HOGWARTS: Mitigating Perspective Distortion for 6DoF Head Pose Estimation via Virtual Camera Space

Sungho Chun
Department of ECE
Kwangwoon University
Seoul, Republic of Korea
asw9161@kw.ac.kr

Ju Yong Chang
Department of ECE
Kwangwoon University
Seoul, Republic of Korea
jychang@kw.ac.kr

Abstract

Estimating 6-degrees-of-freedom (6DoF) head pose from a single RGB image remains challenging under strong perspective distortion. We observe that the widely used design choice of rectangular face cropping introduces a geometric inconsistency. This leads not only to image–label mismatch under 2D rotation augmentation but also exacerbates perspective ambiguity, making it difficult to distinguish projection distortion from the actual appearance and pose of faces located near the image periphery. To address this issue, we propose HOGWARTS, a geometry-driven framework that combines established virtual-camera normalization via homography warping with a dynamic virtual bounding-box correction for metric 6DoF head pose estimation. By explicitly defining a virtual camera that shares the optical center with the original camera and warping the input image into this space, our method compensates for off-axis distortion in a geometrically consistent manner. However, with the virtual focal length held fixed, retaining the original bounding-box size after image normalization leaves a position-dependent scale factor in the regression targets of the pinhole-based translation formulation. We analytically characterize this dependence and derive a dynamic virtual bounding-box correction that separates this geometric factor from the translation parameters learned by the network. Consequently, HOGWARTS enables more geometrically consistent and robust 6DoF pose estimation across varying image locations. Experimental results on the ARKitFace and BIWI datasets demonstrate that HOGWARTS consistently outperforms prior methods under severe perspective distortion as well as in cross-domain settings. Code is available at https://github.com/asw91666/HOGWARTS_RELEASE.

1 Introduction

Estimating 6-degrees-of-freedom (6DoF) head pose from a single RGB image is a fundamental problem in computer vision. It plays a crucial role in a wide range of applications, including augmented and virtual reality (AR/VR), human–computer interaction (HCI), driver monitoring systems, and 3D avatar animation, where accurate and robust pose estimation is essential. With the rapid advancement of deep learning, significant progress has been made in 6DoF head pose estimation [1, 30, 19, 5]. In particular, recent approaches such as PerspNet [19] and TRG [5] have demonstrated strong performance even under severe perspective distortion, such as selfie scenarios where the camera is close to the subject.

Despite the success of these previous studies, the preprocessing pipeline commonly adopted by existing methods, namely simply extracting the detected face region using a rectangular crop, has inherent geometric limitations. We identify two key issues through which this standard cropping operation can hinder learning in the task of 6DoF head pose estimation. First, rectangular cropping

40th Conference on Neural Information Processing Systems (NeurIPS 2026).



Figure 1: Visualization of 6DoF head pose estimation from a single RGB image under severe perspective distortion. The green bounding box denotes the conventional rectangular crop used by previous methods, while the red bounding box denotes the virtual bounding box introduced by HOGWARTS through the proposed virtual camera formulation. The gold and blue meshes represent the predictions of the SOTA method [6] and HOGWARTS, respectively. By geometrically compensating for off-axis distortion, HOGWARTS produces substantially more accurate pose and geometry estimation, particularly near the image periphery.

leads to an inherent inconsistency when combined with 2D rotation augmentation. When a face located away from the optical axis exhibits strong off-axis perspective distortion, applying arbitrary in-plane rotations to the cropped image produces pixel configurations that violate the assumption of the pinhole camera model. Rotating only the pose labels accordingly fails to reflect the actual geometric transformation of the image. This results in an image–label mismatch, where the augmented data violates the underlying image formation process, potentially hindering stable and consistent model optimization.

Second, cropping removes global spatial context that is essential for resolving perspective-induced ambiguities. Existing models that rely solely on cropped inputs [19, 11, 41] must implicitly infer whether the observed appearance variation arises from true 3D geometry or from off-axis projection effects. Recent works [5, 6] attempt to alleviate this issue by incorporating bounding box position and scale as auxiliary inputs, encouraging the model to reason about the global pose in the original camera space. However, such approaches still rely on the network to implicitly learn complex, non-linear relationships between spatial location and geometric distortion, which complicates optimization and increases data dependency.

To address these limitations, we propose HOGWARTS (Head pOse estimation via Geometric WARping in virTual camera Space), a framework that combines virtual-camera image normalization with a position-dependent virtual bounding-box formulation for metric 6DoF head pose estimation. Building on prior perspective-normalization approaches [38, 13, 21], we use homography warping to express the input image in a canonical virtual camera space. This normalization helps mitigate the aforementioned issues. First, since off-axis distortions are compensated for in the virtual camera space, applying 2D rotation augmentation helps preserve geometric consistency between the image and pose labels, mitigating the image–label mismatch problem (see Fig. 3 in the appendix). Second, normalizing off-axis appearance reduces the network's need to implicitly model location-dependent distortion patterns, allowing it to focus on learning pose-relevant appearance features in a canonicalized space, which can facilitate optimization and generalization (see Fig. 1). However, appearance normalization alone does not account for the effect of this transformation on metric translation estimation. With the focal length held fixed, the virtual-camera transformation changes the target's depth and projected scale according to its viewing direction. If the original bounding-box size is retained in the translation formulation, the network must implicitly absorb this position-dependent scale variation into its predictions. We therefore derive a position-dependent virtual bounding-box

correction and incorporate it into pinhole-based translation recovery, analytically separating this geometric factor from the translation parameters learned by the network. The inference pipeline proceeds as follows. We first warp the target region from the original image into the virtual camera space using the corresponding homography transformation. A neural network then predicts the 6DoF head pose in this normalized space. Finally, the estimated pose is mapped back to the original camera coordinate system using the known geometric relationship between the virtual and original cameras.

We evaluate HOGWARTS on the ARKitFace dataset, which represents scenarios with frequent and severe perspective distortion, such as selfies. Furthermore, to assess robustness under unseen cameras and environments, we additionally evaluate cross-domain generalization on BIWI. Across both severe perspective distortion scenarios and cross-domain settings, HOGWARTS consistently outperforms existing state-of-the-art (SOTA) models.

The main contributions of this paper are summarized as follows:

- We analytically show that retaining the original bounding-box size after virtual-camera normalization leaves a position-dependent scale factor in the pinhole-based translation regression targets.
- Based on this analysis, we introduce HOGWARTS for metric 6DoF head pose estimation. Its dynamic virtual bounding-box correction analytically separates this geometric factor from the translation parameters learned by the network.
- We demonstrate that HOGWARTS consistently and significantly outperforms SOTA methods on the ARKitFace and BIWI datasets.

2 Related Works

2.1 Head Pose Estimation

Most existing head pose estimation studies have focused on estimating 3DoF rotation, such as yaw, pitch, and roll, from a single image [2, 18, 15, 3, 39, 16, 4, 33, 32]. These methods have improved rotation representations, uncertainty modeling, and robustness in unconstrained environments, but their problem setting differs from full 6DoF head pose estimation, which additionally requires estimating 3D translation with respect to the camera coordinate system.

Recent studies have extended this task to full 6DoF head pose estimation. Optimization-based methods [19, 11, 41] first predict 2D landmarks or 3D face geometry and then compute the pose through optimization methods such as the Perspective-n-Point (PnP) algorithm [20], but they heavily depend on the accuracy of the intermediate geometric prediction. In contrast, direct regression methods [1, 30, 5, 6] directly estimate the 6DoF pose from the input image. In particular, TRG and TRGv2 improve global pose estimation by using both the cropped image and the spatial location information of the bounding box, but perspective distortion itself still has to be implicitly learned by the neural network.

2.2 Mitigating Perspective Distortion

In single-image pose estimation, off-axis distortion, which occurs as the target on the image plane moves farther away from the principal point, distorts the relationship between appearance and 3D pose and degrades estimation performance. Implicit location-aware methods [22, 5, 29] use the location or size of the bounding box as an additional input to the neural network to compensate for global spatial information lost during cropping. However, these approaches are closer to implicitly learning the effects of perspective distortion from data rather than explicitly removing its geometric cause.

In contrast, explicit normalization methods aim to geometrically correct perspective distortion before pose estimation. In human pose estimation, PCL [38] and PersPose [13] defined a virtual camera space, and in head pose estimation, Li et al. [21] showed that homography-based rectification is effective for head rotation estimation. PersPose also addresses metric root translation within its perspective-normalized 3D human pose estimation framework.

