6DoF Head Pose Estimation through Explicit Bidirectional Interaction with Face Geometry

Supplementary Material

Sungho Chun and Ju Yong Chang

Department of ECE, Kwangwoon University, Korea
{asw9161,jychang}@kw.ac.kr

1 Implementation Details

TRG is trained end-to-end using a mini-batch size of 256 and the number of epochs is set to 30. The Adam optimizer [5] is utilized, starting with an initial learning rate of 10^{-4}, which is decreased by a factor of 10 after 20 epochs. During the training of TRG, we employ augmentation techniques such as random cropping, resizing, color jittering, mask patch augmentation, image rotation, and horizontal flip augmentation on the training images. When training the TRG with multiple datasets, we utilized only the 2D sparse landmarks from the 300W-LP [8]. The face mesh and head rotation labels from the 300W-LP were not used in the training process. The training process is completed in approximately 16 hours on a single RTX 3090 GPU.

2 Calculation of Head Translation from Correction Parameters

Fig. I illustrates the method for calculating head translation T_t from correction parameters c_t and bounding box information I_{bbox}. Through Fig. I, we can derive Eqs. I, II, III, and IV:

$$\frac{T_t^z}{f} = \frac{0.2s_t}{b} \iff T_t^z = \frac{0.2s_t}{b} f, \tag{I}$$

$$\frac{\tau_t^{x,\text{face}}}{b} = \frac{T_t^{x,\text{face}}}{0.2s_t} \iff T_t^{x,\text{face}} = \frac{0.2s_t}{b} \tau_t^{x,\text{face}}, \tag{II}$$

$$\frac{\tau^{x,\text{bbox}}}{f} = \frac{T^{x,\text{bbox}}}{T_t^z}, \tag{III}$$

$$T_t^x = T^{x,\text{bbox}} + T_t^{x,\text{face}}. \tag{IV}$$

By substituting Eq. I into Eq. III, the following can be obtained:

$$T^{x,\text{bbox}} = \frac{0.2s_t}{b} \tau^{x,\text{bbox}}. \tag{V}$$

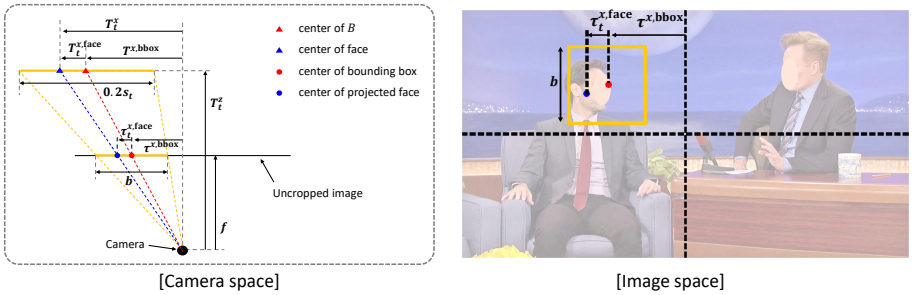

Fig. I: Calculation of head translation T_t from correction parameters c_t and bounding box information I_{bbox}. Best viewed in color.

Table I: Comparison with previous methods for face size estimation on ARKitFace test dataset. The unit is mm^2.

Method	Face Size Error ↓
JMLR [3] † ⋆	937.92
PerspNet [4]	768.58
TRG (Ours)	713.95
TRG (Ours) ⋆	**706.86**

By inserting Eqs. II and V into Eq. IV, the following is obtained:

$$T_t^x = \frac{0.2s_t}{b}\tau^{x,\text{bbox}} + \frac{0.2s_t}{b}\tau_t^{x,\text{face}}, \tag{VI}$$

where $\tau_t^{x,\text{face}}$ represents the x-axis image coordinate of the head center relative to the bounding box center. The normalized value, $\tilde{\tau}_t^{x,\text{face}}$, is obtained by dividing $\tau_t^{x,\text{face}}$ by b, indicating that $\tau_t^{x,\text{face}} = b\tilde{\tau}_t^{x,\text{face}}$. Substituting $b\tilde{\tau}_t^{x,\text{face}}$ for $\tau_t^{x,\text{face}}$ in Eq. VI leads to Eq. 2, as discussed in the main paper. The calculation of T_t^y follows the identical procedure used for T_t^x.

