Pose and Nonlinear Appearance Models for Human Pose Estimation. In *BMVC*. doi:10.5244/C.24.12.

Sam Johnson and Mark Everingham. 2011. Learning Effective Human Pose Estimation from Inaccurate Annotation. In *CVPR*.

Jolibrain Caffe Fork 2018. Caffe. https://github.com/jolibrain/caffe.

Hanbyul Joo, Hao Liu, Lei Tan, Lin Gui, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh. 2015. Panoptic Studio: A Massively Multiview System for Social Motion Capture. In *ICCV*. 3334–3342.

Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. 2018. End-to-end Recovery of Human Shape and Pose. In *CVPR*.

Angjoo Kanazawa, Jason Y. Zhang, Panna Felsen, and Jitendra Malik. 2019. Learning 3D Human Dynamics from Video. In *Computer Vision and Pattern Recognition (CVPR)*.

Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. 2017a. Monocular 3D Human Pose Estimation In The Wild Using Improved CNN Supervision. In *3DV*. IEEE. https://doi.org/10.1109/3dv.2017.00064

Dushyant Mehta, Oleksandr Sotnychenko, Franziska Mueller, Weipeng Xu, Srinath Sridhar, Gerard Pons-Moll, and Christian Theobalt. 2018. Single-Shot Multi-Person 3D Pose Estimation From Monocular RGB. In *3DV*. IEEE. http://gvv.mpi-inf.mpg.de/projects/SingleShotMultiPerson

Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. 2017b. VNect: Real-time 3D Human Pose Estimation with a Single RGB Camera. In *TOG*, Vol. 36. 14. https://doi.org/10.1145/3072959.3073596

Gyeongsik Moon, Juyong Chang, and Kyoung Mu Lee. 2019. Camera Distance-aware Top-down Approach for 3D Multi-person Pose Estimation from a Single RGB Image. In *The IEEE Conference on International Conference on Computer Vision (ICCV)*.

Gregory Rogez, Philippe Weinzaepfel, and Cordelia Schmid. 2017. LCR-Net: Localization-Classification-Regression for Human Pose. In *CVPR*.

Grégory Rogez, Philippe Weinzaepfel, and Cordelia Schmid. 2019. LCR-Net++: Multi-person 2D and 3D Pose Detection in Natural Images. *PAMI* (2019).

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015. ImageNet Large Scale Visual Recognition Challenge. *IJCV* 115, 3 (2015), 211–252. https://doi.org/10.1007/s11263-015-0816-y

Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. 2018. Mobilenetv2: Inverted residuals and linear bottlenecks. In *CVPR*. IEEE, 4510–4520.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2818–2826.

Fengting Yang and Zihan Zhou. 2018. Recovering 3D Planes from a Single Image via Convolutional Neural Networks. In *ECCV*. 85–100.

Andrei Zanfir, Elisabeta Marinoiu, and Cristian Sminchisescu. 2018. Monocular 3D Pose and Shape Estimation of Multiple People in Natural Scenes–The Importance of Multiple Scene Constraints. In *CVPR*. 2148–2157.