Analyzing and Diagnosing Pose Estimation with Attributions

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

Abstract

We present Pose Integrated Gradient (PoseIG), the first interpretability technique designed for pose estimation. We extend the concept of integrated gradients for pose estimation to generate pixel-level attribution maps. To enable comparison across different pose frameworks, we unify different pose outputs into a common output space, along with a likelihood approximation function for gradient back-propagation.

To complement the qualitative insight from the attribution maps, we propose three indices for quantitative analysis. With these tools, we systematically compare different pose estimation frameworks to understand the impacts of network design, backbone and auxiliary tasks. Our analysis reveals an interesting shortcut of the knuckles (MCP joints) for hand pose estimation and an under-explored inversion error for keypoints in body pose estimation. Project page and code: https://qy-h00.github.io/poseig/.

1. Introduction

Human pose estimation of both the body and the hand is a critical vision task for augmented and virtual reality applications. State-of-the-art methods [12, 15, 19, 20, 33, 36] perform impressively on benchmarks but are difficult to compare beyond differences in average end-point-error (EPE). Averaged results on large-scale benchmarks depend on the underlying data distribution and tend to obscure the behaviour of pose estimation systems [10]. As such, we are motivated to find alternative ways to interpret and compare pose estimates across different methods. To that end, we present the first method for estimating pixel-level attribution maps designed specifically for pose estimation.

Integrated Gradients (IG) [35] is a commonly used attribution technique. IG and its derived variants [21, 35, 40] can produce pixel-level attribution maps for various image and natural language classification tasks. IG computes gradients to measure the relationship between changes to an input and changes to the target likelihood. However, IG is not directly applicable to pose estimation. Unlike in classification, where models always directly output a class likelihood, pose estimation models vary in their output, ranging from spatial likelihoods to regressed coordinates. Therefore, we must introduce a likelihood approximation function between the predicted outputs and their targets to approximate the target likelihood. Based on these likelihoods, we can back-propagate the gradients and generate attribution maps. Moreover, to enable meaningful comparison across frameworks, we propose unifying the different outputs into a common output space and use the same likelihood approximation function $S(\cdot)$ for back-propagation. Fig. 1 shows our interpretability pipeline for generating pixel-level attribution maps that can be compared across different pose frameworks.

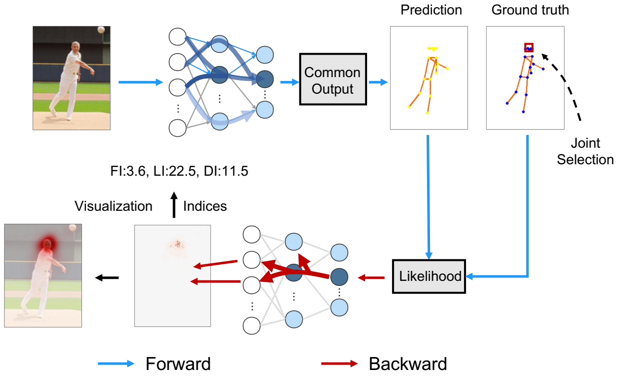


Figure 1. Pose Integrated Gradients (PoseIG) generates spatial attribution maps for pose estimation. Based on the attribution maps, we propose numerical indices to quantitatively characterize the attributions throughout the scene.

Existing works [35] and [21] focus on qualitative attributions and produce visualizations for single inputs. We are interested in these visualizations for pose estimation; however, we additionally target quantitative analysis of the attributions. As such, we have designed attribution-based indices to help analyze and diagnose pose estimation frameworks. Based on PoseIG's attribution maps, we introduce three indices to numerically characterize the attributions. The Foreground Index (FI) measures the extent to which the foreground is considered in the attributions. The Locality Index (LI) measures the amount of attribution around an im-

*Equal contribution

age coordinate, and the Diffusion Index (DI) measures how concentrated or dispersed the attributions are in the scene.

Armed with PoseIG's attribution maps and the associated indices, we study existing body and hand pose estimation frameworks and provide insights on their design and architectures. Finally, we diagnose existing models and find two overlooked issues in pose estimation. First, we reveal an artifically high performance of MCP[1] or knuckle joints in the hand, likely from shortcut learning as a result of data preprocessing. Secondly, we observe an under-explored phenomenon of keypoint inversion [27], where keypoints are mistakenly predicted at the location of other keypoints. Accordingly, we introduce simple mitigating solutions and recommend these be incorporated into future protocols to improve hand and body pose estimation.

Our main contributions can be summarized as follows:

- We introduce PoseIG, the first interpretability technique designed for pose estimation. PoseIG provides pixel-level attributions and can be applied to compare different pose estimation works based on a unified output space and a likelihood approximation function.
- We propose three numerical indices to quantitatively characterize the attributions in the scene.
- Using PoseIG's attributions and indices, we analyze and compare different body and hand pose estimation frameworks and provide insight on their design.
- We diagnose a shortcut problem in hand pose estimation and keypoint inversion errors in human pose estimation and propose simple solutions to alleviate these issues.

We hope it will serve as a useful tool to the community for analyzing, diagnosing, and improving pose estimation frameworks.

2. Related Work

Pose Estimation New architectures [22, 38, 39] and training strategies [12, 34] for pose estimation have steadily improved both 2D and 3D keypoint prediction accuracy. Heatmap methods predict an intermediate heatmap aligned to the joint coordinates, either explicitly [22, 37, 39] or implicitly [12, 34], while regression pipelines [15, 41] directly predict the numerical coordinates of keypoints. The preferred backbones for heatmap methods are fully convolutional, like Hourglass [22] and HRNet [37]. Regression methods use more diverse architectures, including transformers [17, 19] and Graph CNNs [4, 8]. Progress in human and hand pose estimation has been steady, but the improvements are hard to interpret beyond sheer accuracy. This naturally begs the question of how to interpret the benefits of new designs and architectures. To the best of our knowledge, our work is the first to explore interpretability for pose estimation.

Previously, [27] analyzed errors of pose estimation systems and highlighted jitters, misses, and keypoint inversions. With our indices, we further investigate keypoint inversion errors, in which a keypoint is predicted in the ground truth location of another keypoint.