Building on these normalization approaches, we analyze the geometric effects of virtual-camera normalization on bounding-box-based metric translation estimation. With the virtual focal length

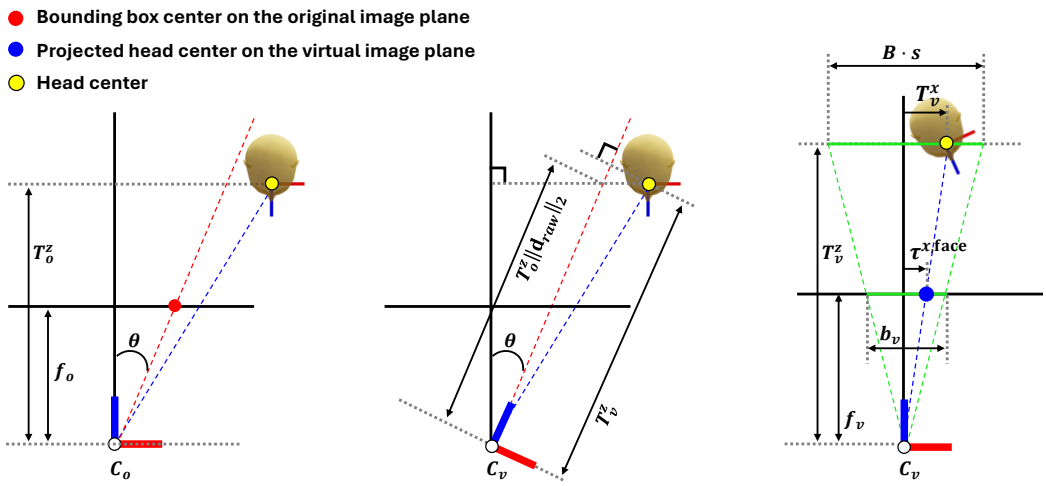


Figure 2: Visualization of the original camera space C_o and the proposed virtual camera space C_v. The virtual camera shares the same optical center as the original camera, while its optical axis is rotated toward the target bounding box center. This geometric formulation enables homography-based normalization of off-axis perspective distortion.

held fixed, the transformation changes the target's depth in camera coordinates and its projected scale as a function of viewing direction. In the adopted pinhole-based translation formulation, retaining the original bounding-box size leaves a position-dependent scale factor in the regression targets, requiring the network to account for this factor in its predictions. We analytically characterize this dependence and derive a dynamic virtual bounding-box correction that separates the known geometric factor from the translation parameters learned by the network.

3 Proposed Method

3.1 Overview

The goal of this study is to estimate the 6DoF pose of the head coordinate system C_h relative to the original camera coordinate system C_o, namely the rotation $R_o \in SO(3)$ and translation $T_o \in \mathbb{R}^3$, from a single RGB image I_o. To this end, we define a virtual camera that shares the optical center with the original camera but is rotated such that its optical axis points toward the center of the target bounding box. The virtual camera coordinate system C_v attached to this camera is then utilized for pose estimation. Fig. 2 illustrates the original and virtual camera coordinate systems.

The overall pipeline consists of three steps. First, the original image I_o is transformed into the virtual camera image I_v through homography warping. Next, I_v undergoes crop and resize operations before being input to the neural network, which estimates the head pose $R_v \in SO(3), T_v \in \mathbb{R}^3$ in the virtual camera space. Finally, the estimated pose R_v, T_v in the virtual space is transformed into the final pose R_o, T_o in the original camera space using the geometric relationship between the original and virtual cameras.

3.2 Defining Virtual Camera and Homography Warping

Following prior work [38, 13, 21], we define a virtual camera space and apply the corresponding homography warping to normalize input images for pose estimation. For clarity of mathematical formulation, we assume a pinhole camera model with zero skew and square pixels, and denote the intrinsic parameter matrix of the original camera as $K_o \in \mathbb{R}^{3\times3}$ (focal length f_o, principal point (c_{ox}, c_{oy})). The rotation matrix ${}^oR_v \in SO(3)$ required to define the pose of the virtual camera is determined from the target bounding box information in the original image and the original intrinsic parameter K_o. Here, ${}^oR_v$ denotes the rotation matrix that transforms a 3D vector defined in the virtual camera coordinate system into the original camera coordinate system.

First, letting (u_{oc}, v_{oc}) denote the center of the bounding box in the original image, the unnormalized viewing ray $\mathbf{d}_{\text{raw}}$ pointing from the optical center of the original camera to that point is derived as follows:

$$\mathbf{d}_{\text{raw}} = \left[\frac{u_{oc} - c_{ox}}{f_o}, \frac{v_{oc} - c_{oy}}{f_o}, 1 \right]^\top. \tag{1}$$

Let the normalized ray direction be $\mathbf{d} = [d_x, d_y, d_z]^\top = \frac{\mathbf{d}_{\text{raw}}}{\|\mathbf{d}_{\text{raw}}\|_2}$. Then, ${}^oR_v$ is defined as the shortest-arc rotation that aligns the canonical optical axis $\mathbf{z} = [0, 0, 1]^\top$ of the virtual camera with $\mathbf{d}$. Specifically, the rotation axis orthogonal to both vectors is given by $\mathbf{v} = \mathbf{z} \times \mathbf{d} = [-d_y, d_x, 0]^\top$, and the rotation angle is $\theta = \arccos(d_z)$. Using Rodrigues' rotation formula, ${}^oR_v$ can be analytically computed as follows:

$${}^oR_v = I + [\mathbf{v}]_\times + \frac{1}{1 + d_z}[\mathbf{v}]_\times^2, \tag{2}$$

where $I \in \mathbb{R}^{3\times3}$ is the identity matrix and $[\mathbf{v}]_\times$ denotes the skew-symmetric matrix of $\mathbf{v}$.

Since the optical axis of the virtual camera points toward the bounding box center, the principal point (c_{vx}, c_{vy}) of the virtual camera is set to the exact center of the warped virtual image, and the focal length f_v is set identical to the focal length f_o of the original camera ($f_v = f_o$). Together, these parameters define the virtual camera intrinsic matrix $K_v \in \mathbb{R}^{3\times3}$.

The homography matrix $H \in \mathbb{R}^{3\times3}$, which maps a pixel q_o on the original image plane to a pixel q_v on the virtual image plane to mitigate perspective distortion, is calculated as follows:

$$q_v \sim H q_o, \quad H = K_v ({}^oR_v)^\top K_o^{-1}, \tag{3}$$

where $q_o, q_v \in \mathbb{R}^3$ denote pixel coordinates expressed in homogeneous coordinates.

3.3 Head Pose Estimation in Virtual Space

Given the original bounding-box size b_o obtained from the face detector, we define the virtual bounding-box size as $b_v = \frac{b_o}{\|\mathbf{d}_{\text{raw}}\|_2}$ (derived in Sec. 3.5). The virtual image I_v obtained through homography warping is cropped around the principal point (c_{vx}, c_{vy}) with a square window of size $b_v \times b_v$ and resized into $\tilde{I}_v \in \mathbb{R}^{3\times H\times W}$, which is then used as the input to the neural network f_θ. This study uses an existing 6DoF head pose estimation model [6] as the base architecture, and the network predicts the head rotation R_v and the parameter $\psi_v \in \mathbb{R}^3$ for estimating the 3D translation T_v in the virtual camera space:

$$R_v, \psi_v, V_h, L_v = f_\theta(\tilde{I}_v), \tag{4}$$

where $\psi_v = [s, \tilde{\tau}^{x,\text{face}}, \tilde{\tau}^{y,\text{face}}]^\top$. Here, s is a learned scale factor that adjusts the canonical head size B in metric units, and $\tilde{\tau}^{x,\text{face}}, \tilde{\tau}^{y,\text{face}}$ are the normalized 2D offsets on the virtual image plane. The normalized offset $\tilde{\tau}^{\text{face}}$ is converted into the actual pixel-unit offset τ^{face}, and this relationship is expressed as $\tau^{\text{face}} = \frac{b_v}{2}\tilde{\tau}^{\text{face}}$. Additionally, $V_h \in \mathbb{R}^{3\times N^V}$ and $L_v \in \mathbb{R}^{2\times N^L}$ represent the 3D face mesh vertices in the head coordinate system C_h and the 2D sparse landmarks on the virtual image plane, respectively, which are used for auxiliary supervision [6].

As shown in Fig. 2, the depth $T_v^z \in \mathbb{R}$ in the virtual camera space is calculated using the scale–depth relationship of the pinhole camera model. Specifically, we assume that, in the virtual camera space, the head is enclosed by a box of size $B \cdot s$, and the depth of this box is equal to the head depth. The projection of this box onto the virtual image plane corresponds to the virtual bounding box. Based on this relationship, we obtain the following equations:

$$T_v^z = \frac{B \cdot s}{b_v} f_v, \quad T_v^x = \frac{T_v^z}{f_v}\tau^{x,\text{face}}, \quad T_v^y = \frac{T_v^z}{f_v}\tau^{y,\text{face}}, \tag{5}$$

where $b_v = \frac{b_o}{\|\mathbf{d}_{\text{raw}}\|_2}$ is the virtual bounding-box size. The derivation of this scaling relation and its effect on the translation regression targets are presented in Sec. 3.5 and Appendix A.4.