3 Comparison with Existing Methods for Estimating Face Size

This section demonstrates that the depth-aware landmark prediction architecture of TRG is effective in inferring face size. For the experiment, we measured the face size error (in mm^2) of JMLR [3], PerspNet [4], and TRG using ARKitFace test data. The ARKitFace test data includes a variety of face appearances and head sizes for subjects ranging in age from 9 to 60 years. We defined face size as the sum of the areas of all triangles belonging to the face geometry and calculated the MAE between the GT and the prediction for face size error.

Table I shows the face size error for TRG and existing models. According to Table I, TRG significantly outperforms optimization-based methods [3, 4].

Table II: Comparison with existing method for translation estimation.

Method	ARKitFace				BIWI		
	Mean ↓	MAE_r ↓	MAE_t ↓	ADD ↓	MAE_r ↓	MAE_t ↓	ADD ↓
Local-to-global	1.61	**0.89**	3.64	8.71	6.02	27.00	62.91
Camera space offset	1.60	0.90	3.63	8.72	2.76	13.82	31.25
TRG (Ours)	**1.58**	0.91	**3.62**	**8.68**	**2.75**	**12.97**	**29.46**

This result demonstrates the superiority of the depth-aware landmark prediction architecture that utilizes head depth information in the face size inference process.

4 Comparison between TRG with a Smaller Amount of Training Data and Head Rotation Estimator

In this section, we demonstrate that the high accuracy of TRG in head rotation estimation on BIWI [2] is not simply due to the use of a large amount of training data [4]. Existing models for estimating head rotation are trained on the 300W-LP dataset [8] and their performance is evaluated on the BIWI dataset. However, TRG is trained on the ARKitFace training dataset, which contains approximately 5.9 times more data frames than the 300W-LP dataset. For our experiment, we sample the ARKitFace training data at a 1/10 rate to train 'TRG (1/10)' and compare its performance with existing head rotation estimators. Note that TRG (1/10) is trained using approximately 0.58 times fewer data frames and about 10 times fewer subjects than the head rotation estimators. The performance of TRG (1/10) on BIWI still surpasses other head rotation estimators: $MAE_r = 3.07$, $MAE_t = 13.91$, ADD = 32.00, GE = 6.08. The performance of the head rotation estimators evaluated on the BIWI dataset is shown in Table 4 of the main manuscript. This result supports our claim that the superior head rotation estimation performance of TRG is due to the landmark-to-image alignment framework rather than the amount of training data.

5 Comparison with Existing Methods for Translation Estimation

Comparison with img2pose [1]. We compare the 'local-to-global' method utilized in img2pose for estimating head translation to our method, which estimates bounding box correction parameters. The 'local-to-global' method involves directly inferring the 6DoF local head pose from cropped image features, then converting this local head pose into a global head pose. The term 'global head pose' refers to the head pose as it would appear in uncropped images, essentially the 6DoF head pose defined in camera space. For our experiments, we developed a baseline model employing the local-to-global approach. This allows us to directly compare its performance against that of TRG. The primary difference

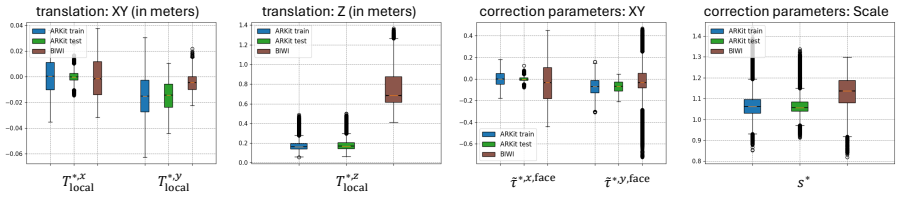

Fig. II: The distribution of ground-truth local head translation [1] and correction parameters in ARKitFace and BIWI. The first and second columns visualize the distribution of local head translation. The third and fourth columns visualize the distribution of the correction parameters. The colors blue, green, and brown represent the distributions of the ARKitFace training data, ARKitFace test data, and BIWI dataset, respectively. The symbol $*$ denotes ground-truth. Best viewed in color.

between the two models lies in their respective methodologies for estimating the 6DoF head pose. Apart from this, all other aspects of the models remain unchanged.