Attribution Analysis Methods for attribution analysis are occlusion-based [25, 43], substitution-based [26, 29] or gradient-based [5, 30, 31, 40]. We focus on gradient-based methods because they provide pixel-level attributions. Early works [5, 21, 31] suffered from gradient saturation [35], but subsequent methods such as integrated gradients (IG) avoid saturation by integrating gradient magnitudes along an input continuum, *e.g.* , a linear interpolation of image intensities between a baseline and the original image. IG and IG-based methods have explored different baseline images [9, 40], integration paths [13] and gradient smoothing [6, 32]. Most of them are used for classification frameworks and are not directly applicable for pose estimation.

Additionally, most existing works only provide visualizations for qualitative analysis. To date, the only quantitative measure for attribution is the diffusion index [9], which measures the dispersion of the attribution. The diffusion index provides only a limited understanding of pose estimation; this work proposes additional indices to better characterize pose estimation frameworks.

3. Method

3.1. Preliminaries

Pose Estimation: For an input image crop of the hand or human body $\mathbf{x} \in \mathbb{R}^{m \times n}$, let $\mathbf{J} \in \mathbb{R}^{n_J \times d}$ denote the corresponding pose of n_J keypoints in d-dimensional space, where $d = 2$ or 3. The pose can be recovered by first encoding the image with $\mathbf{h} = \text{En}(\mathbf{x})$ and then decoding the image representation $\mathbf{h}$ into joint coordinates $\mathbf{J} = \text{De}(\mathbf{h})$.

Heatmap Methods learn a representation $\mathbf{h}$ in the form of a spatial heatmap that serves as a spatial likelihood of the joint. The heatmaps can be either explicit or implicit. In explicit heatmaps [22, 37], the joint location $\hat{\mathbf{J}}$ is decoded by an *argmax* operation:

$$\hat{\mathbf{J}} = \arg\max_{\mathbf{p}}(\mathbf{h}_{\mathbf{p}}). \tag{1}$$

For learning, an L2 loss is applied between $\mathbf{h}$ and a ground truth heatmap $\mathbf{h}^{\text{gt}}$,

$$L_{\text{ex}} = \sum_{\mathbf{p}} (\mathbf{h}_{\mathbf{p}} - \mathbf{h}^{\text{gt}}_{\mathbf{p}})^2, \tag{2}$$

where $\mathbf{h}^{\text{gt}}$ is generated by centering a Gaussian at the ground truth $\mathbf{J}_{\text{gt}}$ and $\mathbf{p}$ denotes the coordinates in the heatmap.

[1] MCP (metacarpophalangeal) joints are at the base of each finger and attach the finger to the palm.

For implicit heatmaps [12, 34], the joint location is decoded by an expectation and a *soft-argmax* with a learnable controlling parameter β on $\mathbf{h}$:

$$\hat{\mathbf{J}} = \sum_{\mathbf{p}} \frac{e^{\beta \mathbf{h}_{\mathbf{p}}}}{\sum_{\mathbf{p}'} e^{\beta \mathbf{h}_{\mathbf{p}'}}} \mathbf{p}. \tag{3}$$

For learning, since the soft-argmax is differentiable, an L2 loss can be applied on the joint locations:

$$L_{\text{im}} = \|\hat{\mathbf{J}} - \mathbf{J}^{\text{gt}}\|_2^2. \tag{4}$$

In the literature, the implicit heatmap method is also referred to as a latent heatmap [12] or integral regression [34].

Coordinate Regression Methods learn a latent representation $\mathbf{h}$ to directly regress the joint coordinates without any use of spatial representations such as heatmaps. The loss applied for learning is also an L2 loss between predicted joints and ground truth joints, as shown in Eq. 4.

Integrated Gradients [35] measure the contribution of input elements, *i.e.* image pixels, towards the prediction of the final network output. The contribution is defined as the integral of gradient magnitudes along a path from a given baseline to the input of interest. Usually, the baseline is a black image and the path is a linear interpolation of the image intensity between the baseline and the input image. From the baseline, the path gradually increases the intensity of the interpolated image in a specified manner and accumulates the attribution of the changes.

More specifically, consider function $F : R^{m\times n} \to [0, 1]$, where F is a pretrained deep network that maps an input 2D image of size $m \times n$ to a real-valued output. With a baseline image $\mathbf{z}$ that represents the absence of features from input $\mathbf{x}$, a straight-line path γ_l from the baseline $\mathbf{z}$ to the input $\mathbf{x}$ can be parameterized as

$$\gamma_l(\mathbf{x}, \mathbf{z}, \alpha) = \mathbf{z} + \alpha \cdot (\mathbf{x} - \mathbf{z}), \tag{5}$$

where α is the coefficient of interpolation. With the straight-line path, the integrated gradient $IG(\cdot)$ for an input x and its baseline $\mathbf{z}$ can be defined as:

$$IG(\mathbf{x}, \mathbf{z}) = (\mathbf{x} - \mathbf{z}) \cdot \int_{\alpha=0}^{1} \frac{\partial F(\gamma_l(\mathbf{x}, \mathbf{z}, \alpha))}{\partial \mathbf{x}} d\alpha. \tag{6}$$

The output of $IG(\mathbf{x}, \mathbf{z})$ is an attribution map of size $m \times n$.

3.2. PoseIG

Baseline and Path Function. The original IG method [35] recommended using black images as baselines. For pose estimation, we observe that a black image baseline results in attributions that are biased towards pixels with dark colors, *i.e.* darker pixels have a lower attribution.

