Analyzing and Diagnosing Pose Estimation with Attributions Supplementary Material

Qiyuan He* Linlin Yang* Kerui Gu Qiuxia Lin Angela Yao
National University of Singapore

In the supplementary material, we present

- the influence of baseline images, likelihood approximation and common output space (see Subsec. 3.2) in Subsec. A.1 and A.2.
- the details of validation for PoseIG (see Subsec. 4.2) in Subsec. A.3 and A.4.
- the figure of the joint grouping (see Subsec. 4.1) in Subsec. B.1.
- more quantitative results on attribution versus MEPE and indices distribution on other models and joints (see Subsec. 4.3) in Subsec. B.2.
- the detailed statistics on easy versus hard cases (see Subsec. 4.3) in Subsec. C.1.
- the details of the impact of architecture and backbone (see Subsec. 4.4) in Subsec. C.2 and C.3.
- the details of the GCN refinement (see Subsec. 5.2) in Sec. D.
- the results on additional datasets (see Subsec. 4.1) in Sec. E.

Note that all the notations and abbreviations here are consistent with the main manuscript.

A. Discussion on PoseIG

A.1. Baseline Image

Different baseline images can significantly change the attribution of PoseIG. We mainly discuss the difference between solid color image,s *i.e.* white or black images, with linear paths and blurry images with blurry paths.

We adopt blurry images with the blurry path for PoseIG because PoseIG with a black/white baseline image is biased to a certain kind of color. To illustrate this, Fig. A provides an extreme case modified artificially. We find that attribution maps with solid color baseline images have two kinds of bias. Firstly, the contribution of pixels with the same color is overlooked. Secondly, the contribution of pixels with opposite colors is preferred. We consider these as severe biases. For instance, when a person wears black clothes or the environment is dark, it is hard for PoseIG with a black baseline image to keep fidelity. Similarly, when the background is too light, the white baseline image also falls into biased attribution. On the contrary, it is observed that silhouettes, image edges and textures are important for pose estimation [11]. Therefore, we choose to adopt blurry images instead of solid color images as the baseline image of PoseIG to avoid such bias.

Figure A. Attribution maps of two extreme cases to compare different baseline images. The first row shows the attribution map of the input where the human area is masked with black pixels, and the second row is masked with white pixels. (a) Input image with targets; attribution maps and attribution KDE heatmaps computed with (b) a white baseline image, (c) a black baseline image, and (d) a blurry image. With solid color baseline images, IG overlooks the contribution of the pixels with the same color and prefers those with a large different color.

A.2. Likelihood Approximation and Output Space

Common likelihood approximation and output space are utilized in PoseIG. Specifically, we use 2D joint location as output space for 2D human pose estimation and 3D joint location as output space for 3D hand pose estimation. They are necessary to ensure fairness in comparison and the axiom of **Implementation Invariance** defined in [20].

In an integrated gradient, we accumulate the gradients for a pixel of specific targets along a path as its contribution. To ensure fairness, the targets here should be semantically the same, and all the models should use the same information additional to themselves to obtain the targets. To ensure implementation invariance, two *functionally equivalent* models

*Equal contribution

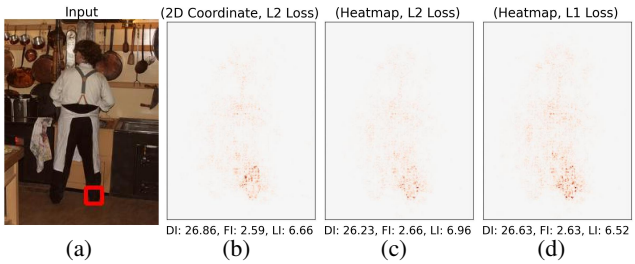


Figure B. Influence of output spaces and likelihood approximations on the same model Simple Baseline ResNet50 [22]. (a) Input with joint target location. (b)-(d) Attribution maps computed with (output space, factor in likelihood approximation) at the top of each image and (DI, FI, LI) at the bottom. Different output space and likelihood approximations lead to different attribution maps.

Mask	Nose	Shoulder	Elbow	Wrist	Hip	Knee	Ankle
M_m	1.68	1.62	1.68	1.57	1.68	1.76	1.70
M_{ro}	2.66	5.17	3.75	2.01	2.18	2.04	2.11
M_{ru}	2.54	3.38	2.66	2.93	2.74	3.04	2.95
M_{rn}	3.36	3.52	3.52	4.13	4.11	4.32	4.77

Table A. R_E of various joints perturbed with different masks. For each joint with each perturbation mask, R_E is significantly larger than 1, indicating that the change of EPE perturbed with the PoseIG attribution map is much larger than that perturbed with randomized maps.

