

# 3D Human Pose Perception from Egocentric Stereo Videos

## — Supplementary Material —

Hiroyasu Akada

Jian Wang

Vladislav Golyanik

Christian Theobalt

Max Planck Institute for Informatics, SIC

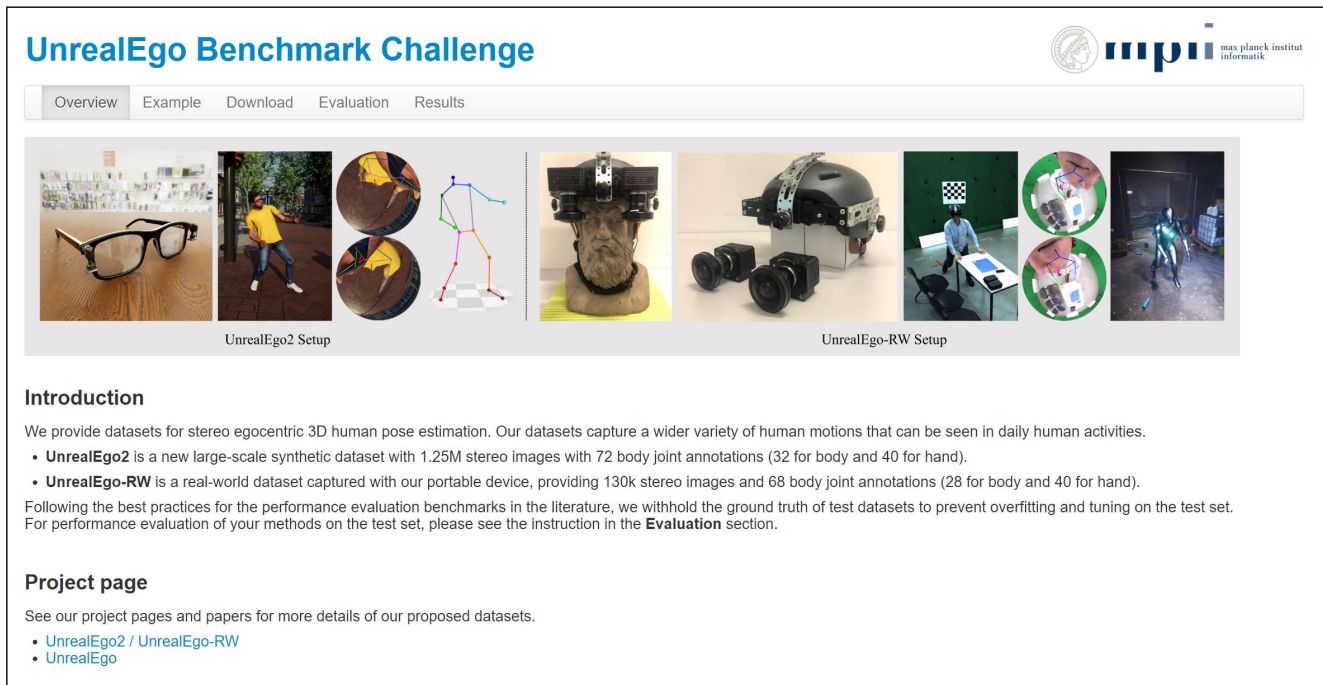


Figure 7. UnrealEgo Benchmark Challenge website.

This supplementary material provides more details about our work, including our benchmark challenge, proposed datasets, method, and additional results. Please watch the video on our project page for dynamic visualizations, including the proposed datasets, *i.e.*, UnrealEgo2 and UnrealEgo-RW.

## A. UnrealEgo Benchmark Challenge

We release the UnrealEgo Benchmark Challenge<sup>1</sup> webpage as shown in Fig. 7. Here, we provide the details of our proposed datasets, *i.e.*, UnrealEgo2 and UnrealEgo-RW, including example visualizations of our setup and data, data download procedure, evaluation protocol, and performance results of submitted works. Please see the webpage for more details.