The baseline image represents the absence of input features for prediction. Based on the observation that silhouettes and image edges are important for pose estimation [14], we introduce a Gaussian blur baseline:

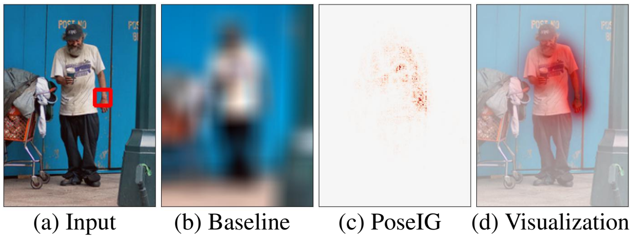


Figure 2. PoseIG components. (a) Input image and (b) its blur baseline, (c)-(d) attribution maps and corresponding kernel density estimation (KDE) heatmaps based on PoseIG.

$$I(\mathbf{x}, \sigma_p) = \mathbf{x} * K(\sigma_p), \tag{7}$$

where $\mathbf{x}$ is convolved with a Gaussian kernel $K(\sigma_p)$ with standard deviation σ_p. Accordingly, we also define a *blurry path* to capture attributions:

$$\gamma_b(\mathbf{x}, \sigma_p, \alpha) = \mathbf{x} * K(\sigma_p - \alpha\sigma_p), \tag{8}$$

where $\gamma_b(\mathbf{x}, \sigma_p, 0) = \mathbf{x} * K(\sigma_p)$ and $\gamma_b(\mathbf{x}, \sigma_p, 1) = \mathbf{x}$. The blurry path generates progressively less blurry images from the baseline $I(\mathbf{x}, \sigma_p)$ to the final image $\mathbf{x}$ progressively. We set $\sigma_p = 19$ to be relatively large, to ensure that the baseline has only coarse image evidence, while limiting potential intensity biases. Fig. 2 (a) shows an input image and its corresponding Gaussian blur baseline. Alternative baselines are explored in Supplementary Sec. A.

Likelihood Approximation and Output Space. Eq. 6 aims to accumulate the change in the target, *i.e.* class likelihood with changing inputs. As the outputs of pose estimation models are not a likelihood scalar but different output modalities (*e.g.* , heatmap vs coordinate), we define a differentiable **likelihood approximation function** for back-propagation based on the distance between prediction and ground truth. With a predicted pose $\hat{\mathbf{J}}$ and the corresponding ground truth pose $\mathbf{J}^{\text{gt}}$ in the metric space, we define the likelihood approximation function for k^{th} joint as:

$$S(\hat{\mathbf{J}}, \mathbf{J}^{\text{gt}}, k) = e^{-c||\hat{\mathbf{J}}_k - \mathbf{J}_k^{\text{gt}}||_2^2}, \tag{9}$$

where $\mathbf{J}_k^{\text{gt}}$ and $\hat{\mathbf{J}}_k$ are the k^{th} joint of $\mathbf{J}^{\text{gt}}$ and $\hat{\mathbf{J}}$, respectively. The hyper-parameter c is set to 0.3 empirically.

As we are interested in making numerical comparisons across different frameworks, it is important to determine a *common* output space. The exact space may vary depending on the models compared; what is important is that the mapping from the model output to common space be differentiable. For example, we can compare 2D heatmaps with coordinate regression in the 2D pose space by mapping the heatmaps with soft-argmax function. For convenience, we prefer to use 2D or 3D pose spaces as the output spaces. We define the pose estimation models together with the mapping to the designated common output space as $M(\cdot)$.

With a common output, the same likelihood approximation function $S(\cdot)$ can be used for back-propagation. The influence of different output spaces and likelihood approximation functions on attribution maps are shown in Supplementary Sec. A.

PoseIG. The attribution map G from PoseIG for k^{th} joint of $\mathbf{J}^{\text{gt}}$ is given as:

$$\begin{aligned} G &= PoseIG(\mathbf{x}, M, \sigma_p, \mathbf{J}^{\text{gt}}, k) \\ &= (\mathbf{x} - I(\mathbf{x}, \sigma_p)) \int_{\alpha=0}^{1} \frac{\partial S(M(\gamma_b(\mathbf{x}, \sigma_p, \alpha)), \mathbf{J}^{\text{gt}}, k)}{\partial \mathbf{x}} d\alpha. \end{aligned} \tag{10}$$

Visualization. Similar to [9], as a complement to the attribution map, we also visualize a heatmap of the attributions based on kernel density estimation (KDE) [23] to highlight the important areas[2]. For more instances of attribution maps, please refer to the Supplementary.

3.3. Attribution Indices

The attribution maps G from $PoseIG(\cdot)$ in Eq. 10 reveal the pixels that have the greatest contribution on the estimated pose. We introduce three indices to quantitatively characterize the attribution maps. Let G^s and G^m denote attribution maps normalized by dividing G by the sum or the max of G, respectively. Additionally, let g_i^s and g_i^m denote the value of the i^{th} element from G^s and G^m, respectively.

Foreground Index (FI). FI measures the extent to which the foreground, *i.e.* the body or the hand, is considered in the attribution and is defined as

$$FI = \frac{\sum_{i=1}^{N} g_i^s m_i}{\sum_{i=1}^{N} m_i} \times N, \tag{11}$$

where N is the number of pixels in G and m_i represents the value of the i^{th} pixel of a dilated binary foreground mask. The dilated mask is obtained from the segmentation mask provided as part of the annotations. We dilate with $\mu = 5$ pixels to give some border around the segment edge. Attributions that are located more on the foreground lead to a higher FI.

Locality Index (LI). LI quantifies the extent of attributions surrounding a joint coordinate. Given the normalized attribution G^s for joint A and the ground truth location $\mathbf{J}_X$ of joint X, $LI_A(X)$ is defined as

$$LI_A(X) = \frac{\sum_{i=1}^{N} g_i^s h(\mathbf{p}_i, \mathbf{J}_X)}{\sum_{i=1}^{N} h(\mathbf{p}_i, \mathbf{J}_X)} \times N, \tag{12}$$

$$\text{where} \;\; h(\mathbf{p}_i, \mathbf{J}_X) = \frac{1}{2\pi\sigma_l} \exp\left(-\frac{||\mathbf{p}_i - \mathbf{J}_X||_2^2}{2\sigma_l^2}\right). \tag{13}$$

Here, $h(\mathbf{p}_i, \mathbf{J}_X)$ represents the weight of the i^{th} pixel based on the distance between its location $\mathbf{p}_i$ in the normalized attribution map G^s and the given joint location $\mathbf{J}_X$. The $\sigma_l = 2$ is chosen empirically. A higher LI mean the higher contribution of pixels close to the designated location p. Ideally, $LI_A(X)$ has the maximum value if $X = A$, *i.e.* the attribution of G for joint A is the strongest around ground truth location J_A. By default, we discuss $LI_A(A)$ and abbreviate it to LI; otherwise we will specify the joints X and A for $LI_A(X)$.