3.4 Pose Transformation to Original Space

Since the virtual camera and the original camera share the same optical center, the transformation between the two coordinate systems is determined solely by the rotation ${}^oR_v$. Therefore, the estimated

pose R_v, T_v in the virtual camera space is transformed into the final pose R_o, T_o in the original camera space as follows:

$$R_o = {}^{o}R_v R_v, \quad T_o = {}^{o}R_v T_v. \tag{6}$$

Summarizing the process of calculating T_o from ψ_v, the final translation in the original camera space can be expressed at once as follows:

$$T_o = {}^{o}R_v \begin{bmatrix} \frac{Bs}{2}\tilde{x}^{\text{face}} \\ \frac{Bs}{2}\tilde{y}^{\text{face}} \\ \frac{Bsf_v}{b_v} \end{bmatrix}. \tag{7}$$

This equation summarizes the entire translation estimation process of the proposed method.

3.5 Derivation and Analysis of the Virtual Bounding-Box Correction

Position-dependent factor in scale regression with a fixed bounding box. In this section, we determine the virtual bounding-box size to compensate for the depth increase induced by the virtual-camera transformation, preventing this increase from introducing an additional multiplicative factor into the scale regression target.

We fix the focal lengths of the original and virtual cameras to the same value, $f_v = f_o = f$. Let $r = \|\mathbf{d}_{\text{raw}}\|_2 = \frac{1}{\cos\theta}$, and let $\mathbf{d} = \frac{\mathbf{d}_{\text{raw}}}{r}$ denote the normalized ray passing through the bounding-box center. Assuming that the head center lies close to this ray, its position in the original camera coordinate system can be approximated as $T_o \approx T_o^z \mathbf{d}_{\text{raw}}$. Since the virtual camera's optical axis is aligned with $\mathbf{d}$, the corresponding depth satisfies

$$T_v^z = \mathbf{d}^\top T_o \approx r T_o^z. \tag{8}$$

We express the virtual bounding-box size using a correction factor $c(r)$ to be determined:

$$b_v = c(r) b_o. \tag{9}$$

Rearranging the depth estimation term in Eq. 5, the scale parameter required to represent a given depth is

$$s = \frac{T_v^z b_v}{Bf} = \frac{T_v^z c(r) b_o}{Bf} \approx r c(r) \frac{T_o^z b_o}{Bf}. \tag{10}$$

Here, $\frac{T_o^z b_o}{Bf}$ is the scale value obtained by applying the relation in Eq. 5 to the depth and bounding-box size in the original camera coordinate system. Retaining the original bounding-box size corresponds to $c(r) = 1$, leaving an additional position-dependent factor r in the scale required to represent the virtual depth.

Derivation of the dynamic bounding-box correction. We seek to account for this known factor through the bounding-box size. To this end, we require the product of the depth increase factor r and the bounding-box correction factor $c(r)$ to equal one:

$$r c(r) = 1 \quad \Longrightarrow \quad c(r) = \frac{1}{r} \quad \Longrightarrow \quad \boxed{b_v = \frac{b_o}{r}}. \tag{11}$$

This choice satisfies $c(1) = 1$, preserving the original bounding-box size on the optical axis. Moreover,

$$T_v^z b_v \approx (r T_o^z) \frac{b_o}{r} = T_o^z b_o. \tag{12}$$

Thus, under the above depth approximation, the scale parameter s no longer needs to absorb the additional multiplicative factor associated with the increase in virtual depth. The relationships between the translation regression targets under the fixed and dynamic bounding-box configurations are derived in Appendix A.4.

4 Experimental Results

4.1 Implementation Details

The input image resolution is set to $H \times W = 192 \times 192$. The number of vertices in the 3D face mesh, N^V, and the number of 2D sparse landmarks, N^L, are set to 1,220 and 68, respectively. All performance evaluations of HOGWARTS in this paper are based on a model trained solely on the ARKitFace training set. Detailed training settings, training procedures, and training time are provided in the appendix.

4.2 Datasets

ARKitFace [19] provides 6DoF head poses, 3D face mesh vertices, and intrinsic camera parameters. This dataset was collected in selfie environments where the distance between the camera and the face is less than 1 m and is therefore strongly affected by perspective distortion. Following the protocols of [19, 5], we used 717,840 frames from 400 subjects for training and 184,884 frames from 100 subjects for testing.

BIWI [9] dataset provides 6DoF head poses and intrinsic camera parameters. BIWI provides one 3D face mesh per subject, which serves as the ground-truth mesh for ADD evaluation. However, frame-wise mesh annotations are not available, so we evaluate only 6DoF head pose estimation performance on this dataset. To verify the generalizability of the proposed method, BIWI was used exclusively for testing, and following prior works [37, 19], predicted bounding boxes were used as inputs. In addition, to remove the head translation bias arising during cross-dataset evaluation, the final translation estimation performance was measured after applying the reference system alignment proposed in [6].

4.3 Evaluation Metrics

To comprehensively evaluate 6DoF head pose estimation and 3D face reconstruction performance, we follow the protocols of prior works [18, 37, 19, 1]. Head rotation accuracy is measured using roll, pitch, and yaw errors, their mean absolute error (MAE_r), and geodesic error (GE). Head translation performance is evaluated using the errors along the x, y, and z axes (t_x, t_y, t_z) and their mean absolute error (MAE_t). In particular, during cross-dataset evaluation, the final error is computed after applying reference system alignment [6] to correct the coordinate system bias inherent across datasets.

In addition, we use the average 3D distance (ADD) metric [17, 19, 5, 6] to jointly evaluate rotation and translation accuracy. ADD is computed as

$$\text{ADD} = \frac{1}{N^V} \sum_{n=1}^{N^V} \left\| \left(R_o V_{h,n}^* + T_o\right) - \left(R_o^* V_{h,n}^* + T_o^*\right) \right\|_2 , \tag{13}$$

where $V_{h,n}^* \in \mathbb{R}^3$ denotes the n-th ground-truth mesh vertex in the head coordinate system C_h, and N^V is the number of vertices. The pairs (R_o, T_o) and (R_o^*, T_o^*) denote the final predicted and ground-truth poses, respectively, expressed in the original camera coordinate system C_o. Both transformations use the same ground-truth mesh, and ADD averages the Euclidean distances between the resulting corresponding vertices. Thus, ADD measures the combined effects of rotation and translation errors without using the predicted mesh.

To assess the 3D face mesh reconstruction accuracy of the proposed model, we additionally report the median and mean distances between the prediction and the ground-truth meshes in the head coordinate system C_h [19]. Rotation errors are reported in degrees, while translation errors, ADD, and reconstruction distance metrics are reported in millimeters (mm).

4.4 Ablation Study

Geometric modeling: In this paragraph, we conduct ablation experiments to analyze the importance of the virtual camera space model (VC) for mitigating perspective distortion and the pinhole-camera-based translation formulation. Direct-T directly regresses 3D translation without this formulation,

Table 1: Ablation study of the proposed HOGWARTS framework on the ARKitFace and BIWI datasets. Direct-T denotes a baseline that directly regresses 3D translation without the pinhole-camera-based translation formulation. VC denotes virtual-camera image normalization via homography warping. All models in this table are trained under the same protocol as HOGWARTS for a controlled ablation. The lower block compares fixed and dynamic virtual bounding-box sizes within HOGWARTS.

Method	ARKitFace			BIWI		
	GE	MAE_t	ADD	GE	MAE_t	ADD
Ablation of geometric components						
Direct-T	1.70	3.62	8.67	5.57	21.22	52.74
VC + Direct-T	1.68	3.59	8.61	5.63	17.89	44.32
TRGv2 (retrained)	1.70	3.70	8.94	5.55	11.45	29.64
HOGWARTS (Ours)	**1.67**	**3.57**	**8.58**	**5.48**	**10.50**	**27.04**
Ablation of virtual bounding box formulation						
$b_v = b_o$	**1.67**	3.67	8.82	5.54	11.22	28.91
$b_v = b_o / \|\mathbf{d}_{raw}\|_2$ (Ours)	**1.67**	**3.57**	**8.58**	**5.48**	**10.50**	**27.04**

while VC + Direct-T additionally applies VC. In this experiment, TRGv2 is retrained under the same training conditions as HOGWARTS and is labeled TRGv2 (retrained) in the table. The experimental results are summarized in Table 1. According to the table, Direct-T and VC + Direct-T show reasonable performance on ARKitFace but exhibit limited generalization performance on BIWI. In contrast, TRGv2 and HOGWARTS, both of which use the pinhole-camera-based translation formulation, demonstrate stable generalization performance even on BIWI. This suggests that the pinhole-camera-based formulation provides a geometric prior that enables more stable inference on unseen test data.

However, the pinhole-camera-based translation formulation alone cannot explicitly decouple the off-axis distortion and appearance deformation that vary non-linearly with the spatial location of the bounding box. Consequently, in TRGv2, which uses this formulation without VC, the neural network must implicitly learn projection distortion patterns that depend on spatial location. This leads to degraded generalization performance when evaluated on datasets with different camera intrinsic parameters or unfamiliar bounding box distributions compared to the training data. In contrast, HOGWARTS, which incorporates VC and dynamic virtual bounding-box sizing, can focus primarily on 6DoF pose inference without additionally modeling complex distortion patterns. The consistent performance improvements observed on both ARKitFace and BIWI demonstrate that such a geometrically integrated approach can significantly improve generalization performance on unseen distributions.

