Evaluating Recent 2D Human Pose Estimators for 2D-3D Human Body Pose Lifting

Soroush Mehraban, and Yiqian Qin

Abstract—Monocular 3D human pose estimation is mainly about predicting 3D pixel coordinates of key body joints based on a 2D image or video. Typically, the first step for 3D human pose estimation is to estimate 2D positions of human body key joints using an off-the-shelf 2D human pose estimation model which is often times proposed a few years ago. In this paper, we evaluate the performance of recently proposed 2D human pose estimation models on 2D-3D human pose lifting task. In addition, we propose three merging strategies to combine the outputs of these 2D human pose estimators, and generate less noisy 2D inputs for 3D human pose estimator, thus improve the 2D-3D human pose lifting performance. To conduct the evaluation, we use a popular benchmark dataset Human3.6M. Among the four recent 2D human pose estimators, ViTPose generates the most precise 2D estimations for the majority of the keypoints. In addition, it surpasses other recent 2D human pose estimators in terms of mean per-joint position error of estimated 3D sequences, resulting in better 2D-3D human pose lifting performance. For the three proposed merging strategies, they are all proved to be effective in reducing the mean per-joint position error of estimated 3D sequences. Notably, manual merging performs the best among the three strategies proposed, and achieves a 1.23% reduction in mean per-joint position error compared to ViTPose. Code is available at https://github.com/SoroushMehraban/2DEstimatorEvaluation/tree/master

Index Terms—Computer Vision, Human Pose Estimation

1 Introduction

Monocular 3D human pose estimation is a fundamental and critical computer vision task that mainly entails predicting 3D pixel coordinates of key body joints (e.g., knees, hips, elbows) based on a 2D image or video. It is being broadly used in a variety of applications (as illustrated in Figure 1), ranging from augmented [1] and virtual reality [2] to autonomous vehicles [3], and from clinical monitoring [4] to human-computer interaction [5].

Currently, 3D human pose estimation still remains an ill-posed problem due to depth ambiguities in 2D input data. A general approach for estimating 3D human pose involves two steps: (i) estimate 2D positions of human body key joints from frames of the video using an off-the-shelf 2D human pose estimation models, (ii) pass the resulting 2D key joints estimations to the 3D human pose estimation model as input to estimate their corresponding 3D key joints positions. However, various 2D human pose estimation models have been proposed lately, and their potential to serve as less noisy 2D input for 3D human pose estimation models remains unexplored.

In summary, the main contributions of our paper are:

- We evaluate the performance of recently proposed 2D human pose estimation models on 2D-3D human pose lifting task. Specifically, the recent 2D human pose estimators evaluated are TransPose [6], MogaNet [7], ViTPose [8], and PCT [9].

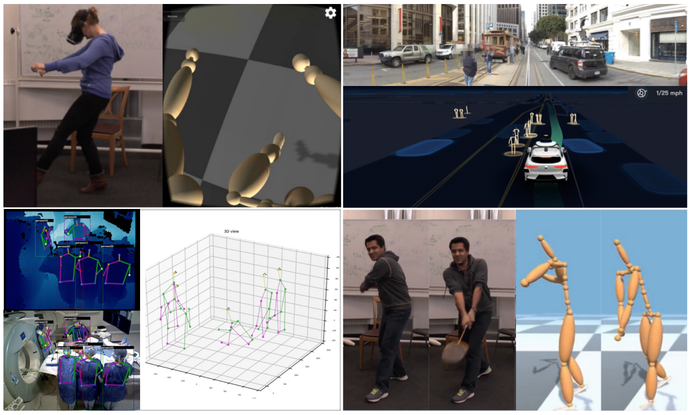

Fig. 1. Examples of 3D human pose estimation applications (images from left to right, top to bottom): Virtual Reality [2], Autonomous Vehicles [10], Clinical Monitoring [11] and Human-Computer Interaction [2].

- We propose three merging strategies which combine the outputs (2D key joints positions) of aforementioned 2D human pose estimation models, and generate less noisy 2D inputs for 3D human pose estimation model, thus improve the 2D-3D human pose lifting performance.