Index	9 LAYERS	18 LAYERS	27 LAYERS	36 LAYERS	45 LAYERS	54 LAYERS
DI	13.58	13.93	16.53	17.44	19.91	21.06
FI	1.375	1.342	1.186	1.110	0.967	0.884

Table B. Quantitative results for model randomization test. Each column records the indices of attribution maps obtained by the model after randomizing a certain number of layers. When corrupting more layers successively, DI gets higher while FI gets lower.

should have the same attribution maps [20]. However, using different output spaces or likelihood approximations violates fairness and implementation invariance.

As shown in Fig. B, the same model with different modalities and likelihood approximation results in various attribution maps. Therefore, common likelihood approximation and output space are necessary. Since the joint coordinate can be obtained from other pose modalities without introducing additional factors, we use it as a common output space. In terms of likelihood approximation, we adopt the L2 Loss for PoseIG. Still, it is also acceptable to use other distances for likelihood approximation.

A.3. Image Perturbation

We follow the standard image perturbation test in [7, 8, 10, 15, 17]. The test is conducted based on the perturbed input. At first, we perturb the input image with the corresponding attribution map and feed it into the model to evaluate the performance. Secondly, we compare it with the performance

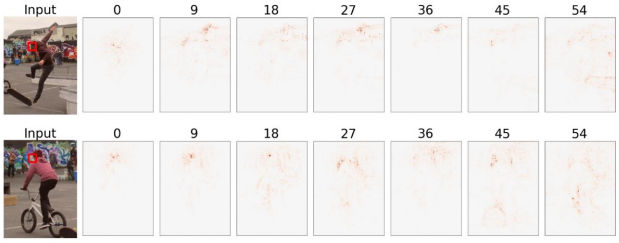


Figure C. Examples of the model randomization test. From left to right, the attribution maps are obtained by the models being successively more corrupted.

of the same image but perturbed with a randomized map. If pixel values of an attribution map indicate the importance for prediction, the performance perturbed with an attribution map should change significantly more than that with a randomized map.

Perturbation Definition. Formally, for an input image I with an attribution map G^m normalized by the maximum value, and a perturbation mask M, the perturbed image $\widetilde{I}$ is defined as:

$$\widetilde{I} = I \cdot (1 - G^m) + M \cdot G^m \tag{1}$$

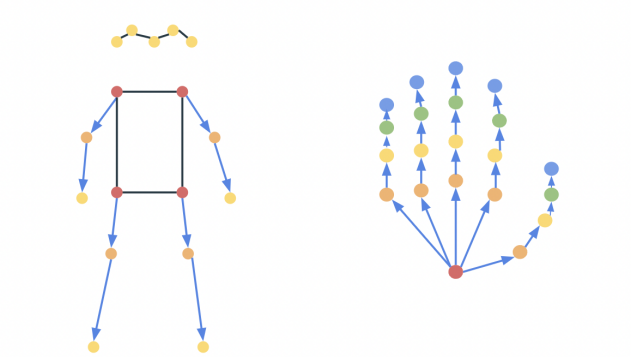
We generate the randomized map G^r, where each pixel is sampled from the original attribution map.

Perturbation Mask. We choose various perturbation masks M on the image perturbation test. The first kind of mask is images with a specific constant value. For instance, the mask encodes the mean value of the image M_m. The second kind of mask is the randomized image where the value of each pixel is sampled from the original image M_{ro}, uniform distribution M_{ru}, or normal distribution M_{rn}.

Test Results. Since we are interested in how it changes from perturbing with a randomized map to perturbing with an attribution map computed by PoseIG, we define the changing ratio of difference on EPE R_E as:

$$R_E = \frac{|E_{attr} - E_{origin}|}{|E_{rand} - E_{origin}|} \tag{2}$$

where E_{attr}, E_{rand} and E_{origin} is the EPE of the input image perturbed with the PoseIG attribution map, randomized map, and without perturbation, respectively. Since our attribution map is computed joint-wise, we evaluate the EPE of that joint correspondingly. If R is more significant than 1, it means that the performance perturbed with the generated attribution map is changed more compared to perturbing with randomized maps, and it is likely more faithful. We show the perturbation test results on Simple Baseline ResNet50 [22] of various joints with different perturbation masks in Tab. A.