Virtual bounding box size: In this study, we analyze the performance difference between two methods for defining the virtual bounding box size b_v during 6DoF pose estimation in the virtual camera space. The first is a fixed bounding box formulation ($b_v = b_o$), which directly uses the bounding box size b_o detected by the face detector. The second is a dynamic bounding box formulation ($b_v = \frac{b_o}{\|\mathbf{d}_{raw}\|_2}$), which scales the virtual bounding box size according to the length of the bounding box center ray $\mathbf{d}_{raw}$.

According to Table 1, changing the bounding box formulation from the fixed method to the dynamic method leaves GE on ARKitFace unchanged, while improving MAE_t and ADD from 3.67/8.82 to 3.57/8.58, respectively. A similar trend is observed on BIWI, where GE, MAE_t, and ADD improve from 5.54/11.22/28.91 to 5.48/10.50/27.04, respectively. These results suggest that the proposed dynamic formulation has a more direct impact on head translation estimation than on head rotation estimation.

As discussed in Sec. 3.5, this performance improvement arises because the geometric scale variation caused by the projection characteristics of the pinhole camera model is explicitly compensated for at the mathematical level, thereby reducing the optimization difficulty of the neural network. If the fixed formulation ($b_v = b_o$) is used, the network must implicitly encode the position-dependent factor $\|\mathbf{d}_{raw}\|_2$ into the scale parameter s to estimate the increasing depth (T_v^z) in the virtual camera space. In contrast, the dynamic formulation explicitly decouples this position-dependent projection scale variation, allowing the neural network to focus primarily on estimating the physical scale of the subject, which significantly improves translation estimation accuracy.

Table 2: Comparison with SOTA methods for 6DoF head pose estimation on the ARKitFace test dataset. Rotation errors are reported in degrees, while translation and ADD metrics are reported in millimeters. HOGWARTS achieves the best performance across both rotation and translation metrics, demonstrating the effectiveness of the proposed virtual camera formulation under severe perspective distortion.

Method	Yaw	Pitch	Roll	MAE_r	GE	t_x	t_y	t_z	MAE_t	ADD
img2pose [1, 19]	5.07	7.32	4.25	5.55	-	1.39	3.72	15.95	7.02	20.54
Direct 6DoF Regression [19]	1.86	2.72	1.03	1.87	-	2.80	5.23	19.16	9.06	21.39
Refined Pix2Pose [28, 19]	1.95	2.62	2.48	2.35	-	2.43	4.23	35.33	14.00	36.44
JMLR [11]	1.13	1.75	0.61	1.16	2.39	0.98	2.48	11.13	4.86	11.87
PerspNet [19]	0.98	1.43	0.55	0.99	-	1.00	2.41	9.73	4.38	10.30
MPC [36]	0.96	1.38	0.59	0.98	-	0.97	2.06	10.02	4.35	10.21
TRG [5]	0.89	1.30	0.57	0.92	1.80	0.83	1.88	8.22	3.64	8.74
TRGv2 [6]	0.85	1.26	0.52	0.87	1.76	0.80	1.90	8.23	3.64	8.76
HOGWARTS (Ours)	**0.79**	**1.21**	**0.47**	**0.82**	**1.67**	**0.78**	**1.85**	**8.10**	**3.57**	**8.58**

Table 3: Comparison with SOTA methods for 3D face mesh reconstruction on the ARKitFace test dataset. Distances are reported in millimeters. HOGWARTS achieves the most accurate face geometry reconstruction, demonstrating that geometrically compensating for off-axis distortion improves not only pose estimation but also 3D facial geometry recovery.

Method	Median	Mean
PRNet [10]	1.97	2.05
3DDFA-v2 [12]	2.35	2.31
Deng *et al.* [8]	2.46	2.55
JMLR [11]	1.86	1.94
PerspNet [19]	1.72	1.76
MPC [36]	1.66	1.74
TRG [5]	1.55	1.61
TRGv2 [6]	1.57	1.59
HOGWARTS (Ours)	**1.52**	**1.53**

Additional analyses: Performance across off-axis angles and the effect of 2D rotation augmentation are further analyzed in Appendix A.5 and Appendix A.6, respectively.

4.5 Comparison with State-of-the-Art Methods

In this section, we compare the proposed HOGWARTS framework with existing SOTA 6DoF head pose estimation methods. The quantitative evaluation results on the ARKitFace [19] and BIWI [9] datasets are summarized in Tables 2, 3, and 4.

Optimization-based methods such as JMLR [11], PerspNet [19], and MPC [36] first estimate 2D landmarks in the image space and 3D head coordinates, and then derive the 6DoF pose through the PnP algorithm [20]. For such approaches to operate accurately, the subject-specific 3D facial geometry must be reliably reconstructed. However, these methods typically rely on rectangularly cropped original images without explicitly compensating for off-axis distortion. Under such settings, separating the intrinsic facial geometry of the subject from appearance deformation caused by projection distortion becomes highly ill-posed. As a result, reconstruction errors in the estimated 3D geometry directly propagate to pose estimation errors, leading to higher ADD errors than HOGWARTS on both ARKitFace and BIWI.

Regression-based methods such as TRG [5] and TRGv2 [6] directly estimate 6DoF pose by jointly using the cropped image and the bounding box information from the original image. This strategy encourages the neural network to implicitly model off-axis distortion patterns that vary non-linearly across image regions. While such approaches can achieve strong performance under in-distribution settings, their generalization performance tends to degrade when evaluated on unseen camera intrinsics or unfamiliar bounding box distributions not observed during training.

In contrast, HOGWARTS explicitly decouples off-axis distortion at the geometric level through the proposed virtual camera model. This alleviates the need for the neural network to implicitly model complex spatial distortion patterns, thereby simplifying the optimization process. As a result, HOGWARTS achieves superior 6DoF pose estimation performance compared to existing baselines on both ARKitFace and BIWI. Furthermore, it demonstrates substantially improved 3D face reconstruction accuracy even in ARKitFace, a close-range selfie environment with severe perspective

Table 4: Comparison with SOTA methods for 6DoF head pose estimation on the BIWI dataset. HOGWARTS achieves the best overall performance on BIWI, demonstrating strong cross-domain generalization under unseen camera intrinsics and image distributions.

Method	Yaw	Pitch	Roll	MAE_r	GE	t_x	t_y	t_z	MAE_t	ADD
3DoF rotation estimation methods										
HopeNet [31]	4.81	6.61	3.27	4.89	9.53	-	-	-	-	-
QuatNet [18]	4.01	5.49	2.94	4.15	-	-	-	-	-	-
FSA-Net [37]	4.27	4.96	2.76	4.00	7.64	-	-	-	-	-
MNN [34]	3.98	4.61	2.39	3.66	-	-	-	-	-	-
TriNet [2]	**3.05**	4.76	4.11	3.97	-	-	-	-	-	-
6DRepNet [15]	3.24	4.48	2.68	3.47	-	-	-	-	-	-
Cao *et al.* [3]	4.21	3.52	3.10	3.61	-	-	-	-	-	-
TokenHPE [39]	3.95	4.51	2.71	3.72	-	-	-	-	-	-
DAD-3DNet [26]	3.79	5.24	2.92	3.98	-	-	-	-	-	-
MFDNet [23]	3.40	4.68	2.77	3.62	-	-	-	-	-	-
Cobo *et al.* (6D) [7]	4.58	4.65	2.71	3.98	7.30	-	-	-	-	-
HRHPE [24]	3.38	4.34	2.52	3.41	-	-	-	-	-	-
Xu et al. [35]	3.83	3.98	2.32	3.38	-	-	-	-	-	-
FaceXFormer [27]	3.91	3.97	2.67	3.52	-	-	-	-	-	-
Full 6DoF pose estimation methods										
img2pose [1]	4.57	3.55	3.24	3.79	7.10	-	-	-	-	-
MPC [36]	3.67	**3.26**	2.16	3.03	-	-	-	-	-	-
JMLR [11]	6.31	6.17	3.72	5.40	8.61	3.77	4.59	31.65	13.33	35.54
PerspNet [19]	3.06	3.48	2.12	2.89	5.61	3.13	4.00	29.80	12.31	32.00
TRGv2 [6]	3.12	3.60	1.90	2.87	5.73	**2.99**	4.13	26.89	11.33	29.25
HOGWARTS (Ours)	3.10	3.35	**1.87**	**2.77**	**5.48**	3.12	**3.61**	**24.77**	**10.50**	**27.04**

distortion. These results suggest that HOGWARTS effectively addresses projection distortion, which has remained a major challenge in 6DoF head pose estimation.

5 Conclusion and Limitation

In this paper, we addressed the fundamental limitations of conventional cropping strategies in 6DoF head pose estimation. To overcome these limitations, we proposed HOGWARTS, a framework that combines established virtual-camera image normalization with a position-dependent virtual bounding-box formulation. We analytically showed that, with the virtual focal length held fixed, retaining the original bounding-box size after normalization leaves a position-dependent scale factor in the pinhole-based translation regression targets. Based on this analysis, we derived a dynamic virtual bounding-box correction that separates this geometric factor from the translation parameters learned by the network. As a result, the proposed method achieves robust and consistent performance even under severe perspective distortion.