Diffusion Index (DI). DI measures the spatial range of attributions in the image, and indicates how local or global the information used for the prediction is. We assume that different keypoints are determined by spatial information and have different extents of attributions and use the same DI as [9]:

$$DI = (1 - \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} |g_i^m - g_j^m|}{2N^2(\overline{g}^m)}) \times s, \tag{14}$$

where $\overline{g}^m$ is the average on g^m and s is a constant that scales the value to a reasonable range. We empirically set $s = 100$ as default. For attribution, DI reflects the spatial range of involved pixels; a larger DI indicates that the given model involves more pixels to make a decision.

4. Experiments

4.1. Datasets and Evaluation Metrics

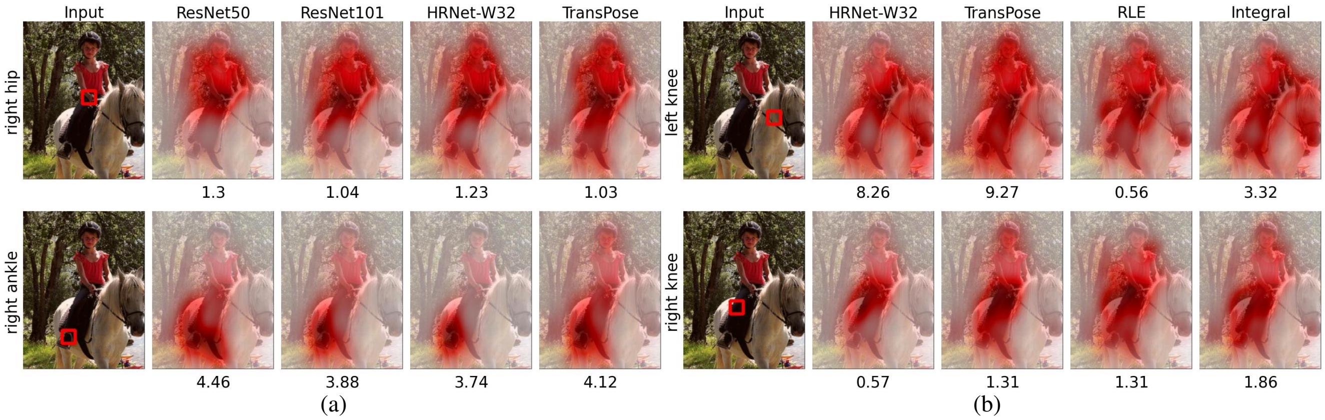
In this paper, we focus on RGB-based pose estimation methods for both the hand and the body. We report in the main paper results on FreiHand [46] and MS COCO [18]; results on additional datasets are given in Supplementary Sec. E. **FreiHand** is a real-world hand pose dataset with RGB images and 21 annotated 3D joints. **MS COCO** is a real-world body pose dataset with 250k person instances and 17 annotated 2D keypoints.

For hand pose estimation, we compare CMR [2], MobRecon [3], I2l-MeshNet [20] and HandAR [36]. For human pose estimation, we compare Simple Baseline ResNet50/ResNet101 [39], HRNet-W32 [33], TransPose [42], Integral Heatmap Regression [34] and Residual Log-likelihood Regression (RLE) [15]. Simple Baseline ResNet50/ResNet101, HRNet-W32 and TransPose are explicit heatmap methods. Integral Regression is an implicit heatmap method, and RLE is a coordinate regression model.

We evaluate the pose accuracy with mean end-point-error (MEPE) based on the average Euclidean distance between the predicted and ground truth joints. For calculating our indices, we follow [30] and select 300 samples with the greatest performance difference in the respective datasets' testing set.

To further explore the attributions, we group the joints based on the kinematic chain. The hand is divided into wrist

[2] For implementation, we use the function scipy.stats.gaussian_kde(·)



(a) (b)

Figure 3. Attribution KDE heatmaps with 2D EPE for human pose estimation models. (a) Comparison of trunk joint and leaf joint. All models prefer local image evidence on the leaf joint compared to the trunk joint; (b) Comparison of occluded cases and un-occluded cases. The coordinate regression RLE and implicit heatmap-based methods Integral Regression have lower EPE than explicit heatmap-based methods HRNet and TransPose on occluded joints.

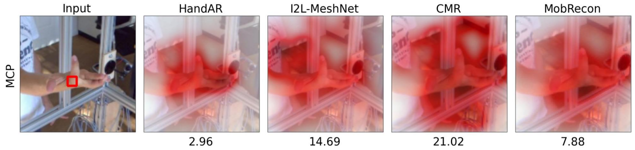


Figure 4. Attribution KDE heatmaps with 3D EPE for four hand pose estimation models. HandAR with hand segmentation as an auxiliary task prefers utilizing more image evidence from the hand area instead of the background.

joints, MCPs, PIPs, DIPs and fingertips; the body is divided into trunk joints (shoulder and hip), branch joints (elbow and knee) and leaf joints (head, ear, eye, ankle and wrist). Please see Supplementary Sec. B for a figure of the grouping).

4.2. Validation of Attributions

We verify the faithfulness of PoseIG's attributions with two commonly used techniques in the interpretability literature: model randomization and image perturbation. For **Model Randomization**, we follow the cascading randomization of [1] by corrupting more convolution parameters of successive layers to random values. The attributions of PoseIG have successively more changes in the scene as more layers are randomized. This confirms that PoseIG is sensitive to network parameters. For **Image Perturbation**, we follow [6,7,11,24,28] and perturb the pixels with the highest attribution magnitudes. The pose accuracy is more impacted when pixels with higher attributions are perturbed compared to random perturbation. Please see Supplementary Sec. A for more details.