<sup>1</sup><https://unrealego.mpi-inf.mpg.de/>

## B. Dataset Comparison

We provide a detailed comparison of existing datasets [1, 2, 4] for egocentric stereo 3D human pose estimation as shown in Table 8. As mentioned in Secs. 2 and 3 of the main paper, UnrealEgo2 adapts the publicly available eyeglasses-based setup [1], offering the largest synthetic dataset with 1.25M in-the-wild stereo fisheye views. Note that it does not share the same motions with UnrealEgo [1]. Therefore, it allows for a more comprehensive evaluation of existing and upcoming methods in various scenarios. UnrealEgo-RW is based on the head-mounted device equipped with two fisheye cameras. It provides the largest real-world dataset with 130k stereo fisheye views among the publicly available real-world datasets. Note that the EgoGlass dataset [4] is not publicly available. Unlike the existing real-world datasets [2, 4], UnrealEgo-RW offers a wider variety of mo-






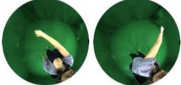
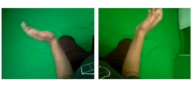



	EgoCap [2]	EgoGlass [4] (Not publicly available)	UnrealEgo [1]	UnrealEgo2 (Ours)	UnrealEgo-RW (Ours)
Device					
Example Data					
Distance to user's face	~25 cm	~1cm	~1cm	~1cm	1~2cm
Egocentric views	30k × 2 views	170k × 2 views	450k × 2 views	1.25m × 2 views	130k × 2 views
Keypoints	body: 17	body: 13	body: 32, hand: 40	body: 32, hand: 40	body: 16
Environment	studio	studio	synthetic 3D world	synthetic 3D world	studio
Motion diversity	low	low	high	high	high

Table 8. Comparison of existing datasets for egocentric stereo 3D human pose estimation.

tions, such as crawling and dancing, making itself a unique and challenging dataset for egocentric stereo 3D human pose estimation.

### C. Data Loading Protocol

We compare video-based methods (*i.e.*, our approach and the Baseline) with the existing methods [1, 4] operating on single frames. For a fair comparison, we pad the input video with the initial frame for the video-based methods as shown in Fig. 8 such that the 3D errors can be computed on the same sequences of the same lengths for both video-based and single-frame-based methods. Specifically, we pad the original video input (colored in red) with the first frame, *i.e.*, frame 0, at the beginning of the sequence, generating several sets of input sequences. Then, 3D poses corresponding to the last frames of the input sequences (colored in blue) are used to calculate the 3D errors, *i.e.*, MPJPE and PAMPJPE, for the video-based methods.

### D. Additional Details of Network Architecture

As mentioned in Sec. 4.4, we use two feature extractors for depth and heatmap data, *i.e.*,  $\mathcal{F}_{\text{Depth}}$  and  $\mathcal{F}_{\text{HM}}$ , respectively.  $\mathcal{F}_{\text{Depth}}$  consists of five convolutional layers with a kernel size of 4, a stride of 2, and a padding size of 1.  $\mathcal{F}_{\text{HM}}$  is composed of four convolutional layers with a kernel size of 4, a stride of 2, and a padding size of 1. Also, we use a transformer decoder that consists of six decoder layers, eight heads, a hidden dimension of  $\frac{C}{2}$ , and an MLP dimension of  $2C$ , where  $C = 512$ . The pose regression head after the transformer decoder consists of three linear layers with input dimension sizes  $\frac{C}{2}$ ,  $\frac{C}{4}$ , and  $\frac{C}{8}$ .

### E. Additional Implementation Details

We implement our method using PyTorch and train and test it on a single NVIDIA Quadro RTX 8000 GPU with 48GB of memory.  $\approx 33\%$  of the memory is allocated for

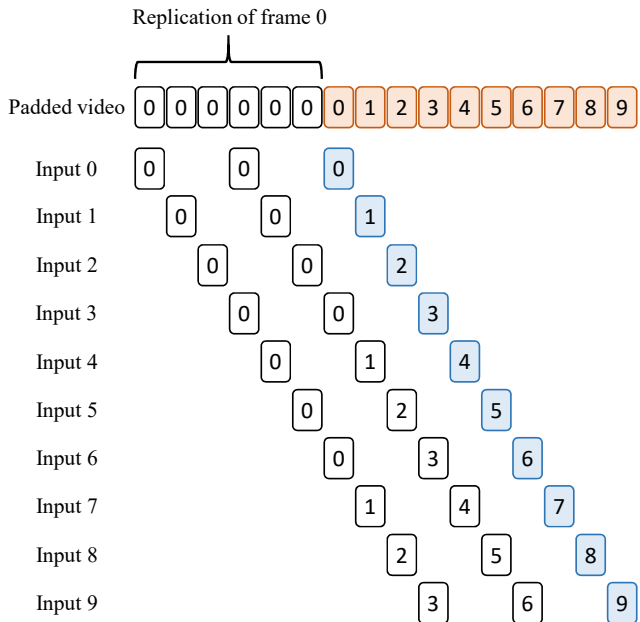


Figure 8. Schematic visualization of our video padding scheme for video-based methods. This example is for a video with 10 frames (red) and the input sequence length of 3, *i.e.*,  $T = 3$ , and a skip size of 3. The video is padded by replicating the first frame, *i.e.*, frame 0, to create 10 sets of input sequences. We use the 3D pose of the last frame (blue) for the calculation of 3D errors.

our model with a batch size of 4 and  $T=5$ . The training time for our model with  $T=5$  is approximately two days on UnrealEgo-RW, four days on UnrealEgo, and eight days on UnrealEgo2. The inference speed of our model with  $T = 5$  is 37.8 frames per second if not accounting for the 3D scene reconstruction with stereo SfM.