According to Table II, the local-to-global baseline exhibits high head pose estimation performance on the ARKitFace test dataset [4], which has a distribution similar to the training data. However, when evaluated across different dataset, specifically on the BIWI dataset [2], the local-to-global baseline significantly lags behind TRG in terms of head pose estimation accuracy. This underperformance is noteworthy in the context of cross-dataset evaluation.

The local-to-global baseline method shifts the focus from estimating global head translation to estimating local head translation. Despite this shift in focus, the disparity in the distribution of the z-axis direction between out-of-distribution dataset and the training dataset remains markedly evident. This disparity is visually demonstrated in Fig. II. According to Fig. II, the local-to-global baseline, trained on the ARKitFace training dataset, must effectively extrapolate head translation in the z-axis direction to generalize to the BIWI dataset. However, this poses a significant challenge for learning-based models.

As discussed in the main paper, specifically in 'Section 4.4 - Use of correction parameter', our approach involves shifting the estimation target from head translation to correction parameters. This strategic change significantly boosts the model's ability to generalize. The effectiveness of this strategic redefinition is quantitatively demonstrated in Table II, providing solid proof of our method's superiority.

Comparison with CLIFF [6]. In this paper, we elucidate the technical distinctions between the bounding box correction parameter estimation method and CLIFF's method of estimating camera space offsets. The latter involves estimating the x- and y-axis offsets of a target defined in 3D camera space. In contrast, our approach technically differs by estimating the x- and y-axis offsets in image space, rather than in camera space.

For experimental purposes, we designed a baseline model focused on camera space offset. The primary distinction between this baseline and TRG is the

Table III: Ablation study for using bounding box information.

Method	ARKitFace				BIWI		
	Mean ↓	MAE_r ↓	MAE_t ↓	ADD ↓	MAE_r ↓	MAE_t ↓	ADD ↓
w/o I_{bbox}	1.60	0.92	3.70	8.89	2.70	14.13	32.51
w/o b	**1.58**	**0.89**	3.66	8.82	**2.67**	14.19	32.79
TRG (Ours)	**1.58**	0.91	**3.62**	**8.68**	2.75	**12.97**	**29.46**

Table IV: Ablation study on the loss functions. The models are evaluated on the BIWI dataset.

Method	MAE_r	GE	MAE_t	ADD
w/o $\mathcal{L}_{\text{rot}}$	3.02	5.91	14.50	33.84
w/o $\mathcal{L}_{\text{cam}}$	3.01	5.85	14.92	35.17
w/o $\mathcal{L}_L$	3.21	6.20	14.80	33.40
TRG (Ours)	**2.75**	**5.35**	**12.97**	**29.46**

method used to estimate head translation, while all other aspects of the models are identical.

Table II shows that TRG slightly outperforms the camera space offset baseline in the ARKitFace test dataset. However, in cross-dataset evaluations with the BIWI dataset, TRG demonstrates a significant advantage in head translation estimation accuracy.

Our method employs a geometrical approach to estimate the x- and y-axis offsets $\tilde{\tau}_t^{x,\text{face}}, \tilde{\tau}_t^{y,\text{face}}$ in image space, which are then used to compute 3D head translation through inverse projection transformation. This geometrical method ensures stable and consistent estimation even across datasets with distributions different from the training data. Conversely, the method used by CLIFF bypasses geometrical transformations and instead directly estimates 3D space offsets using learnable layers. This estimation process, being highly non-linear, may not exhibit stability, especially with out-of-distribution data.

The results of our experiments underscore the excellence of integrating learning-based methods with geometrical transformations through the correction parameter approach. This not only underscores our method's superiority in managing complex, real-world scenarios but also sets the stage for future enhancements in accurate 6DoF head pose estimation.

6 Ablation Experiments

Utilizing bounding box information. We present experimental evidence explaining the rationale behind using bounding box information I_{bbox}, as input for the face regressor. Specifically, we explore the impact of incorporating bounding box size information on estimating head translation.

In our experiments, Table III presents the results of three models: one excluding all bounding box information (w/o I_{bbox}), one excluding bounding box size information (w/o b), and the TRG.

Table V: Comparison with previous methods for 6DoF head pose estimation on ARKitFace test dataset. Models trained with multiple datasets are marked with the symbol ⋆, and retrained model is indicated by the symbol †.