(a) (b)

Figure D. Groups of (a) human joints and (b) hand joints. The groups noted with color are (a) trunk joints (red), branch joints (orange), and leaf joints (yellow); (b) wrist (red), MCP (orange), PIP (yellow), DIP (green), and TIP (blue). The blue arrow indicates the relationship between the child and parent along the kinematic chain.

Joints	ResNet50	ResNet101	HRNet-W32	Transpose	Integral	RLE
trunk	1.961	1.906	1.849	1.863	1.865	1.754
branch	1.986	1.949	1.926	1.907	1.907	1.795
leaf	2.028	1.999	1.978	1.990	1.946	1.853

Table C. FI of different kinds of joints in each pose estimation model. Progressing down from the trunk joint to the leaf joint through the branch joint, the mean value of FI gets larger.

A.4. Model Randomization

We use Simple Baseline ResNet50 [22] as the example to conduct the model randomization test. Specifically, there are 54 convolutional layers in the model, and we successively randomize the parameters of these layers. After randomizing every nine layers, we compute the PoseIG attribution map of that randomized model. Apart from conducting the test qualitatively like previous methods [1], we also verify it quantitatively with the numerical indices. As Tab. B shows, the attribution maps have less FI and LI and higher DI, indicating that the attribution tends to change more with more corrupted parameters. Therefore, PoseIG is sensitive to the parameters in the model, so it can be used to diagnose the model. We also visualize this as shown in Fig. C.