Limitation. Despite its effectiveness, estimating metric 3D translation from a single image still requires inferring physical scale solely from appearance, making depth estimation inherently dependent on the diversity and coverage of the training data.

Broader impact. Nevertheless, the proposed dynamic bounding-box size adjustment strategy could be extended to other human-centric pose estimation tasks, including human body and hand pose estimation, that employ bounding-box-based metric translation formulations. Like other pose estimation technologies, it could potentially be misused for surveillance or privacy-invasive applications.

Acknowledgments and Disclosure of Funding

This work was supported by the Technology Innovation Program (RS-2026-25525549, A development of on-device AI-based multi-sensor fusion intelligent cognitive technology to ensure worker safety in manufacturing logistics processes) funded by the Ministry of Trade, Industry and Resources (MOTIR, Korea); by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. RS-2026-25514284, Development of technology to build a learning dataset for creating emotional AI models); and by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2026-25477414). The authors declare no competing interests.

References

[1] Vitor Albiero, Xingyu Chen, Xi Yin, Guan Pang, and Tal Hassner. img2pose: Face alignment and detection via 6DoF, face pose estimation. In *CVPR*, 2021.

[2] Zhiwen Cao, Zongcheng Chu, Dongfang Liu, and Yingjie Chen. A vector-based representation to enhance head pose estimation. In *WACV*, 2021.

[3] Zhiwen Cao, Dongfang Liu, Qifan Wang, and Yingjie Chen. Towards unbiased label distribution learning for facial pose estimation using anisotropic spherical Gaussian. In *ECCV*, 2022.

[4] José Celestino, Manuel Marques, Jacinto C Nascimento, and João Paulo Costeira. 2D image head pose estimation via latent space regression under occlusion settings. *PR*, 137:109288, 2023.

[5] Sungho Chun and Ju Yong Chang. 6DoF head pose estimation through explicit bidirectional interaction with face geometry. In *ECCV*, 2024.

[6] Sungho Chun, Boeun Kim, Hyung Jin Chang, and Ju Yong Chang. Bidirectional regression for monocular 6DoF head pose estimation and reference system alignment. *PR*, 179:113585, 2026.

[7] Alejandro Cobo, Roberto Valle, José M. Buenaposada, and Luis Baumela. On the representation and methodology for wide and short range head pose estimation. *PR*, 149:110263, 2024.

[8] Yu Deng, Jiaolong Yang, Sicheng Xu, Dong Chen, Yunde Jia, and Xin Tong. Accurate 3D face reconstruction with weakly-supervised learning: From single image to image set. In *CVPRW*, 2019.

[9] Gabriele Fanelli, Matthias Dantone, Juergen Gall, Andrea Fossati, and Luc Van Gool. Random forests for real time 3D face analysis. *IJCV*, 101(3):437–458, 2013.

[10] Yao Feng, Fan Wu, Xiaohu Shao, Yanfeng Wang, and Xi Zhou. Joint 3D face reconstruction and dense alignment with position map regression network. In *ECCV*, 2018.

[11] Jia Guo, Jinke Yu, Alexandros Lattas, and Jiankang Deng. Perspective reconstruction of human faces by joint mesh and landmark regression. In *ECCVW*, 2022.

[12] Jianzhu Guo, Xiangyu Zhu, Yang Yang, Fan Yang, Zhen Lei, and Stan Z Li. Towards fast, accurate and stable 3D dense face alignment. In *ECCV*, 2020.

[13] Xiaoyang Hao and Han Li. PersPose: 3D human pose estimation with perspective encoding and perspective rotation. In *ICCV*, 2025.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[15] Thorsten Hempel, Ahmed A Abdelrahman, and Ayoub Al-Hamadi. 6D rotation representation for unconstrained head pose estimation. In *ICIP*, 2022.

[16] Thorsten Hempel, Ahmed A Abdelrahman, and Ayoub Al-Hamadi. Toward robust and unconstrained full range of rotation head pose estimation. *TIP*, 33:2377–2387, 2024.

[17] Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, Gary Bradski, Kurt Konolige, and Nassir Navab. Model based training, detection and pose estimation of texture-less 3D objects in heavily cluttered scenes. In *ACCV*, 2013.

[18] Heng-Wei Hsu, Tung-Yu Wu, Sheng Wan, Wing Hung Wong, and Chen-Yi Lee. QuatNet: Quaternion-based head pose estimation with multiregression loss. *TMM*, 21(4):1035–1046, 2018.

[19] Yueying Kao, Bowen Pan, Miao Xu, Jiangjing Lyu, Xiangyu Zhu, Yuanzhang Chang, Xiaobo Li, and Zhen Lei. Toward 3D face reconstruction in perspective projection: Estimating 6DoF face pose from monocular image. *TIP*, 32:3080–3091, 2023.

[20] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. EPnP: An accurate O(n) solution to the PnP problem. *IJCV*, 81(2):155–166, 2009.

[21] Xiao Li, Dong Zhang, Ming Li, and Dah-Jye Lee. Accurate head pose estimation using image rectification and a lightweight convolutional neural network. *TMM*, 25:2239–2251, 2022.

[22] Zhihao Li, Jianzhuang Liu, Zhensong Zhang, Songcen Xu, and Youliang Yan. CLIFF: Carrying location information in full frames into human pose and shape estimation. In *ECCV*, 2022.

[23] Hai Liu, Shuai Fang, Zhaoli Zhang, Duantengchuan Li, Ke Lin, and Jiazhang Wang. MFDNet: Collaborative poses perception and matrix Fisher distribution for head pose estimation. *TMM*, 24:2449–2460, 2022.

[24] Hai Liu, Shijia Qian, Tingting Liu, Zelin Cao, Minhong Wang, Jianping Ju, and Zhaoli Zhang. HRHPE: FRoI guides heterogeneous relationship representation learning for precise head pose estimation. *Neurocomput.*, 647:130623, 2025.

[25] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

[26] Tetiana Martyniuk, Orest Kupyn, Yana Kurlyak, Igor Krashenyi, Jiři Matas, and Viktoriia Sharmanska. DAD-3DHeads: A large-scale dense, accurate and diverse dataset for 3D head alignment from a single image. In *CVPR*, 2022.

[27] Kartik Narayan, Vibashan VS, Rama Chellappa, and Vishal M Patel. FaceXFormer: A unified transformer for facial analysis. In *ICCV*, 2025.

[28] Kiru Park, Timothy Patten, and Markus Vincze. Pix2Pose: Pixel-wise coordinate regression of objects for 6D pose estimation. In *ICCV*, 2019.

[29] Aditya Prakash, Arjun Gupta, and Saurabh Gupta. Mitigating perspective distortion-induced shape ambiguity in image crops. In *ECCV*, 2024.

[30] Markus Roth and Dariu M Gavrila. intrApose: Monocular driver 6 DOF head pose estimation leveraging camera intrinsics. *TIV*, 8(8):4057–4068, 2023.

[31] Nataniel Ruiz, Eunji Chong, and James M Rehg. Fine-grained head pose estimation without keypoints. In *CVPRW*, 2018.

[32] Yu Tian, Tianqi Shao, Tsukasa Demizu, Xuyang Wu, and Hsin-Tai Wu. HPE-CogVLM: Advancing vision language models with a head pose grounding task. *TCSVT*, 2026.

[33] Satyajit Tourani, Siddharth Tourani, Arif Mahmood, and Muhammad Haris Khan. Unsupervised discovery of facial landmarks and head pose. In *CVPR*, 2025.

[34] Roberto Valle, José M Buenaposada, and Luis Baumela. Multi-task head pose estimation in-the-wild. *TPAMI*, 43(8):2874–2881, 2020.

[35] Luhui Xu, Zecheng Li, Yanling Gan, and Haiying Xia. Soft-label guided stacked dual attention network for head pose estimation and its application to classroom gaze analysis. *Sci. Rep.*, 16: 405, 2026.

[36] Miao Xu, Xiangyu Zhu, Yueying Kao, Zhiwen Chen, Jiangjing Lyu, and Zhen Lei. Multi-level pixel-wise correspondence learning for 6DoF face pose estimation. *TMM*, 26:9423–9435, 2024.

[37] Tsun-Yi Yang, Yi-Ting Chen, Yen-Yu Lin, and Yung-Yu Chuang. FSA-Net: Learning fine-grained structure aggregation for head pose estimation from a single image. In *CVPR*, 2019.

[38] Frank Yu, Mathieu Salzmann, Pascal Fua, and Helge Rhodin. PCLs: Geometry-aware neural reconstruction of 3D pose with perspective crop layers. In *CVPR*, 2021.

[39] Cheng Zhang, Hai Liu, Yongjian Deng, Bochen Xie, and Youfu Li. TokenHPE: Learning orientation tokens for efficient head pose estimation via transformers. In *CVPR*, 2023.

[40] Zhengyou Zhang. A flexible new technique for camera calibration. *TPAMI*, 22(11):1330–1334, 2000.

[41] Wojciech Zielonka, Timo Bolkart, and Justus Thies. Towards metrical reconstruction of human faces. In *ECCV*, 2022.