Method	Yaw	Pitch	Roll	MAE_r	GE	t_x	t_y	t_z	MAE_t	ADD
img2pose [1, 4]	5.07	7.32	4.25	5.55	-	1.39	3.72	15.95	7.02	20.54
Direct 6DoF Regress [4]	1.86	2.72	1.03	1.87	-	2.80	5.23	19.16	9.06	21.39
Refined Pix2Pose [4, 7]	1.95	2.62	2.48	2.35	-	2.43	4.23	35.33	14.00	36.44
JMLR [3] † ⋆	1.13	1.75	0.61	1.16	2.39	0.98	2.48	11.13	4.86	11.87
PerspNet [4]	0.98	1.43	**0.55**	0.99	1.81	1.00	2.41	9.73	4.38	10.30
TRG (Ours)	0.89	1.30	0.57	0.92	**1.80**	0.83	**1.88**	8.22	3.64	8.74
TRG (Ours) ⋆	**0.88**	**1.29**	0.57	**0.91**	1.84	**0.81**	1.90	**8.17**	**3.62**	**8.68**

According to the results from the ARKitFace test dataset, TRG shows a marginal improvement in head translation accuracy compared to the models that do not incorporate bounding box size information. However, the benefits of using bounding box size information become more apparent in cross-dataset evaluations.

These experimental results suggest that bounding box size information plays a critical role in estimating head translation, primarily due to its strong correlation with the camera-to-face distance. Based on these findings, we advocate for incorporating bounding box information into head pose estimation strategies.

Loss functions. We conduct an ablation study to investigate the influence of each loss function. We evaluate the model's performance on the BIWI dataset when $\mathcal{L}_L, \mathcal{L}_{\text{cam}}$, and $\mathcal{L}_{\text{rot}}$ are individually excluded. Table IV presents the ablation study results for the loss functions. The results indicate that excluding either the rotation loss or the translation loss leads to a performance degradation. Additionally, omitting the sparse 2D landmark loss $\mathcal{L}_L$ significantly increases the head rotation and translation error of TRG. This is because the sparse 2D landmark loss substantially contributes to the quality of the multi-scale feature map generated by the feature extractor.

7 Detailed Quantitative Results on ARKitFace Test Data

For the benefit of our readers' research, we provide results detailing the performance of existing models and TRG on the ARKitFace test data. Table V shows a detailed comparison of head rotation error and translation error between TRG and existing 6DoF head pose estimation methods.

8 Qualitative Results

We present additional qualitative results of JMLR [3], PerspNet [4], and TRG, not included in the main manuscript due to space limitations. Figs. III and IV display the results from the ARKitFace test data and BIWI dataset, respectively. Moreover, to illustrate our proposed method's effectiveness in real-world conditions, we provide further qualitative results for images sourced from the internet

in Figs. V and VI. Figs. V and VI show results inferred without knowledge of camera intrinsics, where the focal length was simply determined by the sum of the image's width and height. Our method shows reasonable performance even without precise knowledge of the camera intrinsic. Also, please see the attached supplementary videos, which include the results for image sequences.

References

1. Albiero, V., Chen, X., Yin, X., Pang, G., Hassner, T.: img2pose: Face alignment and detection via 6dof, face pose estimation. In: CVPR (2021)
2. Fanelli, G., Dantone, M., Gall, J., Fossati, A., Van Gool, L.: Random forests for real time 3d face analysis. IJCV **101**, 437–458 (2013)
3. Guo, J., Yu, J., Lattas, A., Deng, J.: Perspective reconstruction of human faces by joint mesh and landmark regression. In: ECCVW (2022)
4. Kao, Y., Pan, B., Xu, M., Lyu, J., Zhu, X., Chang, Y., Li, X., Lei, Z.: Toward 3d face reconstruction in perspective projection: Estimating 6dof face pose from monocular image. IEEE TIP **32**, 3080–3091 (2023)
5. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015)
6. Li, Z., Liu, J., Zhang, Z., Xu, S., Yan, Y.: Cliff: Carrying location information in full frames into human pose and shape estimation. In: ECCV (2022)
7. Park, K., Patten, T., Vincze, M.: Pix2pose: Pixel-wise coordinate regression of objects for 6d pose estimation. In: ICCV (2019)
8. Zhu, X., Lei, Z., Liu, X., Shi, H., Li, S.Z.: Face alignment across large poses: A 3d solution. In: CVPR (2016)