4.3. Quantitative Analysis

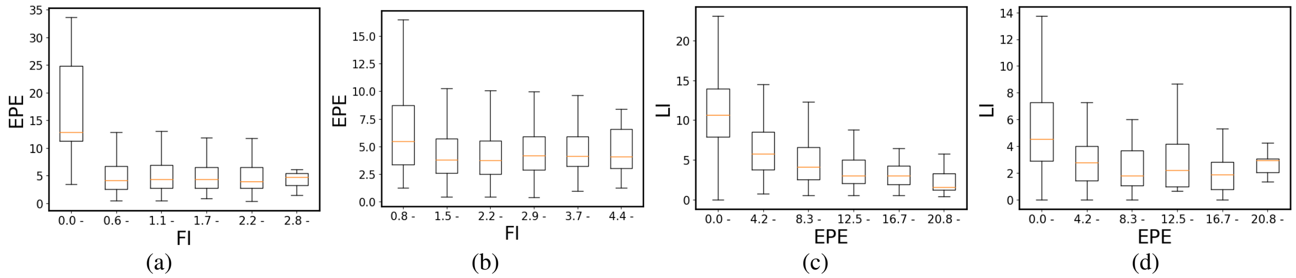
We use PoseIG to explore the common characteristics over different pose estimation models, including the relationship between attribution and MEPE, the attribution of different joints and influencing factors.

Attribution vs MEPE. We begin by analyzing the correlation of our indices and MEPE based on the statistics of MCPs for human hand and leaf joints. Fig. 5 shows the MEPE with respect to FI in (a)-(b) for two different hand pose estimation approaches. One observation is that most keypoints with 3D EPE larger than 15 mm have extremely low FI. Additionally, for tested hand models, it also illustrated that the range of EPE rapidly shrinks as FI increases.

As for LI, Figs. 5 (c)-(d) correspond to Simple Baseline with ResNet50 [39] and RLE [15] on body pose estimation. Both plots show that the predictions with EPE > 10 pixels mostly have low LI. This trend is more significant with the heatmap method but also exists with coordinate regression. We refer the reader to the Supplementary Sec. B for the exploration of other models and the comparison between the body and the hand.

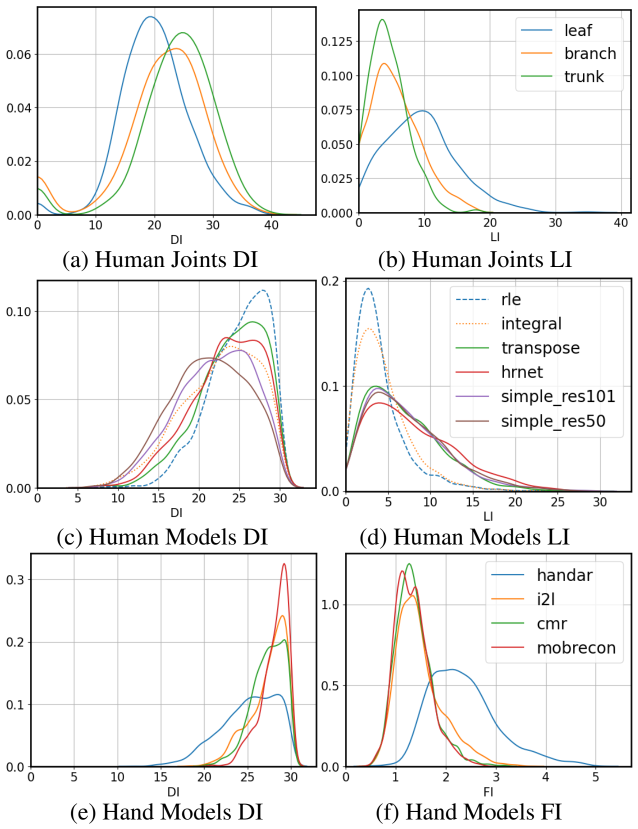
Different Joints. At the joint level, Fig. 6 (a)-(b) shows that as we progress down the kinematic chain away from the trunk joint, the attributions of all the models decrease in dispersion (a) and increase in localization (b), *i.e.* more image evidence comes from the local area around the ground truth location. On the other hand, the trunk joints and branch joints require more dispersed sets of pixels to determine their locations. This trend can be observed in the attributions in Figs. 3 (a) right hip versus right ankle. Similarly, hand pose estimation also shows the trend that PIPs, DIPs and TIPs further from the root of the kinematic chain require more local image evidence to make predictions. We refer the reader to Supplementary Sec. B for the exploration of hand pose estimation.

Influencing Factors. Same as [10,27], we divide MS COCO into easy, medium and hard cases by considering three factors, *i.e.* , the amount of joints or keypoints present (11-17, 6-10, 1-5), the percentage of occlusion ($<10\%$, 10-50%, $> 50\%$), and the largest dimension of the bounding box



(a) (b) (c) (d)

Figure 5. The MEPE of joints related to the corresponding FI and LI on the FreiHand dataset and the MS COCO dataset. (a)-(b) The box-plots of MEPE and FI with respect to MCPs for I2L-MeshNet [20] and HandAR [36]. (c)-(d) The box-plots of MEPE and FI with respect to leaf joints for Simple Baseline [39] and RLE [15].



(a) Human Joints DI (b) Human Joints LI

(c) Human Models DI (d) Human Models LI

(e) Hand Models DI (f) Hand Models FI

Figure 6. Index distributions estimated based on histogram. (a)-(b) correspond to DI and LI of different joints in the MS COCO dataset predicted by Simple Baseline ResNet50 [39]. (c)-(d) correspond to DI and LI of different body pose estimation methods, where regression methods is drawn as dashed line and implicit heatmap methods is drawn as dotted line. [15,33,39,42]. (e)-(f) are DI and FI, respectively, of various hand pose estimation methods [2, 3, 20, 36].

input (>128px, 96-128px, 64-96px, 32-64px).