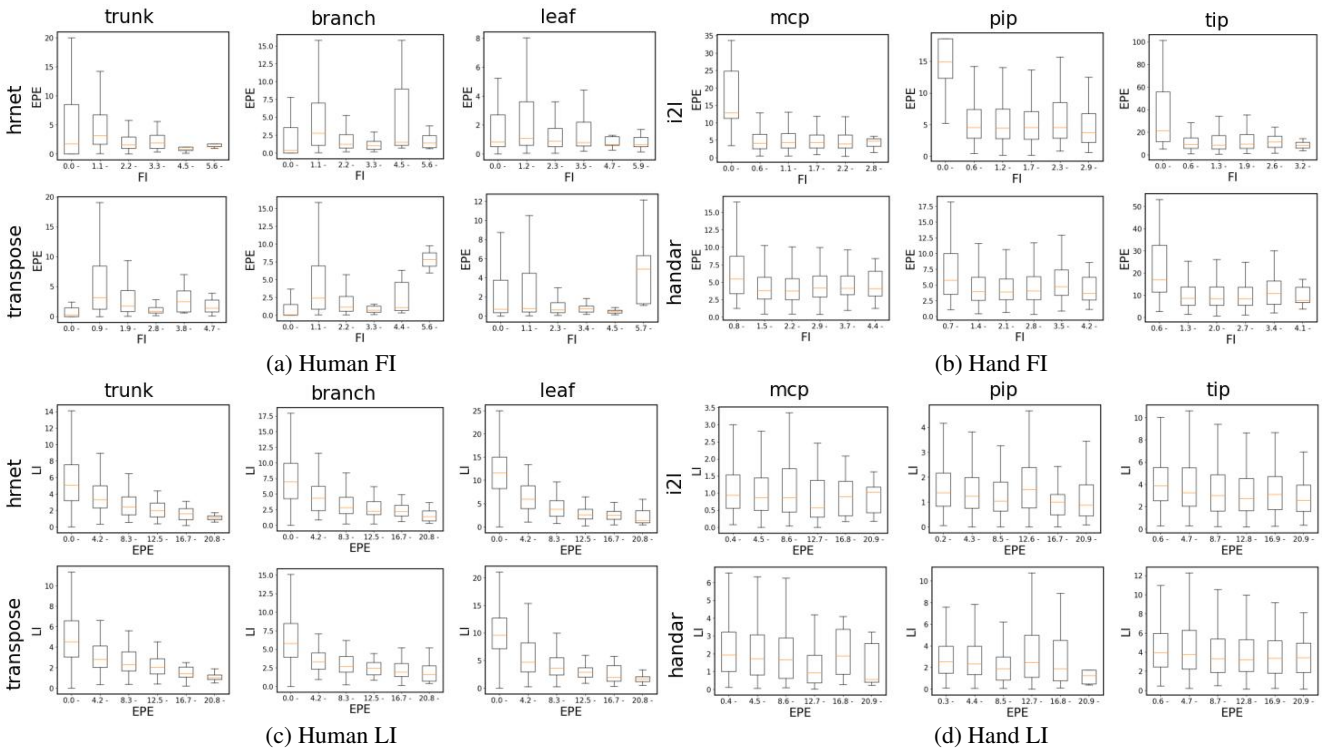
B. Joints Statistics

B.1. Joint grouping

Human. We divide human joints into three groups, namely trunk, branch, and leaf joints. As Fig. D shows, the shoulder and hip are categorized into trunk joints; the elbow and knee are categorized into branch joints; and the wrist, ankle, nose, eye, and ear are categorized into leaf joints.
Additionally, we define four kinds of joint pairs when discussing keypoint inversion [16], which are symmetric, child-as-parent, parent-as-child, and others. Specifically, symmetric pairs include left shoulder versus right shoulder, left elbow versus right elbow, left wrist versus right wrist, left hip versus right hip, left knee versus right knee, and left ankle versus right ankle; child-as-parent pairs include left elbow versus left shoulder, left hip versus left elbow, right elbow versus right shoulder, right hip versus right elbow, left knee versus left hip, left ankle versus left knee, right knee versus right hip, right ankle versus right knee; parent-as-child includes all the reversed ordered child-as-parent pairs. As illustrated in Fig. D (a), each arrow's starting point and ending point correspond to parent and child, respectively.
Hand. Hand joints are divided into five groups, including wrist, MCP, PIP, DIP, and TIP, based on the kinematic chain of hand and fingers. Refer to Fig. D (b) for detailed grouping information.

B.2. Quantitative Results

We provide the indices distribution of each joint group over more models, including CMR [4], MobRecon [5], I2l-MeshNet [13] and HandAR [21] for hand pose estimation. For human pose estimation, the results include ResNet50/ResNet101 [22], HRNet-W32 [18], TransPose [23], Integral Heatmap Regression [19] and Residual Log-likelihood Regression (RLE) [12].
Attribution versus MEPE. As Fig. E shows, for all the 2D human models, the trend is consistent with the trend in Subsec. 4.3 on EPE versus LI. Similarly, for all 3D hand pose models, the relationship between EPE versus FI holds the same. Additionally, it clearly shows that the accuracy of the hand model is more related to FI, while the accuracy of the human model is more related to LI.
Diffusion Index. In terms of DI, each human pose model shows a similar trend: progressing down the kinematic chain leads to higher DI. However, in hand pose estimation, such a trend is different. Specifically, the difference in DI among the joints of the hand is less. We postulate this is because 3D hand pose estimation requires more dispersed spatial information on the image to obtain depth information for all kinds of joints. Additionally, 3D hand pose is often obtained from a dense 3D hand mesh with successive regression. This may also lead to less difference in DI among joints.
Foreground Index. In 2D human pose estimation, FI gets larger, progressing down along the kinematic chain as Tab. C shows, which means joints closer to the root of the kinematic chain commonly require more global information. However, the trend is different in 2D hand pose estimation as MCP becomes the joint with the least FI, as discussed in the MCP shortcut.
Locality Index. As shown in Fig. F, progressing down along the kinematic chain, LI becomes higher. This indicates that the model prefers to use local image evidence more in 2D



(a) Human FI (b) Hand FI

(c) Human LI (d) Hand LI

Figure E. Attribution versus MEPE on different joints of two human models on **MS COCO** and two hand models on **FreiHand**. (a) FI versus MEPE on human models; (b) FI versus MEPE on hand models; (c) LI versus MEPE on human models; (d) LI versus MEPE on hand models. We can see that the performance of the human models is more influenced by LI, while FI influences the hand models more.

Index	ResNet50	ResNet101	HRNet-W32	Transpose	Integral*	RLE**
DI	4.03↑	3.71↑	3.58↑	3.12↑	2.68↑	2.17↑
FI	0.52↓	0.52↓	0.50↓	0.51↓	0.43↓	0.41↓

Table D. Difference of DI and FI between easy and hard cases among human models. (↑) indicates the number is higher in hard cases, while (↓) indicates it is lower in hard cases. (*) indicates implicit heatmap methods, and (**) indicates coordinate regression methods. The difference between the easy and hard cases of explicit heatmap methods is larger.

human pose estimation and 3D hand pose estimation.