2 Related Work

2D human pose estimation. These models receive a single RGB image as input and output locations of main joints in 2D pixel coordinate. Cascaded Pyramid Network (CPN) [12] introduces GlobalNet, a feature pyramid network aimed at localizing keypoints that are easily detectable, such as eyes and hands. Furthermore, the CPN incorporates an

- *Soroush Mehraban is with the Department of Biomedical Engineering, University of Toronto, Toronto, ON, Canada, M5S 1A1. E-mail: soroush.mehraban@mail.utoronto.ca*
- *Yiqian Qin is with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada, M5S 1A1. E-mail: yiqian.qin@mail.utoronto.ca*

additional module called RefineNet, specifically devised to handle the localization of occluded keypoints. Stacked Hourglass [13] employs several stacked hourglass modules, enabling iterative bottom-up and top-down inference processes. TransPose [6] uses a CNN backbone to extract some high-level image features and then uses a transformer encoder to process these extracted features. MogaNet [7] proposes a new family of pure ConvNet structure which shows competitive results in various computer vision tasks, including object detection, semantic segmentation, and 2D human pose estimation. ViTPose [8] utilizes only a pure vision transformer for extracting image features and by using two deconvolution layers as the decoder, it generates heatmaps containing the 2D keypoints of different areas of the body. PCT [9] proposes a structured representation to explore the joint dependency. This way, they prevent the model output to generate unrealistic pose estimates.

Monocular 3D human pose estimation. Originally, this objective involved determining the 3D coordinates of joints directly from video frames, without the need for any intermediary processes [14], [15], [16], [17]. Inspired by the rapid development and availability of accurate 2D pose estimation models, these models currently receive a sequence of 2D human pose as input and lift them to 3D coordinate system. VideoPose3D [18] uses dilated temporal convolutions over 2D keypoints to infer the 3D pose sequence. PoseFormer [19] is the first method that proposes spatial transformers to extract intra-frame information between joints and temporal transformers to extract inter-frame information. PoseFormerV2 [20] enhances its computational efficiency by utilizing a frequency-domain representation, which also conferred robustness against abrupt movements in noisy data. STCFormer [21] proposes two parallel branches, one using spatial transformers and other using temporal transformers. P-STMO [22] introduces masked pose modeling and achieves a lower final error through self-supervised pretraining. Enfalt *et al.* [23] reduce computational complexity by utilizing masked token modeling. In StridedFormer [24] the traditional fully-connected layers in the feed-forward network of the transformer encoder are substituted with strided convolutions. This modification aims to gradually reduce the sequence length and effectively enhance the central frame. MotionBERT [25] further improves the performance by using spatial-temporal stack of transformers in one branch and temporal-spatial transformers in another branch. MotionAGFormer [26] uses spatial-temporal transformers in one branch but leverages Graph Convolutional Networks (GCNs) in another branch to capture a complementary information and output more accurate results.

Among all the 3D human pose estimation models, although MotionBERT and MotionAGFormer achieve the best final performance, they use Stacked Hourglass for 2D pose estimation. Since there is some preprocessing on their 2D input data that is unknown, for a fair comparison, we select the 2D-3D lifting model used by others. They all use 2D keypoints from VideoPose3D's paper, estimated by CPN, and the preprocessing steps are publicly available. Among these models, we select PoseFormerV2 due to its accelerated training capabilities, achieved by handling one-third of the sequence in the time domain and converting the remaining portion into the frequency domain through Discrete Cosine Transform (DCT). Subsequently, only the low-frequency coefficients are utilized for subsequent processing.

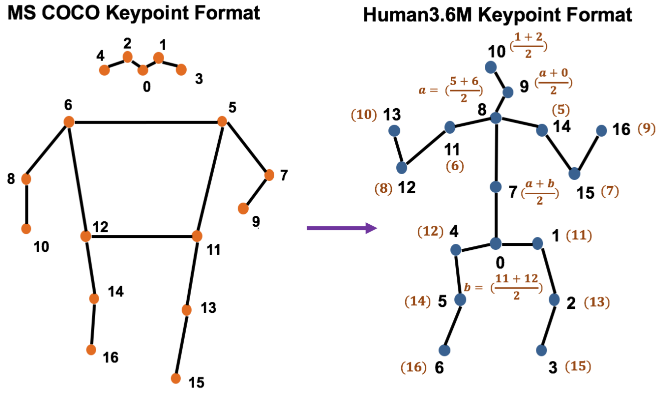

Fig. 2. MS COCO and Human3.6M keypoints format. For models trained on MS COCO dataset, we manually convert them to Human3.6M denoted by brown color.