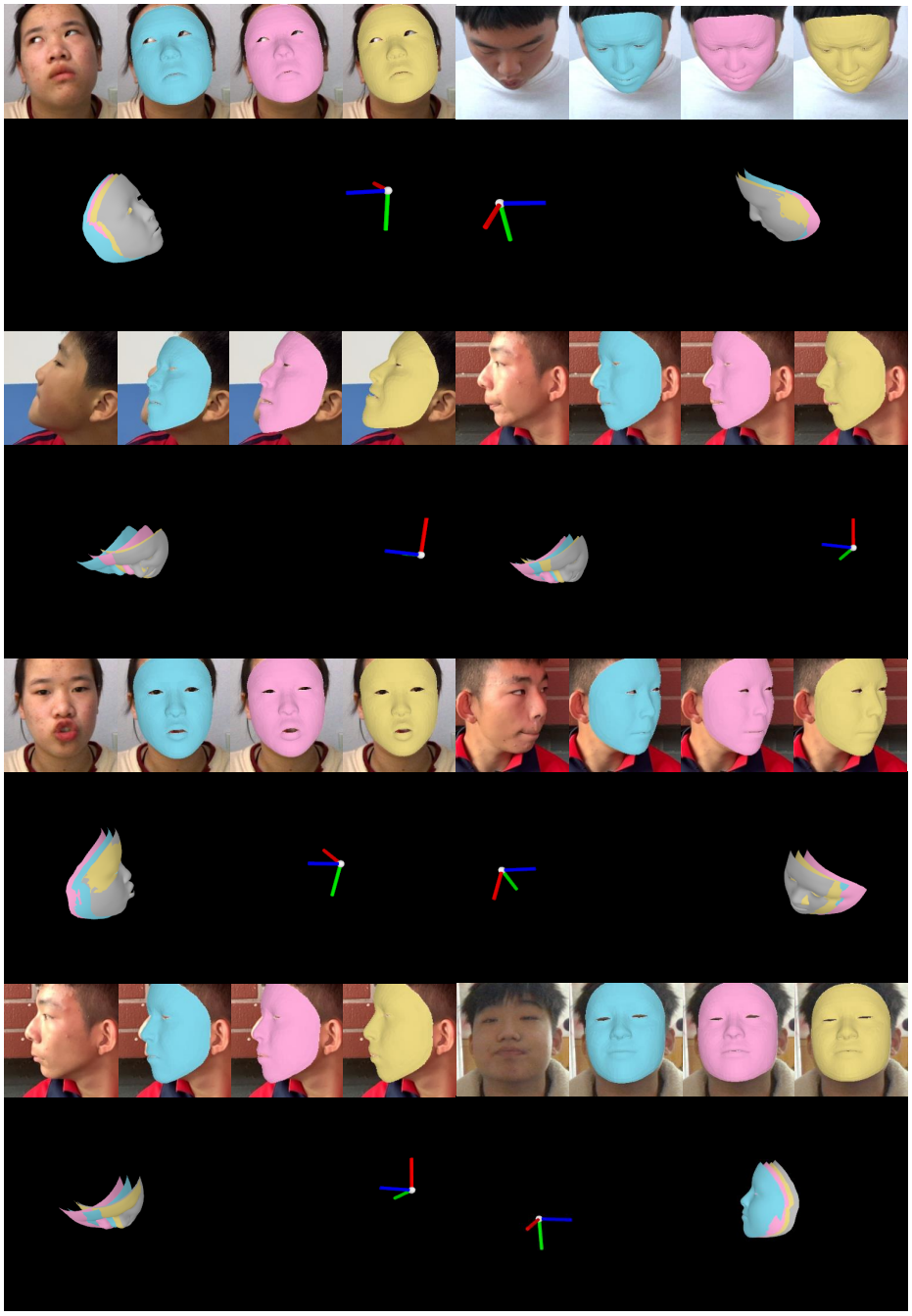

Fig. III: Qualitative comparison on ARKitFace test dataset. The colors cyan, pink, gold, and gray represent JMLR, PerspNet, TRG, and ground truth, respectively. The red, green, and blue axes respectively represent the X, Y, and Z axes of the camera coordinate system. Best viewed in color.