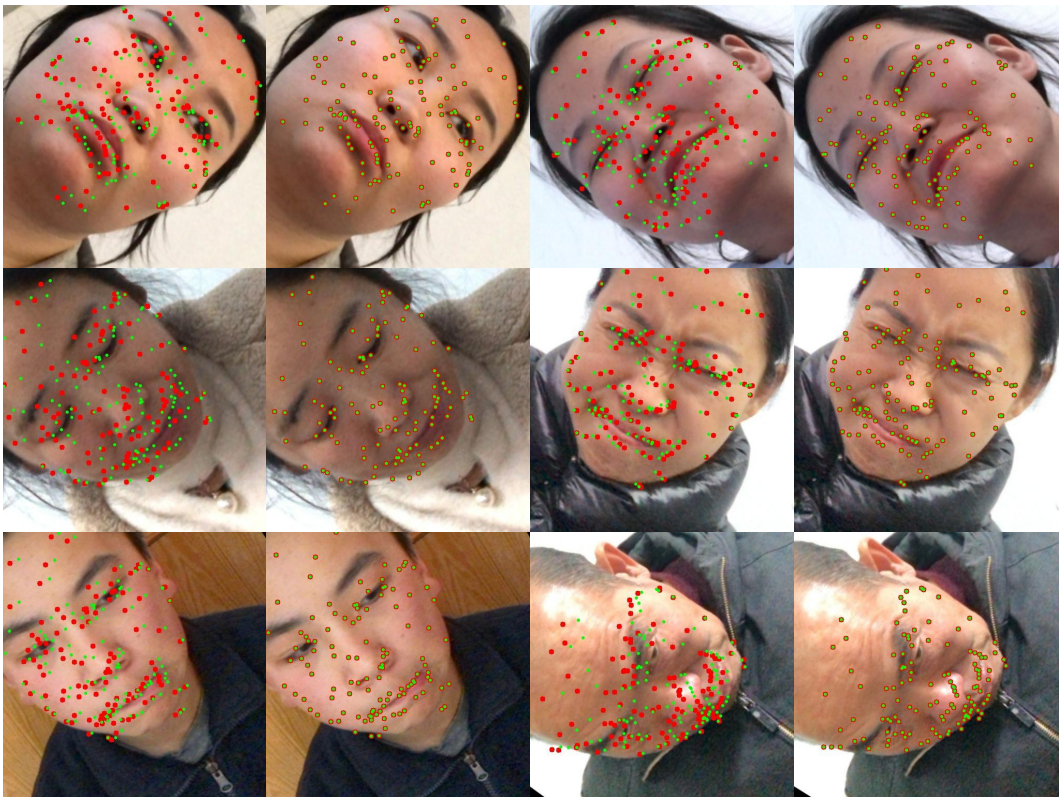


Figure 3: Visualization of the image–label mismatch problem caused by rotation augmentation. Columns 1 and 3 show the conventional rectangular crop method widely used in prior works, while columns 2 and 4 show the proposed virtual camera formulation. The green dots represent the physical pixel trajectories of the face vertices, obtained by projecting the 3D face vertices onto the image plane before rotation augmentation and then applying the same 2D affine transformation as the image rotation. In contrast, the red dots represent the projected face vertices generated using the updated head pose labels after rotation augmentation. Existing models are trained to estimate the labels corresponding to these red dots. Under the conventional crop formulation, the green and red dots become misaligned due to perspective distortion, whereas they remain aligned in the proposed virtual camera space.

A Technical appendices and supplementary material

A.1 Image–Label Mismatch

In this section, we describe the image–label mismatch problem that occurs when the widely used conventional crop method is combined with image rotation augmentation. Fig. 3 visually illustrates this mismatch problem. The green dots in columns 1 and 3 of Fig. 3 represent the projection of the head-space vertices V_h onto the image I_o using R_o^*, T_o^*, and K_o, followed by the application of rotation augmentation together with I_o. Here, $*$ denotes the ground truth. This visualization illustrates how the face pixels move when rotation augmentation is applied.

The red dots in columns 1 and 3 represent the projection of V_h onto the rotation-augmented image I_o^r using $R_o^{*,r} \in SO(3)$, T_o^*, and K_o. Here, $R_o^{*,r}$ denotes the rotation matrix obtained by applying rotation augmentation to R_o^*. Existing models are trained to estimate $R_o^{*,r}$ and T_o^* that generate these red dots. However, as shown in Fig. 3, because the labels no longer align with the image after rotation augmentation, the model is fundamentally trained using inconsistent supervision.

This image–label mismatch occurs because applying rotation augmentation to pixels that have already undergone perspective projection transforms them into positions that cannot be explained by the pinhole camera model. To address this issue, off-axis distortion must first be mitigated before

applying rotation augmentation. For this purpose, we define a virtual camera that directly faces the bounding box ray and warp the original image onto the virtual image plane.

The green dots in columns 2 and 4 of Fig. 3 represent the projection of V_h onto the image I_v using R_v^*, T_v^*, and K_v, followed by rotation augmentation together with I_v, while the red dots represent the projection of V_h onto the rotation-augmented image I_v^r using $R_v^{*,r} \in SO(3)$, T_v^*, and K_v. Here, $R_v^{*,r}$ denotes the rotation matrix obtained by applying rotation augmentation to R_v^*. According to Fig. 3, the red and green dots overlap at the same positions, indicating that the image–label mismatch problem is effectively mitigated under the proposed virtual camera model.

A.2 Loss Functions

In this section, we define the objective function for training the proposed neural network end-to-end. Since our neural network is designed to infer the pose in the virtual camera space, we calculate losses not only for the main outputs of the network, namely the head rotation R_v and 3D translation T_v in the virtual space, but also for the virtual 2D sparse landmarks L_v and 3D face mesh vertices V_h for auxiliary learning [6]. To this end, it is first necessary to transform the ground truth provided in the dataset, which is defined in the original camera coordinate system, into the corresponding ground truth in the proposed virtual space.

A.2.1 Ground Truth in Virtual Camera Space

Let the original head rotation ground truth of the target defined in the dataset be $R_o^* \in SO(3)$, and the original 3D translation ground truth be $T_o^* \in \mathbb{R}^3$. To define the learning objective in the virtual camera coordinate system C_v, the inverse rotation of ${}^oR_v$ is applied to the ground truth in the original space:

$$R_v^* = ({}^oR_v)^\top R_o^*, \quad T_v^* = ({}^oR_v)^\top T_o^*. \tag{14}$$

Furthermore, to calculate the sparse landmark loss, the landmark ground truth L_o^* defined on the original 2D image plane must also be transformed into the virtual space. This transformation is performed by expressing each coordinate of L_o^* in homogeneous coordinates and multiplying it by the homography matrix H, thereby obtaining the landmark ground truth L_v^* on the virtual image plane. This process is formulated as follows:

$$\begin{bmatrix} L_{v,n}^* \\ 1 \end{bmatrix} \sim H \begin{bmatrix} L_{o,n}^* \\ 1 \end{bmatrix}, \tag{15}$$

where $L_{o,n}^* \in \mathbb{R}^2$ and $L_{v,n}^* \in \mathbb{R}^2$ denote the n-th column vectors of L_o^* and L_v^*, respectively.

The 3D face mesh vertices V_h are defined in an independent local head coordinate system (head space) that is unaffected by changes in the camera coordinate system. Therefore, for the mesh vertices, the ground truth V_h^* provided by the existing dataset is used directly without a separate virtual-space transformation process.

A.2.2 Network Loss Functions

Based on the transformed ground truth in the virtual space and the ground truth in the head coordinate system, we define the overall loss function used to optimize the network outputs. First, the core rotation and translation losses are defined using a Frobenius norm loss and an L1 loss, respectively:

$$\mathcal{L}_{\text{rot}} = \|R_v - R_v^*\|_F, \quad \mathcal{L}_{\text{trans}} = \|T_v - T_v^*\|_1. \tag{16}$$

In addition, for auxiliary supervision, a loss $\mathcal{L}_L$ for the virtual 2D sparse landmark error and a loss $\mathcal{L}_{head}$ for the 3D face mesh vertex error are defined as follows:

$$\mathcal{L}_L = \frac{1}{N^L} \sum_{n=1}^{N^L} \|L_{v,n} - L_{v,n}^*\|_1, \quad \mathcal{L}_{\text{head}} = \frac{1}{N^V} \sum_{n=1}^{N^V} \|V_{h,n} - V_{h,n}^*\|_1. \tag{17}$$

Here, $L_{v,n}$ and $V_{h,n}$ denote the n-th column vectors of L_v and V_h, respectively. Consequently, the total objective function is formulated as follows:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{rot}}\mathcal{L}_{\text{rot}} + \lambda_{\text{trans}}\mathcal{L}_{\text{trans}} + \lambda_{\text{head}}\mathcal{L}_{\text{head}} + \lambda_L\mathcal{L}_L. \tag{18}$$

A.3 Implementation Details

In this study, we adopted TRGv2 [6] with a ResNet18 [14] backbone as the base network architecture for the experiments. The number of sampling points used in the projection-and-sample mechanism was set to $18 \times 18 = 324$ at the 1st stage and 305 at the 2nd stage. HOGWARTS was trained end-to-end with a mini-batch size of 128 for a total of 30 epochs on the ARKitFace dataset. For model optimization, we used the AdamW optimizer [25] with a weight decay of 10^{-4}. The momentum parameters β_1 and β_2 were set to 0.9 and 0.999, respectively. The initial learning rates for the encoder and decoder parameters were set to 10^{-4} and decayed by a factor of 10 after 20 epochs. The weights for each term in the final loss function, λ_{head}, λ_{rot}, λ_{trans}, and λ_L, were set to 20, 10, 0.5, and 1.25, respectively. To improve generalization performance during training, data augmentation techniques including random cropping, color jittering, image rotation, and horizontal flipping were applied. All experiments were conducted using a single NVIDIA RTX 3090 GPU, and the total training time was approximately 16 hours.