C. Model Comparison

C.1. Regression versus Explicit Heatmap

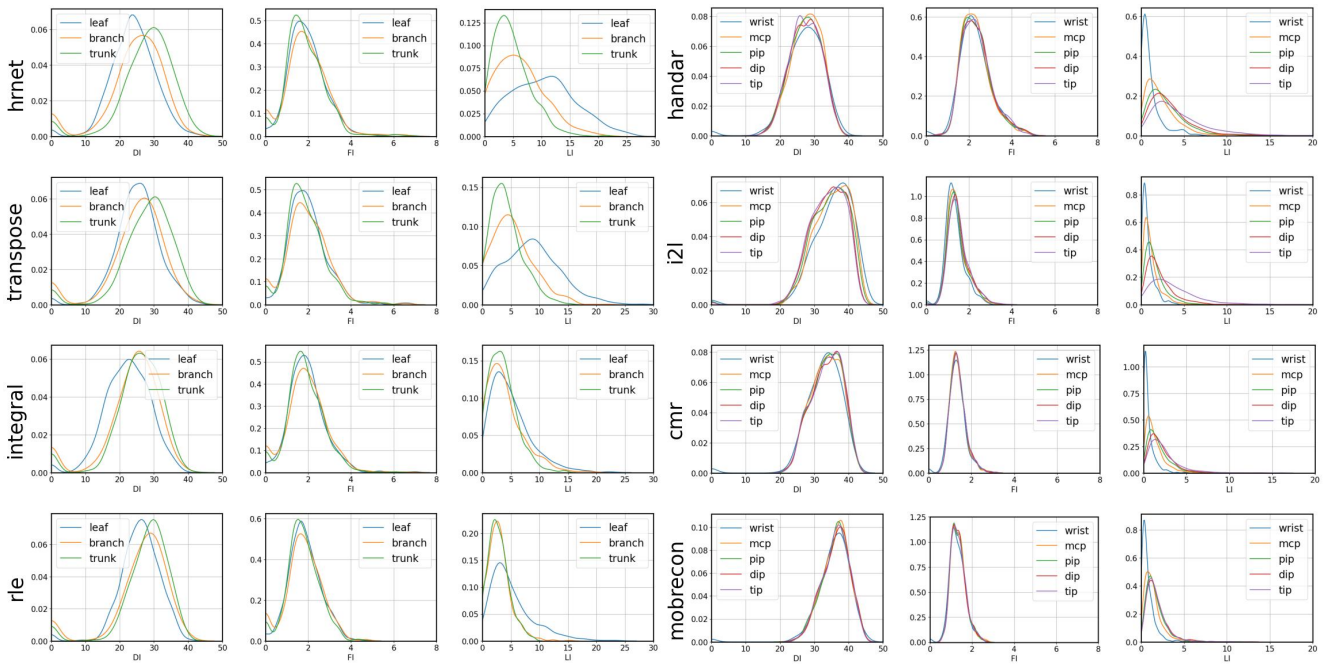
Comparing explicit heatmap methods such as HRNet [18], TransPose [23] and regression methods, including coordinate regression such as RLE [12] and integral regression (implicit heatmap) [19], in Fig. F, it is clear that the mean LI of the explicit heatmap methods are larger than the two other methods.

In terms of easy and hard cases, we show the difference of DI and FI between easy and hard cases for each model in Tab. D. DI increases while EPE and FI decrease significantly from predicting easy cases to hard cases. And we find that the difference is less among coordinate regression methods and integral regression. This indicates that the attribution maps of explicit heatmap methods are more sensitive to hard cases, which may be why this method performs worse on hard cases than the other two kinds of methods [9].

C.2. HRNet versus ResNet

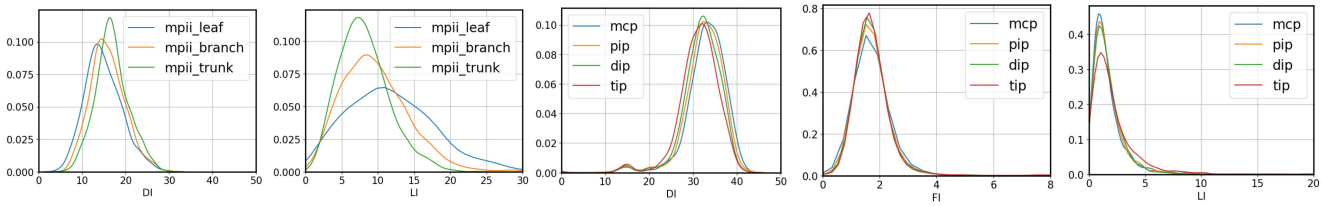
HRNet [18] has both higher DI and higher LI than Simple Baseline ResNet50/ResNet101 [22]. The mean DI of HRNet and Simple Baseline ResNet50 is 26.63 and 22.89, while the mean LI is 7.18 and 6.67, respectively. To investigate further, we analyze their attribution maps on different joint groups.