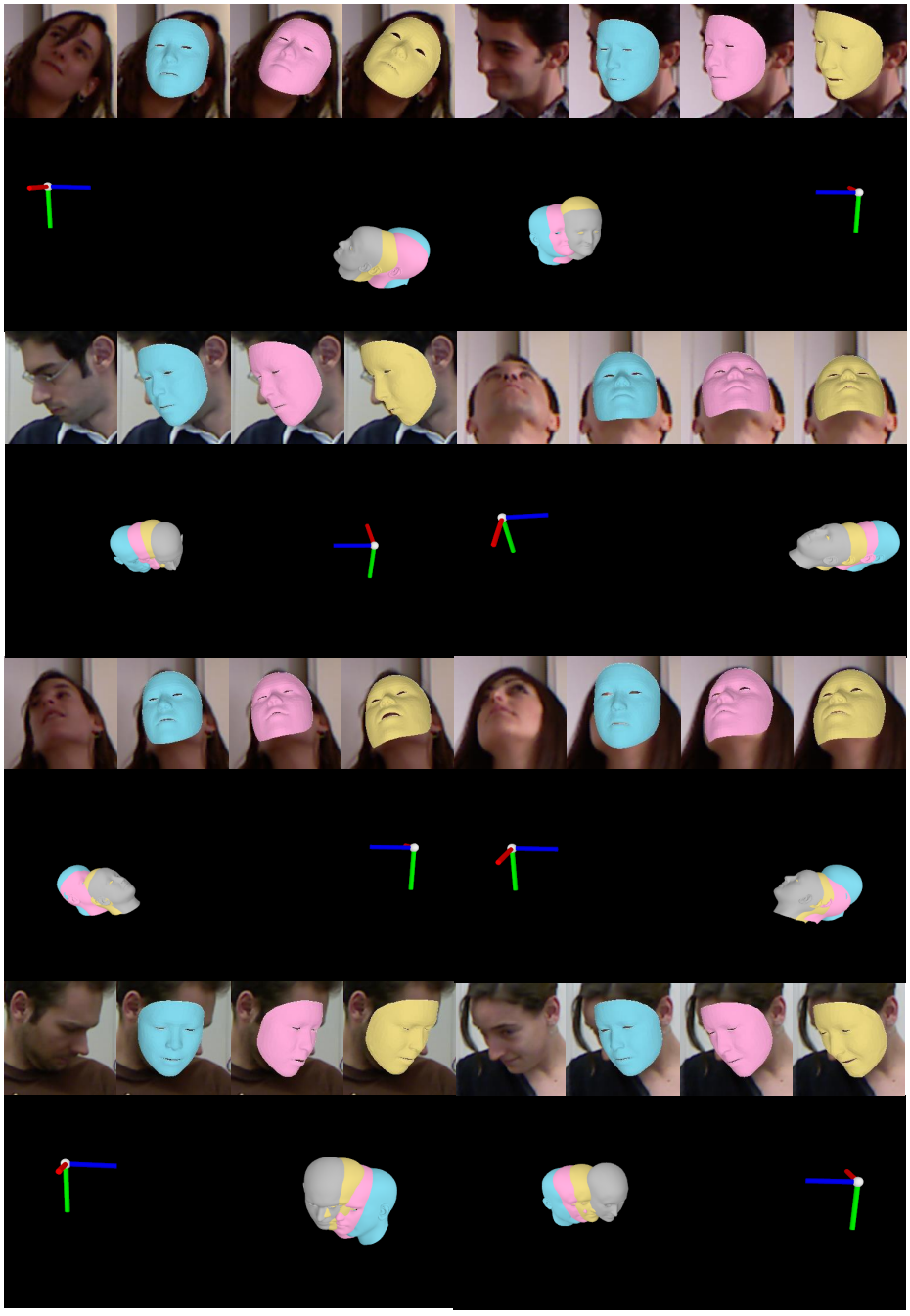

Fig. IV: Qualitative comparison on BIWI dataset. The colors cyan, pink, gold, and gray represent JMLR, PerspNet, TRG, and ground truth, respectively. The red, green, and blue axes respectively represent the X, Y, and Z axes of the camera coordinate system. Best viewed in color.

Fig. V: Qualitative results of TRG on in-the-wild data.

Fig. VI: Qualitative results of TRG on in-the-wild data. For each case, the left side shows the face rendered on the image, while the right side shows the face rendered in camera space. In camera space, the blue and red axes represent the Z- and X-axes, respectively. Best viewed in color.