A.4 Translation Regression Targets under Fixed and Dynamic Bounding Boxes

We now denote the network outputs required to represent the same ground-truth translation T_v^* under the dynamic configuration $b_v = b_o/r$ and the fixed configuration $b_v = b_o$ by ψ^*_{dyn} and ψ^*_{fix}, respectively. The superscript $*$ denotes the parameter values required to represent the ground-truth translation, rather than actual network predictions.

To represent the same depth $T_v^{*,z}$, Eq. 5 requires

$$T_v^{*,z} = \frac{Bs^*_{\text{dyn}} f_v}{b_o/r} = \frac{Bs^*_{\text{fix}} f_v}{b_o}, \tag{19}$$

which gives

$$s^*_{\text{fix}} = r s^*_{\text{dyn}}. \tag{20}$$

Combining Eq. 5 with $\tau^{\text{face}} = (b_v/2)\tilde{\tau}^{\text{face}}$, the x and y translation components can be written as $T_v^x = (Bs/2)\tilde{\tau}^{x,\text{face}}$ and $T_v^y = (Bs/2)\tilde{\tau}^{y,\text{face}}$, respectively. Therefore, representing the same x and y translation components requires the normalized 2D offset vectors to satisfy

$$\frac{Bs^*_{\text{dyn}}}{2}\tilde{\tau}^{\text{face},*}_{\text{dyn}} = \frac{Bs^*_{\text{fix}}}{2}\tilde{\tau}^{\text{face},*}_{\text{fix}}. \tag{21}$$

Substituting the scale relation derived above yields

$$\tilde{\tau}^{\text{face},*}_{\text{fix}} = \frac{1}{r}\tilde{\tau}^{\text{face},*}_{\text{dyn}}. \tag{22}$$

Combining these relations according to the output vector ordering in Eq. 4, we obtain

$$\boxed{\psi^*_{\text{fix}} = \text{diag}\left(r, \frac{1}{r}, \frac{1}{r}\right)\psi^*_{\text{dyn}}}. \tag{23}$$

Thus, relative to the dynamic configuration, the fixed bounding-box configuration incorporates multiplicative factors of r and $1/r$ into the scale and normalized 2D offset targets, respectively. The dynamic bounding-box formulation explicitly accounts for this known, position-dependent factor in translation estimation, thereby avoiding the additional multiplicative dependence on r in the effective regression targets that the network is trained to predict.

A.5 Performance Across Off-Axis Angles

In this section, we evaluate the performance gains of HOGWARTS across off-axis image locations through comparisons with TRGv2 and a fixed virtual bounding-box baseline ($b_v = b_o$). The comparison with TRGv2 evaluates the complete framework, while the comparison with the fixed baseline examines the contribution of dynamic virtual bounding-box sizing. The TRGv2 baseline used in this analysis is the retrained model reported in Table 1.

We evaluate models trained on ARKitFace using the BIWI dataset, following the evaluation protocol described in the main paper. The fixed baseline applies virtual-camera normalization but retains

Table 5: Performance across off-axis angles on BIWI. The $b_v = b_o$ baseline applies virtual-camera normalization while retaining the original bounding-box size. HOGWARTS uses the dynamic formulation $b_v = b_o/r$. All errors are reported in millimeters (mm); lower values are better. For each bin and metric, the best and second-best results are marked in bold and underlined, respectively. Tied results share the same rank.

		MAE_t			Depth error			ADD		
Bin (θ)	# Samples	TRGv2	$b_v = b_o$	HOGWARTS	TRGv2	$b_v = b_o$	HOGWARTS	TRGv2	$b_v = b_o$	HOGWARTS
$0.08^\circ \le \theta < 3.29^\circ$	1,299	<u>11.56</u>	**10.64**	**10.64**	<u>30.28</u>	**27.61**	**27.61**	31.73	**28.98**	<u>29.08</u>
$3.29^\circ \le \theta < 6.51^\circ$	4,895	9.15	**8.62**	<u>8.84</u>	22.43	**20.80**	<u>21.42</u>	24.00	**22.64**	<u>23.28</u>
$6.51^\circ \le \theta < 9.72^\circ$	4,886	<u>11.97</u>	12.66	**11.26**	<u>28.83</u>	30.12	**26.14**	<u>31.03</u>	32.47	**28.70**
$9.72^\circ \le \theta < 12.94^\circ$	2,091	15.40	<u>14.04</u>	**12.36**	35.22	<u>31.91</u>	**27.73**	37.99	<u>34.70</u>	**30.53**
$12.94^\circ \le \theta \le 16.15^\circ$	269	<u>12.56</u>	13.01	**11.56**	<u>27.45</u>	28.41	**23.87**	<u>32.20</u>	33.10	**28.52**

the original bounding-box size, whereas HOGWARTS uses $b_v = \frac{b_o}{r}$, where $r = \|\mathbf{d}_{\text{raw}}\|_2$. Both virtual-camera variants set $f_v = f_o$. For each sample, we compute the off-axis angle θ between the original camera's optical axis and the viewing ray passing through the bounding-box center (u_{oc}, v_{oc}):

$$\theta = \arctan\left(\sqrt{\left(\frac{u_{oc} - c_{ox}}{f_o}\right)^2 + \left(\frac{v_{oc} - c_{oy}}{f_o}\right)^2}\right). \tag{24}$$

Here, (c_{ox}, c_{oy}) denotes the principal point of the original camera. This angle is related to the position-dependent geometric factor by $r = \frac{1}{\cos\theta}$. We divide the observed range, $0.08^\circ \le \theta \le 16.15^\circ$, into five equal-width bins and report MAE_t, depth error, and ADD for each bin. Both comparisons use the same samples and bin boundaries. The relative error reduction against each baseline is computed as

$$\text{Gain } (\%) = \frac{E_{\text{baseline}} - E_{\text{HOGWARTS}}}{E_{\text{baseline}}} \times 100, \tag{25}$$

where E denotes the corresponding error metric. Table 5 reports the results. HOGWARTS achieves lower errors than TRGv2 in all five bins for all three metrics. Although the gains over TRGv2 do not increase monotonically with θ, the relative gains in the two highest-angle bins exceed those in the three lower-angle bins, indicating an overall tendency toward larger gains in more severe off-axis regimes. The largest gains over TRGv2 occur at $9.72^\circ \le \theta < 12.94^\circ$, reaching 19.74%, 21.26%, and 19.66% in MAE_t, depth error, and ADD, respectively. In comparison with the fixed virtual bounding-box baseline, HOGWARTS achieves comparable MAE_t and depth error, with a slightly higher ADD, in the first bin ($0.08^\circ \le \theta < 3.29^\circ$). In the second bin ($3.29^\circ \le \theta < 6.51^\circ$), HOGWARTS has slightly higher errors for all three metrics. For every bin with $\theta \ge 6.51^\circ$, HOGWARTS achieves lower errors for all three metrics. The largest relative gain over this baseline in MAE_t occurs at $9.72^\circ \le \theta < 12.94^\circ$, reaching approximately 11.97%. The largest relative gains in depth error and ADD occur at $12.94^\circ \le \theta \le 16.15^\circ$, reaching approximately 15.98% and 13.84%, respectively.

The comparison with TRGv2 demonstrates that the complete HOGWARTS framework improves performance across the evaluated image locations. The fixed-baseline comparison further indicates that dynamic virtual bounding-box sizing provides additional benefits at larger off-axis angles. This observation is consistent with the geometric analysis in Appendix A.4: retaining the original bounding-box size leaves an r-dependent factor in the effective translation regression targets, whereas the dynamic formulation analytically separates this factor from the parameters learned by the network. Together, these results support the effectiveness of the proposed strategy in off-axis regions within the evaluated angle range.

A.6 Ablation Study on Rotation Augmentation

We evaluate the effect of 2D rotation augmentation on BIWI by comparing TRGv2 and HOGWARTS trained with and without this augmentation. TRGv2 uses conventional rectangular cropping, while HOGWARTS applies virtual-camera normalization. We report MAE_t and ADD, where ADD jointly reflects rotation and translation errors.