3 Proposed Method

Our method involves using recent 2D estimation models, trained on the MS COCO Keypoint dataset [27], to estimate 2D keypoints on the Human3.6M dataset [28]. Following that, we use the estimated 2D pose sequences as input for the PoseFormerV2 model and train the model to infer the underlying 3D structure of the human body. Finally, we propose multiple merging strategies to combine different estimated 2D sequences and further improve the final performance.

3.1 2D Human Pose Estimation

State-of-the-art models such as ViTPose, PCT, MogaNet, and TransPose, trained on the MS COCO dataset, are used to estimate 2D pose sequences for Human3.6M dataset. However, the 2D pose output format differed from that of the Human3.6M dataset. To align them, we manually converted the formats as illustrated in Figure 2. Additionally, we incorporated 2D pose sequences from the VideoPose3D paper, including CPN fine-tuned on Human3.6M and Detectron with and without fine-tuning. While the fine-tuned sequences were already in Human3.6M format, we manually converted the Detectron sequences without fine-tuning to match the required format.

3.2 Merging Strategy

Three different merging strategies are proposed to improve the final 2D-3D lifting performance by introducing less noisy 2D data.

Manual merging. For this merging strategy, each human body keypoint of the different 2D estimators is compared with a ground truth, and for a single keypoint estimated with different 2D estimators, the one that has the least distance with the ground truth among all the training frames in Human3.6M is selected. For the ground truth, we project the motion capture 3D coordinates into 2D pixels by leveraging the camera intrinsic and extrinsic parameters. Specifically,

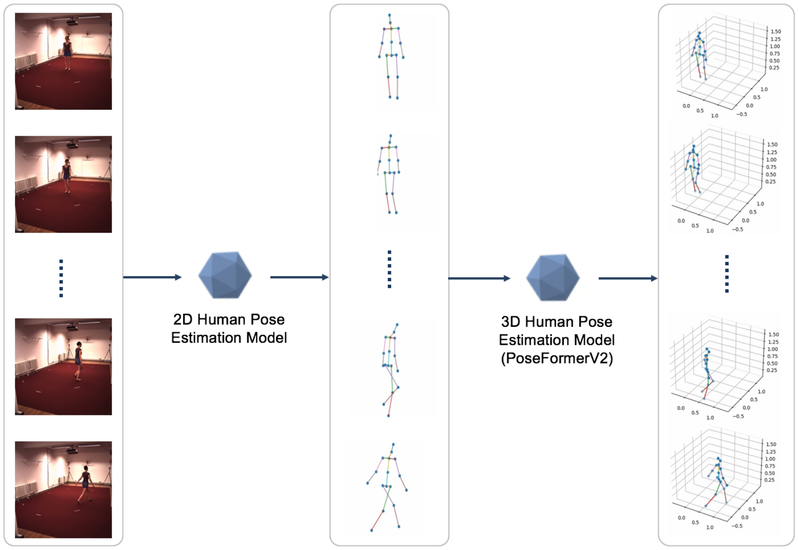
2D Human Pose Estimation Model
3D Human Pose Estimation Model (PoseFormerV2)

Fig. 3. **3D human pose estimation pipeline.** Initially, a 2D pose sequence is derived from RGB video through the use of a 2D pose estimator. Subsequently, PoseFormerV2 is trained to perform the task of lifting 2D poses to 3D, and the MPJPE is assessed as a measure of performance.