For the various models, on the hard cases, LI and FI both decrease significantly while DI and EPE increase compared to the easy cases. In other words, models use less local image evidence and more background information for predicting the poses of challenging cases where only parts of the body are present and or visible. Comparing in Fig. 3 (b) the occluded left knee versus the visible right knee, it becomes clear that each model uses more dispersed image evidence for the occluded case. Additionally, we also find a reduction in FI from easy cases to hard cases. We refer the reader to Supplementary Sec. B for detailed statistics.

4.4. Exploration with PoseIG

We use PoseIG as a tool to study existing hand or body pose estimation frameworks [2, 3, 15, 16, 20, 36, 39, 41, 44].

Heatmap versus Coordinate Regression. Previous works [10] have postulated that implicit heatmaps use more global image evidence than explicit heatmaps. As Fig. 6 (d) shows, the models vary. The LI of implicit heatmaps [34] and coordinate regression (RLE [15]) is significantly lower than for explicit heatmaps [33, 39, 42]. Fig. 6 (d) clearly shows two different types of distributions, indicating that the loss (L2 on joint coordinate vs. heatmap) is likely the most important factor influencing the use of local image evidence.

Fig. 3 (a) shows that under occlusion, the implicit heatmap (Integral Regression) and coordinate regression (RLE) have lower EPE and use more information from the rest of the body than the explicit heatmap methods. This echoes the fact that explicit heatmap methods struggle with hard cases with occlusions, as stated in [10]. See Supplementary Sec. C for more details.

Architecture Differences. One of the claimed advantages of HRNet is that it can leverage features from different resolutions and capture more spatial information [33]. Fig. 6 (c) confirms that the attribution of HRNet is more dispersed than ResNet50 or ResNet101, *i.e.* it has a higher DI. However, it also has high LI, suggesting that it locates joints more precisely. As Fig. 3 (a) shows, for the trunk joint right hip, HRNet utilizes more global image evidence, including the head and leg area, compared to Simple Baseline ResNet50. In terms of the leaf joint right ankle, HRNet prefers more local spatial information than Simple Baseline ResNet50. See Supplementary Sec. C for more details.

Backbone Differences. Fig. 6 (c) shows that the DI of TransPose [42], a transformer backbone, is higher than all other heatmap-based CNN models but lower than coordinate regression models RLE [15]. This observation echoes the intuitive conjecture that transformers gather more global image evidence than CNNs [42]. Intuitively, more global information can help predict occluded joints more precisely. However, comparing the model performance with respect to the left knee in Fig. 3 (b), we find that TransPose even performs worse than other heatmap-based CNN-based methods, even though it utilizes more image evidence from the human body instead of the foreground, where the occluded joint locates, since it wrongly estimates the right knee as the left knee. More discussion on this can be found in Supplementary Sec. C.

Model Size. Comparing Simple Baseline [39] with ResNet50 versus ResNet101 backbones, it appears that model size has a limited effect on attribution. The most apparent difference is in the MS COCO dataset, where the larger model has lower EPE and higher DI, as shown in Fig. 6 (c). It is likely that the deeper model has a larger receptive field and therefore captures more dispersed image evidence. From the visualized examples in Fig. 3 (a), we speculate that increasing model size leads to a higher DI in trunk joints and a higher LI in leaf joints.

Auxiliary Task. Fig. 6 (e)-(f) shows that for hand pose estimation, adding hand segmentation as an auxiliary task [36] leads to higher FI and lower DI compared to the works without auxiliary tasks. This means that the model has a higher preference for using the pixels in the hand area to estimate the pose. As auxiliary hand segmentation is [36]'s main difference, we conjecture that a focus on foreground pixels is beneficial for lowering MEPE. An instance is visualized in Fig. 4, and it illustrates that HandAR prefers utilizing the image evidence on the hand area over the other models.

Human Body versus Hand. As stated, human body poses exhibit different characteristics from hand poses. We found that the performance of the human model is influenced by LI, while the performance of the hand model is influenced by FI. This difference is likely due to the inherent differences in the two tasks. Aside from the underlying pose, the appearance variations of body pose estimation arise from different clothing and partial presence in the scene. As such, body pose models need both local and background information. However, hands are always presented in full, with appearance variations from self-occlusion and object interactions [45, 46]. As such, more information on the hand mask area is required, and not only on the area around the joint itself, *i.e.* , global foreground information.

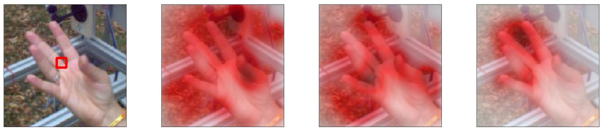
5. Model Diagnosis

5.1. MCP Shortcut

When exploring 2D hand pose estimation with PoseIG, we find that the MCP joints (the base of each finger) are the most accurate with the lowest EPE, yet they also have the lowest FI and the highest DI, *i.e.* they are mainly predicted based on pixels from the background.

Index	Joint	baseline	w/ AC	w/ crop noise
MEPE	NON-MCP	15.23/41.04	16.49/17.27	14.93/15.14
	MCP	12.46/43.89	12.56/12.87	12.3/11.68
FI	NON-MCP	1.81/2.46	1.77/2.73	1.77/2.83
	MCP	1.75/2.28	1.8/2.87	1.85/3.01
LIR	MMCP	16.25%	10.00%	6.56%

Table 1. Indices of attribution generated by the models tested on original/toy testing sets. With 'augment-then-crop' or crop noise, the reduction of performance on toy testing sets is much less.



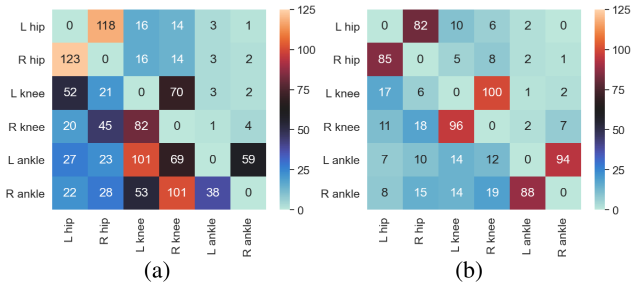
(a) input (b) baseline (c) w/ AC (d) w/crop noise

Figure 7. (a) input image, with the target joint marked as a red square; (b)-(d) attribution KDE heatmaps for baseline, w/ crop noise and w/ AC respectively. We can see that w/ crop noise and w/ AC alleviate the unusual phenomenon effectively.