Table 6 shows that rotation augmentation increases MAE_t from 11.09 to 11.45 mm and ADD from 28.73 to 29.64 mm for TRGv2. For HOGWARTS, rotation augmentation reduces MAE_t from 11.21 to 10.50 mm and ADD from 29.01 to 27.04 mm.

Table 6: Effect of 2D rotation augmentation on BIWI. All errors are reported in millimeters (mm); lower values are better.

Model	Rotation augmentation	MAE_t	ADD
TRGv2 (retrained)	Without	11.09	28.73
TRGv2 (retrained)	With	11.45	29.64
HOGWARTS	Without	11.21	29.01
HOGWARTS	With	10.50	27.04

Table 7: Sensitivity of HOGWARTS to focal-length errors on BIWI. GE is reported in degrees, while MAE_t and ADD are reported in millimeters. Lower values are better.

Focal-length perturbation	GE	MAE_t	ADD
0%	5.48	10.50	27.04
$+0.3\%$	5.47	10.36	26.69
-0.3%	5.48	10.71	27.60

The differing effects on translation error and ADD are consistent with our analysis that virtual-camera normalization mitigates the image–label mismatch associated with applying 2D rotation augmentation to conventional off-axis crops. In this ablation, HOGWARTS benefits from rotation augmentation in translation accuracy and the joint pose metric.

A.7 Sensitivity to Calibration Errors

HOGWARTS assumes that the camera intrinsic parameters are obtained through calibration before pose estimation. We therefore evaluate sensitivity to small residual focal-length errors in this calibrated setting. Based on the calibration accuracy reported by Zhang [40], we set the focal-length perturbation range to $\pm 0.3\%$ on BIWI.

The trained model parameters are kept fixed, and perturbations are applied only at test time. Specifically, we perturb the focal length used for virtual-camera warping and pinhole-based translation estimation as

$$f' = (1 + \delta)f, \qquad \delta \in \{-0.003, 0, 0.003\}. \tag{26}$$

Table 7 reports the resulting performance. Across the tested range, the maximum absolute changes from the unperturbed result are 0.01° in GE, 0.21 mm in MAE_t, and 0.56 mm in ADD. These results indicate limited sensitivity to the small focal-length perturbations evaluated here.

A.8 Qualitative Results

Figs. 4 and 5 visually demonstrate the performance of HOGWARTS and TRGv2. As shown in the figures, HOGWARTS infers head poses that are closer to the ground truth (GT) compared to TRGv2 on the ARKitFace and BIWI datasets, which exhibit severe perspective distortion. This is attributed to the proposed off-axis distortion mitigation mechanism, which enables more stable model training.

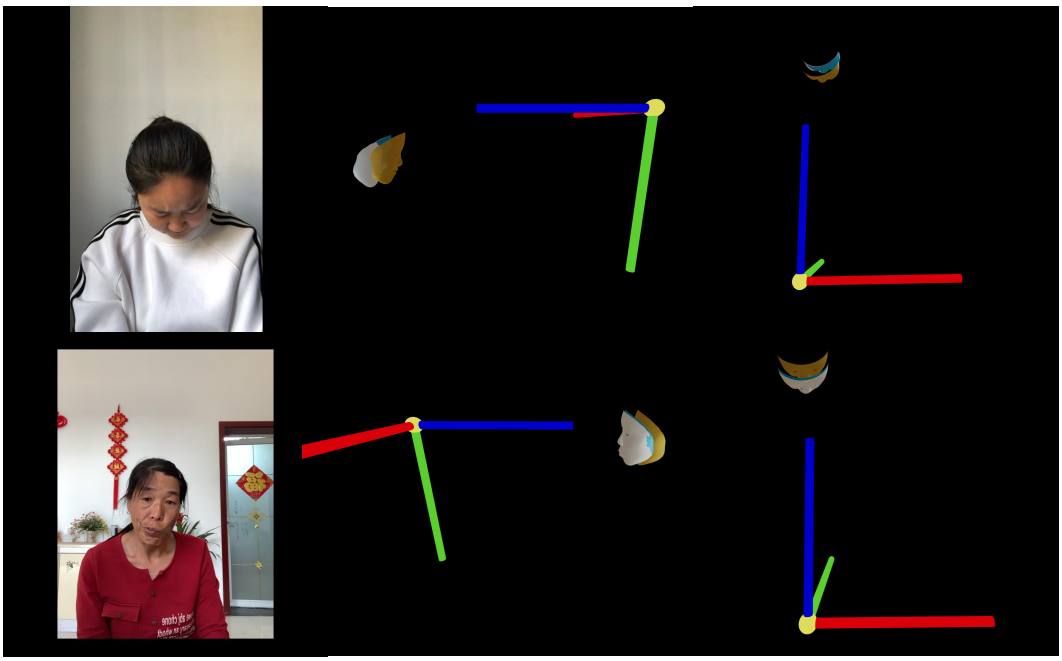


Figure 4: Qualitative results on the ARKitFace dataset. The blue, gold, and white colors visualize the face meshes of HOGWARTS, TRGv2, and GT, respectively.

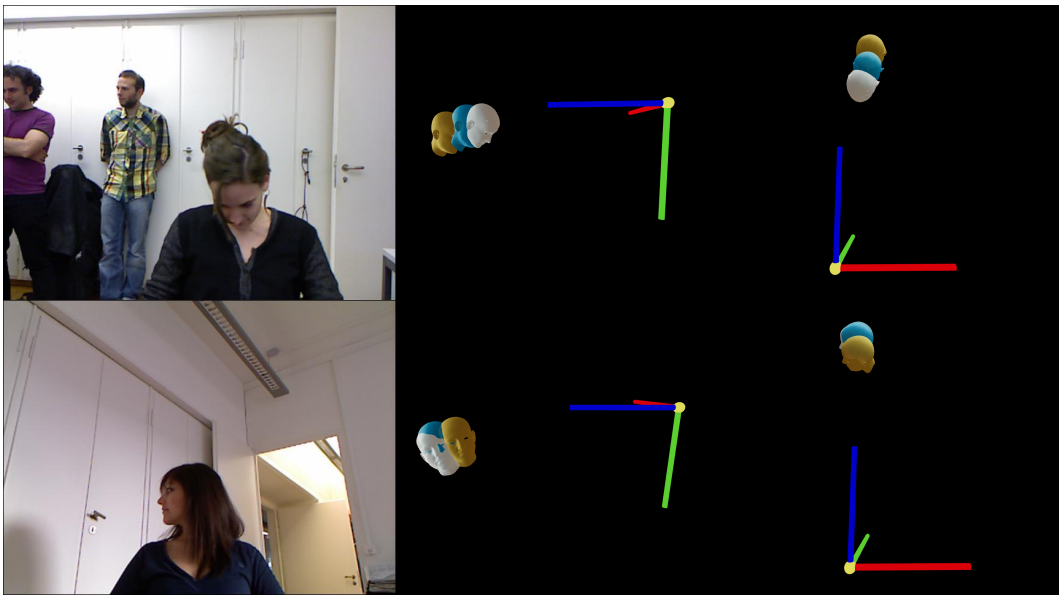


Figure 5: Qualitative results on the BIWI dataset. The blue, gold, and white colors visualize the face meshes of HOGWARTS, TRGv2, and GT, respectively.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract, introduction, and experimental sections clearly describe and support the main contributions of the paper, including the proposed virtual camera formulation, geometric translation modeling, and performance improvements under severe perspective distortion and cross-domain settings.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The limitations of monocular metric translation estimation and its dependence on training data diversity are discussed in Sec. 5.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: The paper does not present formal theoretical theorems or proofs. Instead, it provides geometric formulations and mathematical derivations for the proposed method.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper and appendix provide detailed descriptions of the datasets, network architecture, geometric formulation, implementation settings, training procedures, and evaluation protocols required to reproduce the experiments.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Code is available at `https://github.com/asw91666/HOGWARTS_RELEASE`. The experiments are conducted using publicly available datasets.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper and appendix describe the datasets, input resolution, optimizer, learning schedule, augmentation strategy, GPU environment, and evaluation metrics in detail.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: The experiments report deterministic evaluation metrics without repeated multi-run statistical analysis. Conducting multiple full training runs was computationally expensive.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The appendix specifies the GPU environment, training duration, optimizer configuration, and other implementation details required to estimate the computational cost of the experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: This research complies with the NeurIPS Code of Ethics. The work focuses on geometric modeling for head pose estimation using publicly available datasets.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: Sec. 5 discusses the broader applicability of the proposed geometric formulation as well as potential concerns regarding the misuse of pose estimation technologies.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not introduce high-risk generative models, autonomous systems, or sensitive deployment scenarios requiring additional safeguards.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All datasets, prior methods, and external assets used in this work are properly cited and used in accordance with their respective licenses and terms of use.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.

- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The implementation code is released at `https://github.com/asw91666/HOGWARTS_RELEASE`, together with a README describing environment setup, data preparation, training, and evaluation, and a license file.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing experiments or human subject studies conducted by the authors.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve human subject studies or data collection requiring IRB approval.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs were not used as part of the core methodology, experiments, data generation, or scientific contributions of this work.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.