given 3D coordinates P_W in world coordinate system, we use

$$P_C = R(P_W - T) \tag{1}$$

to convert it to camera coordinates system $P_C = (X_c, Y_c, Z_c)$ where R and T are rotation and translation parameters, respectively. Next, It is projected to 2D coordinates using

$$u = f\frac{X_c}{Z_c} + c_x, \tag{2}$$

$$v = f\frac{Y_c}{Z_c} + c_y, \tag{3}$$

where $P_p = (u, v)$ is 2D coordinates in pixel coordinates system. The intrinsic parameters, f, c_x, c_y, denote focal length and image center, respectively. Finally, for selecting the estimator d for a joint j, the 2D coordinates $P^t_{p',j}$ is represented as

$$\begin{aligned} &P^t_{p',j} = P^t_{d,j}, \\ &\text{where } d = \arg\min_{1 \leq i \leq D} \sum_{t=1}^{T} ||P^t_{p,j} - P^t_{i,j}||. \end{aligned} \tag{4}$$

In formula above, $D = 4$ is the number of 2D estimators and T is the total number of frames in Human3.6M used for training.

Average merging. In this merging approach, we compute the average of ViTPose, PCT, and MogaNet for each individual frame within the sequence. This averaging process aims to mitigate the impact of noise in the 2D input. Given that each estimator introduces varying levels of noise for a specific frame, combining their outputs through averaging is anticipated to yield a less noisy 2D input. This strategy leverages the diversity in noise patterns among the estimators, providing a more balanced and refined outcome across frames. Note that TransPose is not used for averaging because in general it is having more noise compared to the rest (see experimental results for more details).

Weighted average merging. In this methodology, We follow a similar path as before but introduce a refinement by incorporating the confidence scores of each estimation as weights in a weighted average. Notably, PCT provides confidence scores in the form of logits rather than conventional probability scores. To align these scores between 0 and 1, we normalize the confidence scores of PCT by dividing them by the maximum value within the sequence. Subsequently, we normalize the confidence scores across various estimators and employ them as weights in the weighted average. This strategy allows us to account for the confidence levels associated with each estimator's output, offering a more informed combination of results.

3.3 2D-3D Lifting

Following estimation of 2D pose sequences using different estimators and the proposed merging strategies, PoseFormerV2 [20] is trained for the task of 2D-3D lifting (Figure 3). PoseFormerV2 takes a sequence of $T = 27$ frames as input. To enhance computational efficiency, the central $T' = 3$ frames are utilized in a spatial transformer to capture intra-frame relationships among various body joints. To effectively capture long-range human dynamics in the original sequence, all $T = 27$ frames are transformed into Discrete Cosine Transform (DCT) coefficients, and a low-pass filter retains $N = 3$ coefficients for each joint trajectory. Subsequently, the output tokens from both the spatial transformer and the low-pass filter are combined and fed into a temporal transformer. Within this transformer, an attention module processes the tokens, with those associated with the frequency domain directed to a Multi-Layer Perceptron (MLP), while tokens related to the time domain undergo conversion into the frequency domain through DCT. After passing through the MLP, they are reverted to the time domain using Inverse Discrete Cosine Transform (IDCT). Finally, a regression head module is employed to estimate the 3D pose at the central frame. For both training and evaluation, Mean Per Joint Position Error (MPJPE) is used. It is defined as

$$L_{3D} = \Sigma_{t=1}^{T}\Sigma_{j=1}^{J}\|\hat{\mathbf{P}}_{t,j} - \mathbf{P}_{t,j}\|, \tag{5}$$

where J is number of joints, T is number of frames in batch of data, and $\hat{P}$ and P are the ground-truth 3D motion capture and estimated 3D pose, respectively.

4 Experimental Results

In this section, we'll delve into the quantitative comparison of the 2D outputs generated by each 2D estimator. We'll systematically assess the final results produced by the 2D-3D lifting model when trained on varied 2D datasets. Subsequently, we will qualitatively compare the ultimate 3D output.

4.1 Quantitative Comparison between 2D sequences

The 2D sequences generated by different 2D estimators are initially transformed into the Human3.6M format, as illustrated in Figure 2. These converted sequences are then compared with the 2D ground truth, calculated through the 3D-2D camera projection process outlined in Equations 2 to 3. For the comparison, we incorporate all the training data from subjects 1, 5, 6, 7, and 8 in Human3.6M. Subsequently, we calculate the average for each joint by considering all frames across all the videos. The comparison for a subset of joints is illustrated in Figure 4. ViTPose generally surpasses other estimators in terms of mean per-joint position error, leading to more precise keypoint outputs. Nevertheless, for certain keypoints, such as the Left Knee, PCT tends to yield more accurate keypoints on average compared to ViTPose. We hypothesize that the disparities in errors across various body regions can be attributed to the distinct biases inherent in each model, stemming from the use of different architectures. Building upon this concept, during manual merging, ViTPose is predominantly employed. However, for keypoints where PCT exhibits lower average errors on training data, ViTPose's estimations are substituted to achieve less noisy 2D data.