Methods	jumping	falling down	exercising	pulling,	singing	rolling	crawling	laying
Zhao <i>et al.</i> [4]	86.37	143.97	98.84	84.67	79.22	120.34	207.73	112.88
Akada <i>et al.</i> [1]	82.57	136.13	94.41	82.81	70.89	90.20	186.56	105.88
Baseline	<u>62.28</u>	<u>109.76</u>	<u>71.88</u>	<u>61.62</u>	<u>54.40</u>	<u>76.09</u>	<u>175.29</u>	<u>86.14</u>
Ours	<b>47.75</b>	<b>84.25</b>	<b>57.45</b>	<b>39.18</b>	<b>40.31</b>	<b>52.31</b>	<b>139.62</b>	<b>72.51</b>

---

Methods	sitting on the ground	crouching - normal	crouching - turning	crouching - to standing	crouching - forward	crouching - backward	crouching - sideways	standing - whole body
Zhao <i>et al.</i> [4]	218.99	130.78	125.36	80.44	76.18	98.01	102.09	77.31
Akada <i>et al.</i> [1]	235.04	125.91	142.52	83.43	85.58	88.81	92.56	71.56
Baseline	<u>172.89</u>	<u>95.20</u>	<u>125.79</u>	<u>63.35</u>	<u>58.71</u>	<u>63.55</u>	<u>72.01</u>	<u>53.54</u>
Ours	<b>109.49</b>	<b>67.12</b>	<b>95.43</b>	<b>44.99</b>	<b>38.55</b>	<b>47.40</b>	<b>48.17</b>	<b>38.56</b>

---

Methods	standing - upper body	standing - turning	standing - to crouching	standing - forward	standing - backward	standing - sideways	dancing	boxing
Zhao <i>et al.</i> [4]	75.62	79.19	80.22	80.34	77.16	91.50	89.84	76.23
Akada <i>et al.</i> [1]	70.95	76.79	98.95	75.46	73.99	77.94	85.35	74.09
Baseline	<u>51.36</u>	<u>59.91</u>	<u>52.53</u>	<u>57.87</u>	<u>56.66</u>	<u>62.88</u>	<u>63.51</u>	<u>52.70</u>
Ours	<b>36.66</b>	<b>45.03</b>	<b>33.74</b>	<b>42.96</b>	<b>38.05</b>	<b>45.04</b>	<b>47.06</b>	<b>37.34</b>

---

Methods	wrestling	soccer	baseball	basketball	american football	golf	average
Zhao <i>et al.</i> [4]	89.71	90.54	81.88	59.68	108.89	81.82	88.12
Akada <i>et al.</i> [1]	89.80	85.57	71.49	62.38	103.31	76.67	84.53
Baseline	<u>68.36</u>	<u>61.22</u>	<u>58.34</u>	<u>52.11</u>	<u>94.33</u>	<u>57.12</u>	<u>63.44</u>
Ours	<b>48.94</b>	<b>41.12</b>	<b>47.84</b>	<b>42.10</b>	<b>75.88</b>	<b>41.49</b>	<b>46.20</b>

Table 9. Quantitative evaluation of **device-relative** 3D pose estimation on the UnrealEgo [1] test split based on the 30 motion categories [1] (MPJPE with *mm*-scale), with training on UnrealEgo [1].

Methods	jumping	falling down	exercising	pulling,	singing	rolling	crawling	laying
Zhao <i>et al.</i> [4]	77.56	132.93	76.28	76.22	71.67	154.17	176.94	98.35
Akada <i>et al.</i> [1]	71.76	113.21	72.61	71.35	65.57	124.76	167.20	86.73
Baseline	<u>49.93</u>	<u>91.86</u>	<u>48.01</u>	<u>48.09</u>	<u>44.52</u>	<u>107.01</u>	<u>141.23</u>	<u>70.73</u>
Ours	<b>31.46</b>	<b>66.37</b>	<b>29.16</b>	<b>25.69</b>	<b>28.15</b>	<b>69.15</b>	<b>119.08</b>	<b>56.29</b>