We find that the variance of DI and LI over three joint groups of HRNet is higher than Simple Baseline ResNet50. The variance of DI and LI of HRNet is 3.15 and 5.68, while that of Simple Baseline ResNet50 is 2.14 and 4.49. We postulate that utilizing features from different resolutions makes the attribution maps of HRNet differentiate more on joint groups. As shown in Fig. H, HRNet uses more dispersed information for trunk joints and more local image evidence for leaf joints.



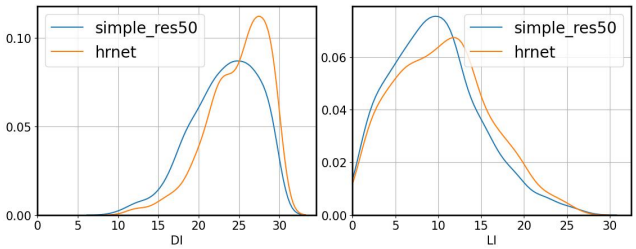
(a) Human Joints (b) Hand Joints

Figure F. Indices distribution of (a) human models on **MS COCO** and (b) hand models on **FreiHand**. The difference among each kind of joint is similar over various models. Progressing down on the kinematic chain in either human or hand leads to less LI.



(a) **MPII** (b) **DexYCB**

Figure G. Indices distribution of (a) human models on **MPII** and (b) hand models on **DexYCB**. The trend is similar to **MS COCO** and **FreiHand**, respectively. Since human masks are not provided in **MPII**, only DI and LI are computed for **MPII**.



(a) DI of trunk joints (b) LI of leaf joints

Figure H. Comparison between HRNet and Simple Baseline ResNet50 on (a) DI of trunk joints and (b) LI of leaf joints. HRNet has higher DI on trunk joints while higher LI on leaf joints.

C.3. Transformer versus CNN

As discussed in Subsec 4.4, TransPose [23], an explicit heatmap method using Transformer as the backbone, performs better on easy cases but worse on hard cases than HRNet based on CNN [18]. We find their EPE on easy/hard cases: TransPose(1.88/9.47) and HRNet(2.13/8.59). However, the attribution map of TransPose has less difference between easy and hard cases compared to other CNN-based explicit heatmap methods. We postulate that although TransPose utilizes more image evidence from the foreground without occlusion, it may mistakenly predict another similar joint in that foreground, *i.e.* keypoint inversion.

D. Model Diagnosis

Here, we show the refinement details in Subsec. 5.2. Assuming that we have the output of Simple Baseline, we target

refining the output with a refinement block by establishing an explicit pose topology. In this case, we use the GCN network like [6,24]. For human pose estimation, we construct a graph of joint heatmaps, $\mathcal{G} = (\mathcal{V}, A)$. Here, $\mathcal{V}$ is the heatmaps of human joints (*i.e.* , the output of Simple Baseline), and A is a 17×17 adjacency matrix for all the human joints. Specifically, A is the edge connections between the human joints. It is symmetrical and satisfies $A_{ij} = 1$ if joints i and j are the same or connected, and $A_{ij} = 0$ otherwise. The normalized Laplacian and the scaled Laplacian are defined based on the adjacency matrix A [6]. With the graph $\mathcal{G}$, we follow [24] and build the refinement block. For training, we freeze the parameters of Simple Baseline and only learn the refinement block using the supervision of heatmaps and the training strategy as [22]. During testing, we use the output of the refinement block as the final output.

E. Additional Dataset

Apart from **MS COCO** and **FreiHand**, we use additional datasets, including **MPII** [2] for human pose estimation and **DexYCB** [3] for hand pose estimation to conduct experiments on PoseIG. Specifically, we analyze the attribution maps of Simple Baseline ResNet50 [22] on **MPII** and HandOccNet [14] on **DexYCB**. As shown in Fig. G, the trends are similar to those on **MS COCO** and **FreiHand** in Fig. F. The conclusion in Subsec. B.2 holds for these two other datasets.