4.2 Quantitative comparison between 3D sequences

Table 1 compares the estimated 3D sequences with the motion capture 3D ground truth on the Human3.6M dataset after training PoseFormerV2 using different 2D estimations as input. By comparison, ViTPose, which demonstrates state-of-the-art performance on the MS COCO keypoints dataset, attains the lowest mean per-joint position error among the four recent models assessed for this task. Nevertheless, the ultimate performance is 2.96 mm lower when compared to the scenario where PoseFormerV2 is trained with the CPN model. It's important to note that CPN undergoes fine-tuning on the Human3.6M dataset. We consider this approach unfair since the training and testing data in the Human3.6M dataset share identical environments and cameras, with subjects positioned at nearly the same distances. Consequently, CPN may acquire biases from the Human3.6M dataset that may not be applicable to real-world scenarios where the subject is situated in a completely different environment. Through the utilization of the 2D sequences obtained via the merging strategies, we can enhance the performance of PoseFormerV2. Among the various merging strategies, manual merging proves to be the most effective, resulting in a 1.23% reduction in error compared to ViTPose.

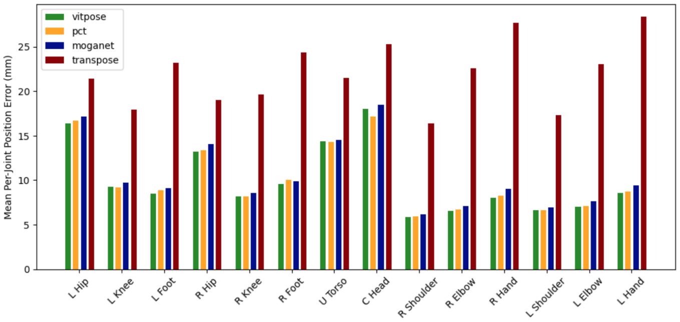

Fig. 4. The mean per-joint position error between each tested 2D estimator and the 2D ground truth. 'L' denotes left and 'R' denotes right.

TABLE 1
The mean per-joint position error (mm) comparisons of estimated 3D keypoints on Human3.6M after training the PoseFormerV2 model using different 2D estimations.

2D Estimator	finetuned	MPJPE (mm)↓
Detectron [2]	×	59.56
Detectron [2]	✓	55.91
CPN [3]	✓	49.65
MogaNet [4]	×	54.77
TransPose [5]	×	66.20
PCT [6]	×	53.26
ViTPose [7]	×	52.61
Merge (Manual)	×	51.96
Merge (Average)	×	52.53
Merge (Weighted Average)	×	52.50

4.3 Qualitative comparison between 3D sequences

Figure 5 visualizes the difference between sample estimated 3D sequences and the motion capture 3D ground truth on the Human3.6M dataset after training PoseFormerV2 using different 2D estimations as input. Overall, the PoseFormerV2 trained by using 2D estimations from ViTPose exhibits the best alignment with the ground truth (e.g., more precise feet and hands estimations), compared to other three recent 2D pose estimators evaluated for 2D-3D human pose lifting task. Nevertheless, similar to the quantitative result, it displays a slightly worse alignment with the ground truth when compared to the scenario where PoseFormerV2 is trained with the 2D estimations from CPN model. In addition, the PoseFormerV2 trained by using 2D estimations from manual merging displays a slightly superior alignment with the ground truth (e.g., slightly more precise hands estimations), when compared to other two merging strategies.