---

Methods	sitting on the ground	crouching - normal	crouching - turning	crouching - to standing	crouching - forward	crouching - backward	crouching - sideways	standing - whole body
Zhao <i>et al.</i> [4]	197.32	104.21	84.19	81.33	82.07	102.84	98.11	81.29
Akada <i>et al.</i> [1]	155.48	94.83	73.55	77.62	74.47	96.44	87.52	74.36
Baseline	<u>118.10</u>	<u>71.00</u>	<u>54.45</u>	<u>57.12</u>	<u>53.80</u>	<u>74.18</u>	<u>67.15</u>	<u>52.77</u>
Ours	<b>61.89</b>	<b>41.20</b>	<b>28.36</b>	<b>31.81</b>	<b>32.73</b>	<b>37.46</b>	<b>36.35</b>	<b>30.54</b>

---

Methods	standing - upper body	standing - turning	standing - to crouching	standing - forward	standing - backward	standing - sideways	dancing	boxing
Zhao <i>et al.</i> [4]	67.85	77.03	70.49	75.27	75.27	73.25	86.13	70.45
Akada <i>et al.</i> [1]	63.76	70.02	67.97	68.17	66.25	67.35	76.35	63.22
Baseline	<u>43.01</u>	<u>51.62</u>	<u>47.24</u>	<u>48.98</u>	<u>48.78</u>	<u>46.52</u>	<u>55.92</u>	<u>45.44</u>
Ours	<b>21.28</b>	<b>35.02</b>	<b>21.47</b>	<b>29.88</b>	<b>27.64</b>	<b>24.22</b>	<b>35.93</b>	<b>24.07</b>

---

Methods	wrestling	soccer	baseball	basketball	american football	golf	average
Zhao <i>et al.</i> [4]	84.59	78.02	92.78	72.90	105.01	67.98	79.64
Akada <i>et al.</i> [1]	74.57	71.44	82.21	62.54	84.99	63.42	72.82
Baseline	<u>56.01</u>	<u>49.55</u>	<u>59.51</u>	<u>45.32</u>	<u>62.62</u>	<u>47.67</u>	<u>52.23</u>
Ours	<b>31.56</b>	<b>27.26</b>	<b>37.42</b>	<b>29.27</b>	<b>49.17</b>	<b>24.88</b>	<b>30.53</b>

Table 10. Quantitative evaluation of **device-relative** 3D pose estimation on the UnrealEgo2 test split based on the 30 motion categories [1] (MPJPE with *mm*-scale), with training on UnrealEgo2.