To further investigate, we filter the samples with low interpretability - a sample is considered low-interpretable if both its MEPE and FI fall in the first quartile over all the joints during testing. This criteria suggests that the prediction is relatively accurate but uses little foreground spatial information. We further define the Low-Interpretablity Rate (LIR) as the proportion of low-interpretable samples for a given type of joint, *e.g.* MCP or PIP, and check for joints with high LIR, *i.e.* low interpretability. In coordinate regression, the LIR of MCP is 7.44% while the LIR of other joints is 2.7%; the LIR of MMCP (MCP of the mid finger) reaches 16.25%.

A visualization of a low-interpretability case (see Fig.. 7 (a)) reveals that the attribution concentrates on the background of the image. We speculate that some shortcut learning is occurring, likely due to the pre-processing strategy used in the data augmentation. The default used by most works is a *'crop-then-augment'* [16, 41, 44, 45] strategy, versus the less common *'augment-then-crop'* [20, 36].

'Crop-then-augment' sometimes introduces black borders; these borders are likely a shortcut for the MCP joints since the MCP joints are commonly located near the image center. On the contrary, *'augment-then-crop'* does not produce such borders. To verify, we create a toy dataset with hand bounding boxes that have perturbed cropping centers and cropping scales. We take coordinate regression as an example and train models with different settings. Specifically, we use the default training setting, *i.e.* , *'crop-then-augment'* and translation augmentation of [-20,20] pixels, as baseline. We further investigate the training with an additional crop center noise of [-20,20] pixels (w/ crop noise) and using *'augment-then-crop'* (w/ AC) instead of *'crop-*



(a) (b)

Figure 8. Keypoint inversion confusion matrices, where the row/column combination denotes the ground truth/predicted joint ID. It annotates frequency of inversion for (a) Simple Baseline ResNet50 and (b) Simple Baseline ResNet50 refined with GCN. The frequency for predicting left ankle as left knee decreases from 101 to 14, indicating refinement makes the model suffer less from child-as-parent inversion.

Category	Symmetric	Child-as-Parent	Parent-as-Child	Other
w/ Keypoint Inversion	5.37/4.71	4.91/4.63	4.88/4.15	3.97/3.71

Table 2. $LI_A(B)$ for an ordered pair of joints (A, B). The figure shows the average value of ResNet50/ResNet50 refined with GCN, where the keypoint inversion is detected on the original model's prediction. And refinement with GCN significantly decreases $LI_A(B)$ of the ordered pairs with keypoint inversion.

then-augment'.

As shown in Tab. 1, the baseline seemingly has a lower MEPE due to the contribution of the MCP. However, it also has a high LIR and its performance decreases significantly if testing on the toy dataset. When training with w/ AC or w/ crop noise, this phenomenon is significantly alleviated. The KDE heatmaps in Fig. 7 (b)-(d) also verify that the model training with w/ AC or w/ crop noise will make predictions based on pixels around MMCP. We therefore conclude that the shortcut is due to the combination of the black borders and the relatively static crop centers.

5.2. Keypoint Inversion

Keypoint inversion [27] is an error in pose estimation where the model predicts a keypoint A near the ground truth location of keypoint B, *e.g.* when the left knee is predicted at the location of the right knee or vice versa. Formally, [27] defines keypoint inversion for an ordered pair of joints (A, B) if $||\hat{\mathbf{J}}_A - \mathbf{J}_B|| < \epsilon$ and $||\hat{\mathbf{J}}_A - \mathbf{J}_A|| > \epsilon$. Here, $\hat{\mathbf{J}}_A$ is the model prediction on joint A, $\mathbf{J}_A$ and $\mathbf{J}_B$ is the ground truth location of joint A and joint B, ϵ is the error threshold. In our experiments, we use $\epsilon = 5$. For example, on Simple Baseline ResNet50 [39], we observe that about 16.7% of the cases of keypoints with EPE > 10pxs are keypoint inversions.

For a more detailed characterization, we define four categories of inversions and note their frequency: symmetric pair (*e.g.* left wrist as right wrist, 21.2%), child-as-parent (*e.g.* left ankle as left knee 28.7%), parent-as-child (*e.g.* right elbow as right wrist, 10.7%) and others (3.4%). For details on the groupings, please see Supplementary Sec. B. Fig. 8 (a) shows, taking the lower body as the instance, that the most common inversions occur on symmetric pairs of trunk joints and predicting leaf joints as branch joints.

To further investigate, we analyze keypoint inversion with the LI index. For each ordered pair of joints (A, B), we compute $LI_A(B)$, and average on each pairs of a group, as shown in Tab. 2. We find that $LI_A(B)$ of keypoint pairs with inversions are significantly larger than non-problematic pairs. This reveals that the keypoint inversion commonly occurs with utilizing the image evidence near the incorrect keypoint.

To alleviate this, we add a Graph Convolutional Network (GCN) [44] as a refinement block after the output of the network to establish an explicit topology for poses. The refinement block takes the predictions of Simple Baseline as input and outputs refined predictions. See Supplementary Sec. D for details.

After refinement, three kinds of inversions decrease significantly and we note their percentage change as follows: child-as-parent (-68.6%), parent-as-child (-49.5%), others (-38.2%). We postulate tat this is related to the attribution map. As Tab. 2 shows, refinement with GCN decreases $LI_A(B)$ on the pairs with keypoint inversion predicted by the original baseline, which means that the prediction depends less on the spatial information near other joints. In terms of symmetric keypoint inversion, although the mean of $LI_A(B)$ decreases over the original error, it increases over the symmetric keypoint inversion detected in the refined model from 4.26 to 4.30. In other words, this refinement makes it depend more on the symmetric joint. As Fig. 8 (b) shows, it introduces additional symmetric keypoint inversion (+12.2%).