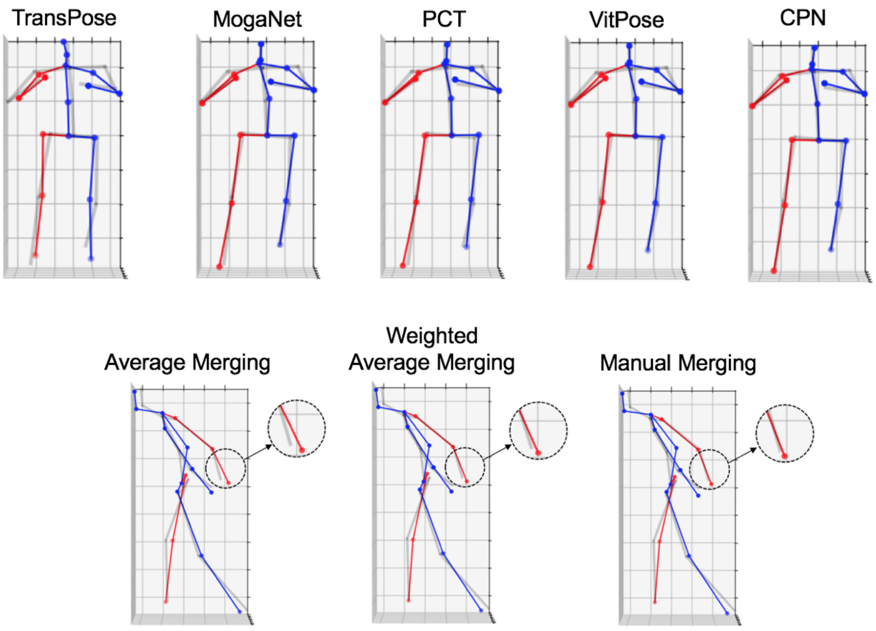

Fig. 5. Qualitative comparisons of estimated 3D keypoints on Human3.6M after training the PoseFormerV2 model using different 2D estimations. The transparent gray skeleton is the ground-truth 3D pose. The right part of the estimated body is denoted by red color, the torso and left part of the estimated body are denoted by blue color.

5 Conclusion

Among the four recent 2D human pose estimators utilized in the 2D-3D pose lifting process, ViTPose exhibited the most promising results. Specifically, it generates the most precise 2D estimations for the majority of the keypoints, and achieves the lowest mean per-joint position error of estimated 3D sequences. Additionally, we introduced three merging strategies to combine the outputs of the 2D estimators, and all proved effective in reducing the final error in the estimated 3D sequences. Notably, manual merging emerged as the most successful among the proposed strategies, resulting in a 1.23% reduction in error compared to ViTPose.

Acknowledgments

We would like to express our appreciation to Parsa Mirdehghan for offering valuable feedback on our project. Our thanks also go to Dr. David Lindell and all the teaching assistants for their contributions to assembling this exceptional course.

References

[1] H.-Y. Lin and T.-W. Chen, "Augmented reality with human body interaction based on monocular 3d pose estimation," in *International Conference on Advanced Concepts for Intelligent Vision Systems*. Springer, 2010, pp. 321–331.

[2] D. Mehta, S. Sridhar, O. Sotnychenko, H. Rhodin, M. Shafiei, H.-P. Seidel, W. Xu, D. Casas, and C. Theobalt, "Vnect: Real-time 3d human pose estimation with a single rgb camera," *Acm transactions on graphics (tog)*, vol. 36, no. 4, pp. 1–14, 2017.

[3] P. Bauer, A. Bouazizi, U. Kressel, and F. B. Flohr, "Weakly supervised multi-modal 3d human body pose estimation for autonomous driving," in *2023 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2023, pp. 1–7.

[4] A. Bigalke, L. Hansen, J. Diesel, C. Hennigs, P. Rostalski, and M. P. Heinrich, "Anatomy-guided domain adaptation for 3d in-bed human pose estimation," *Medical Image Analysis*, vol. 89, p. 102887, 2023.

[5] T. L. Munea, Y. Z. Jembre, H. T. Weldegebriel, L. Chen, C. Huang, and C. Yang, "The progress of human pose estimation: A survey and taxonomy of models applied in 2d human pose estimation," *IEEE Access*, vol. 8, pp. 133 330–133 348, 2020.

[6] S. Yang, Z. Quan, M. Nie, and W. Yang, "Transpose: Keypoint localization via transformer," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 11 802–11 812.