References

[1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In *NeurIPS*, 2018. 3

[2] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *CVPR*, June 2014. 6

[3] Yu-Wei Chao, Wei Yang, Yu Xiang, Pavlo Molchanov, Ankur Handa, Jonathan Tremblay, Yashraj S. Narang, Karl Van Wyk, Umar Iqbal, Stan Birchfield, Jan Kautz, and Dieter Fox. DexYCB: A benchmark for capturing hand grasping of objects. In *CVPR*, 2021. 6

[4] Xingyu Chen, Yufeng Liu, Chongyang Ma, Jianlong Chang, Huayan Wang, Tian Chen, Xiaoyan Guo, Pengfei Wan, and Wen Zheng. Camera-space hand mesh recovery via semantic aggregationand adaptive 2d-1d registration. In *CVPR*, 2021. 3

[5] Xingyu Chen, Yufeng Liu, Dong Yajiao, Xiong Zhang, Chongyang Ma, Yanmin Xiong, Yuan Zhang, and Xiaoyan Guo. Mobrecon: Mobile-friendly hand mesh reconstruction from monocular image. In *CVPR*, 2022. 3

[6] Hongsuk Choi, Gyeongsik Moon, and Kyoung Mu Lee. Pose2mesh: Graph convolutional network for 3d human pose and mesh recovery from a 2d human pose. In *ECCV*, 2020. 6

[7] Gabriel Erion, Joseph D Janizek, Pascal Sturmfels, Scott M Lundberg, and Su-In Lee. Improving performance of deep learning models with axiomatic attribution priors and expected gradients. *Nature Machine Intelligence*, 2021. 2

[8] Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *ICCV*, 2017. 2

[9] Kerui Gu, Linlin Yang, and Angela Yao. Removing the bias of integral pose regression. In *ICCV*, 2021. 4

[10] Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. In *NeurIPS*, 2019. 2

[11] Kuo-Wei Lee, Shih-Hung Liu, Hwann-Tzong Chen, and Koichi Ito. Silhouette-net: 3d hand pose estimation from silhouettes. *arXiv preprint arXiv:1912.12436*, 2019. 1

[12] Jiefeng Li, Siyuan Bian, Ailing Zeng, Can Wang, Bo Pang, Wentao Liu, and Cewu Lu. Human pose regression with residual log-likelihood estimation. In *ICCV*, 2021. 3, 4

[13] Gyeongsik Moon and Kyoung Mu Lee. I2l-meshnet: Image-to-lixel prediction network for accurate 3d human pose and mesh estimation from a single rgb image. In *ECCV*. Springer, 2020. 3

[14] JoonKyu Park, Yeonguk Oh, Gyeongsik Moon, Hongsuk Choi, and Kyoung Mu Lee. Handoccnet: Occlusion-robust 3d hand mesh estimation network. In *CVPR*, 2022. 6

[15] Vitali Petsiuk, Abir Das, and Kate Saenko. Rise: Randomized input sampling for explanation of black-box models. *arXiv preprint arXiv:1806.07421*, 2018. 2

[16] Matteo Ruggero Ronchi and Pietro Perona. Benchmarking and error diagnosis in multi-instance pose estimation. In *ICCV*, 2017. 3

[17] Wojciech Samek, Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller. Evaluating the visualization of what a deep neural network has learned. *IEEE TNNLS*, 2016. 2

[18] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, 2019. 3, 4, 5

[19] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *ECCV*, 2018. 3, 4

[20] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *ICML*, 2017. 1, 2

[21] Xiao Tang, Tianyu Wang, and Chi-Wing Fu. Towards accurate alignment in real-time 3d hand-mesh reconstruction. In *ICCV*, 2021. 3

[22] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *ECCV*, 2018. 2, 3, 4, 6

[23] Sen Yang, Zhibin Quan, Mu Nie, and Wankou Yang. Transpose: Keypoint localization via transformer. In *ICCV*, 2021. 3, 4, 5

[24] Xiaozheng Zheng, Pengfei Ren, Haifeng Sun, Jingyu Wang, Qi Qi, and Jianxin Liao. Rethinking regression-based method for 3d hand pose estimation. In *BMVC*, 2021. 6