6. Discussion

In this paper, we introduce the gradient-based interpretability technique PoseIG, as well as three indices to analyze pose estimation. Using PoseIG and these interpretative tools, we provide insight to understand the attributions for existing pose estimation works. Moreover, we show the approach's potential to diagnose and improve pose estimation frameworks. In the future, we would like to perfect the details (*e.g.* , hyper-parameters, baselines) of PoseIG, explore PoseIG for multi-person pose estimation and provide more interpretative tools to the pose estimation community.

Acknowledgments This research / project is supported by the Ministry of Education, Singapore, under its MOE Academic Research Fund Tier 2 (STEM RIE2025 MOE-T2EP20220-0015).

References

[1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In *NeurIPS*, 2018. 5

[2] 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. 4, 6

[3] 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. 4, 6

[4] 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. 2

[5] Dumitru Erhan, Y. Bengio, Aaron Courville, and Pascal Vincent. Visualizing higher-layer features of a deep network. *Technical Report, Univeristé de Montréal*, 2009. 2

[6] 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, 5

[7] Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *ICCV*, 2017. 5

[8] Liuhao Ge, Zhou Ren, Yuncheng Li, Zehao Xue, Yingying Wang, Jianfei Cai, and Junsong Yuan. 3d hand shape and pose estimation from a single rgb image. In *CVPR*, 2019. 2

[9] Jinjin Gu and Chao Dong. Interpreting super-resolution networks with local attribution maps. In *CVPR*, 2021. 2, 4

[10] Kerui Gu, Linlin Yang, and Angela Yao. Removing the bias of integral pose regression. In *ICCV*, 2021. 1, 5, 6

[11] Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. In *NeurIPS*, 2019. 5

[12] Umar Iqbal, Pavlo Molchanov, Thomas Breuel Juergen Gall, and Jan Kautz. Hand pose estimation via latent 2.5 d heatmap regression. In *ECCV*, 2018. 1, 2, 3

[13] Andrei Kapishnikov, Subhashini Venugopalan, Besim Avci, Ben Wedin, Michael Terry, and Tolga Bolukbasi. Guided integrated gradients: An adaptive path method for removing noise. In *CVPR*, 2021. 2

[14] 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. 3

[15] 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. 1, 2, 4, 5, 6, 7

[16] Moran Li, Yuan Gao, and Nong Sang. Exploiting learnable joint groups for hand pose estimation. In *AAAI*, 2021. 6, 7

[17] Kevin Lin, Lijuan Wang, and Zicheng Liu. End-to-end human pose and mesh reconstruction with transformers. In *CVPR*, 2021. 2

[18] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 4

[19] Weian Mao, Yongtao Ge, Chunhua Shen, Zhi Tian, Xinlong Wang, Zhibin Wang, and Anton van den Hengel. Poseur: Direct human pose regression with transformers. *arXiv preprint arXiv:2201.07412*, 2022. 1, 2

[20] 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. 1, 4, 6, 7

[21] T Nathan Mundhenk, Barry Y Chen, and Gerald Friedland. Efficient saliency maps for explainable ai. *arXiv preprint arXiv:1911.11293*, 2019. 1, 2

[22] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, 2016. 2

[23] Emanuel Parzen. On estimation of a probability density function and mode. *The annals of mathematical statistics*, 33(3), 1962. 4

[24] Vitali Petsiuk, Abir Das, and Kate Saenko. Rise: Randomized input sampling for explanation of black-box models. *arXiv preprint arXiv:1806.07421*, 2018. 5

[25] Vitali Petsiuk, Rajiv Jain, Varun Manjunatha, Vlad I Morariu, Ashutosh Mehra, Vicente Ordonez, and Kate Saenko. Black-box explanation of object detectors via saliency maps. In *CVPR*, 2021. 2

[26] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Model-agnostic interpretability of machine learning. *arXiv preprint arXiv:1606.05386*, 2016. 2

[27] Matteo Ruggero Ronchi and Pietro Perona. Benchmarking and error diagnosis in multi-instance pose estimation. In *ICCV*, 2017. 2, 5, 8

[28] 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. 5

[29] Ludwig Schallner, Johannes Rabold, Oliver Scholz, and Ute Schmid. Effect of superpixel aggregation on explanations in lime–a case study with biological data. *arXiv preprint arXiv:1910.07856*, 2019. 2

[30] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, 2017. 2, 4

[31] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013. 2

[32] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017. 2

[33] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, 2019. 1, 4, 6

[34] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *ECCV*, 2018. 2, 3, 4, 6

[35] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *ICML*, 2017. 1, 2, 3

[36] Xiao Tang, Tianyu Wang, and Chi-Wing Fu. Towards accurate alignment in real-time 3d hand-mesh reconstruction. In *ICCV*, 2021. 1, 4, 6, 7

[37] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *TPAMI*, 43(10), 2020. 2

[38] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *CVPR*, 2016. 2

[39] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *ECCV*, 2018. 2, 4, 5, 6, 7, 8

[40] Shawn Xu, Subhashini Venugopalan, and Mukund Sundararajan. Attribution in scale and space. In *CVPR*, 2020. 1, 2

[41] Linlin Yang, Shile Li, Dongheui Lee, and Angela Yao. Aligning latent spaces for 3d hand pose estimation. In *ICCV*, 2019. 2, 6, 7

[42] Sen Yang, Zhibin Quan, Mu Nie, and Wankou Yang. Transpose: Keypoint localization via transformer. In *ICCV*, 2021. 4, 6, 7

[43] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *ECCV*, 2014. 2

[44] 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, 7, 8

[45] Christian Zimmermann and Thomas Brox. Learning to estimate 3d hand pose from single rgb images. In *ICCV*, 2017. 7

[46] Christian Zimmermann, Duygu Ceylan, Jimei Yang, Bryan Russell, Max Argus, and Thomas Brox. Freihand: A dataset for markerless capture of hand pose and shape from single rgb images. In *ICCV*, 2019. 4, 7