[7] S. Li, Z. Wang, Z. Liu, C. Tan, H. Lin, D. Wu, Z. Chen, J. Zheng, and S. Z. Li, "Efficient multi-order gated aggregation network," *arXiv preprint arXiv:2211.03295*, 2022.

[8] Y. Xu, J. Zhang, Q. Zhang, and D. Tao, "Vitpose: Simple vision transformer baselines for human pose estimation," *Advances in Neural Information Processing Systems*, vol. 35, pp. 38 571–38 584, 2022.

[9] Z. Geng, C. Wang, Y. Wei, Z. Liu, H. Li, and H. Hu, "Human pose as compositional tokens," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 660–671.

[10] T. W. Team, "Utilizing key point and pose estimation for the task of autonomous driving," 2022. [Online]. Available: https://waymo.com/blog/2022/02/utilizing-key-point-and-pose-estimation.html

[11] V. Srivastav, T. Issenhuth, A. Kadkhodamohammadi, M. de Mathelin, A. Gangi, and N. Padoy, "Mvor: A multi-view rgb-d operating room dataset for 2d and 3d human pose estimation," *arXiv preprint arXiv:1808.08180*, 2018.

[12] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu, and J. Sun, "Cascaded pyramid network for multi-person pose estimation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7103–7112.

[13] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII 14*. Springer, 2016, pp. 483–499.

[14] G. Pavlakos, X. Zhou, and K. Daniilidis, "Ordinal depth supervision for 3d human pose estimation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7307–7316.

[15] G. Pavlakos, X. Zhou, K. G. Derpanis, and K. Daniilidis, "Coarse-to-fine volumetric prediction for single-image 3d human pose,"

in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7025–7034.

[16] X. Sun, B. Xiao, F. Wei, S. Liang, and Y. Wei, "Integral human pose regression," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 529–545.

[17] K. Zhou, X. Han, N. Jiang, K. Jia, and J. Lu, "HEMlets pose: Learning part-centric heatmap triplets for accurate 3d human pose estimation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 2344–2353.

[18] D. Pavllo, C. Feichtenhofer, D. Grangier, and M. Auli, "3d human pose estimation in video with temporal convolutions and semi-supervised training," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 7753–7762.

[19] C. Zheng, S. Zhu, M. Mendieta, T. Yang, C. Chen, and Z. Ding, "3d human pose estimation with spatial and temporal transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 11 656–11 665.

[20] Q. Zhao, C. Zheng, M. Liu, P. Wang, and C. Chen, "PoseFormerV2: Exploring frequency domain for efficient and robust 3d human pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023, pp. 8877–8886.

[21] Z. Tang, Z. Qiu, Y. Hao, R. Hong, and T. Yao, "3d human pose estimation with spatio-temporal criss-cross attention," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4790–4799.

[22] W. Shan, Z. Liu, X. Zhang, S. Wang, S. Ma, and W. Gao, "P-STMO: Pre-trained spatial temporal many-to-one model for 3d human pose estimation," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part V*. Springer, 2022, pp. 461–478.

[23] M. Einfalt, K. Ludwig, and R. Lienhart, "Uplift and upsample: Efficient 3d human pose estimation with uplifting transformers," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, January 2023.

[24] W. Li, H. Liu, R. Ding, M. Liu, P. Wang, and W. Yang, "Exploiting temporal contexts with strided transformer for 3d human pose estimation," *IEEE Transactions on Multimedia*, vol. 25, pp. 1282–1293, 2022.

[25] W. Zhu, X. Ma, Z. Liu, L. Liu, W. Wu, and Y. Wang, "Motionbert: Unified pretraining for human motion analysis," *arXiv preprint arXiv:2210.06551*, 2022.

[26] S. Mehraban, V. Adeli, and B. Taati, "Motionagformer: Enhancing 3d human pose estimation with a transformer-gcnformer network," *arXiv preprint arXiv:2310.16288*, 2023.

[27] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*. Springer, 2014, pp. 740–755.

[28] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6M: Large scale datasets and predictive methods for 3d human sensing in natural environments," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 7, pp. 1325–1339, 2013.