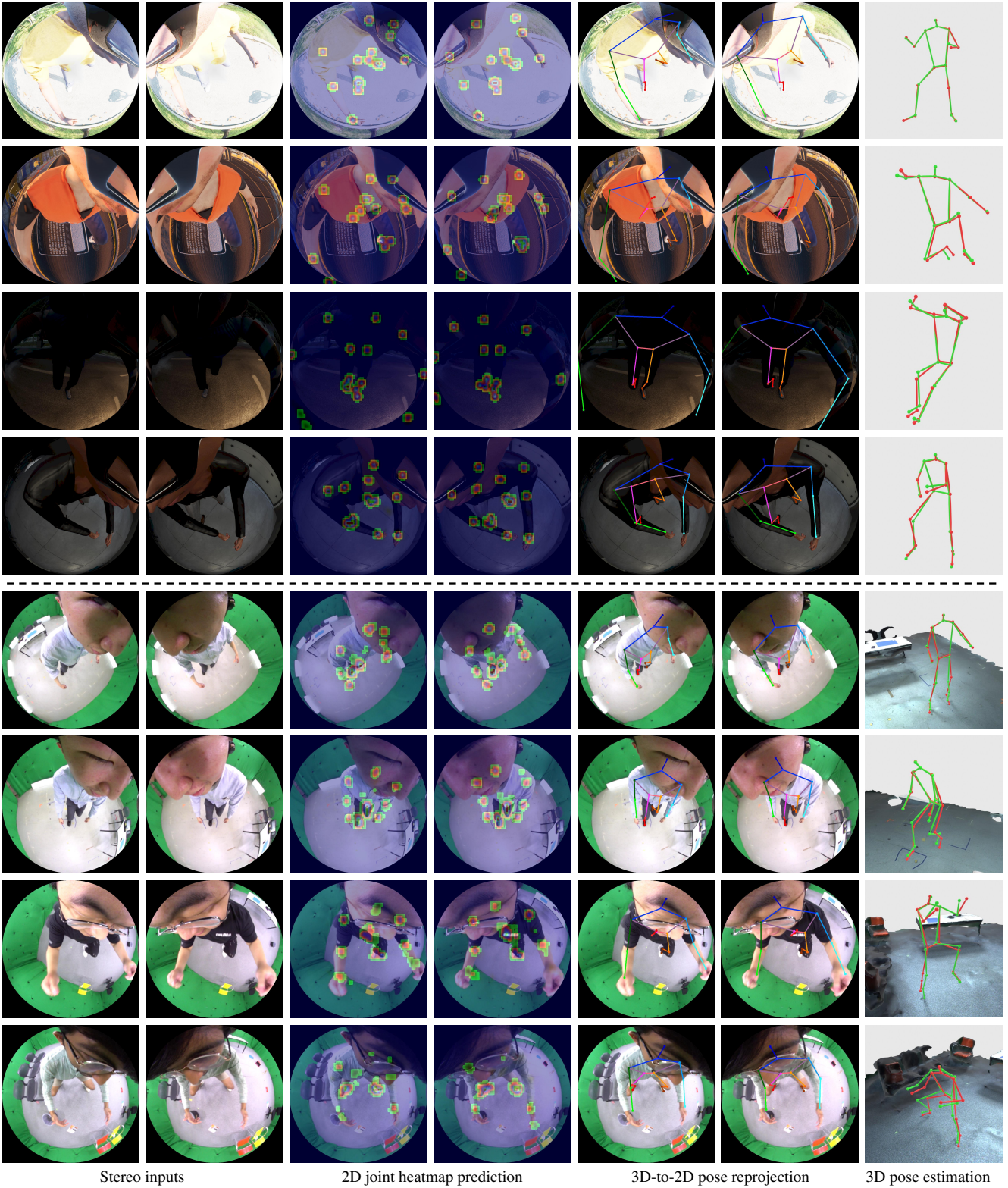


Figure 9. Visualization of outputs from our model on UnrealEgo2 (above) and UnrealEgo-RW (below). 3D pose estimation and ground truth are displayed in red and green, respectively. For UnrealEgo-RW, we show ground-truth scene meshes for visualization in the right column.

## F. Quantitative Results per Motion Category

Following the 30 motion categories proposed in UnrealEgo [1], we report the quantitative results for each motion type. Tables 9 and 10 summarise the metrics for device-relative 3D pose estimation on the UnrealEgo [1] and UnrealEgo2 test splits, respectively. The numbers demonstrate that our method outperforms the compared methods across all of the motion categories by a large margin. Another observation is that the 3D pose estimation for some complex motion types, such as crawling, shows relatively high 3D errors. This is mainly because of severe self-occlusions in all input frames in these scenarios. These results suggest possible future research directions in stereo egocentric 3D human pose estimation, such as adding motion priors to develop methods that are more robust to the occlusions.

## G. Additional Visualization

We provide additional qualitative results of our method in Fig. 9, including 2D heatmaps, 3D-to-2D pose reprojections and the estimated 3D poses. Please also watch our supplementary video for dynamic visualizations of the proposed datasets as well as more qualitative results.

## H. Limitations and Future Work

Our framework adopts SfM, which makes it difficult to achieve real-time inference. One solution could be to replace the SfM with a deep-learning approach as shown in [3] although this requires ground-truth depth maps of egocentric views for training. Furthermore, introducing additional exocentric cameras can effectively solve the self-occlusion of egocentric views. Thus, future work could focus on improving these aspects.

## References

- [1] Hiroyasu Akada, Jian Wang, Soshi Shimada, Masaki Takahashi, Christian Theobalt, and Vladislav Golyanik. UnrealEgo: A new dataset for robust egocentric 3d human motion capture. In *European Conference on Computer Vision (ECCV)*, 2022. 1, 2, 3, 5
- [2] Helge Rhodin, Christian Richardt, Dan Casas, Eldar Insafutdinov, Mohammad Shafiei, Hans-Peter Seidel, Bernt Schiele, and Christian Theobalt. Egocap: egocentric marker-less motion capture with two fisheye cameras. *ACM Transactions on Graphics (TOG)*, 35(6):1–11, 2016. 1
- [3] Jian Wang, Diogo Luvizon, Weipeng Xu, Lingjie Liu, Kripasindhu Sarkar, and Christian Theobalt. Scene-aware egocentric 3d human pose estimation. In *Computer Vision and Pattern Recognition (CVPR)*, 2023. 5
- [4] Dongxu Zhao, Zhen Wei, Jisan Mahmud, and Jan-Michael Frahm. Egoglass: Egocentric-view human pose estimation from an eyeglass frame. In *International Conference on 3D Vision (3DV)*, 2021. 1, 2, 3