Unsupervised Discovery of Facial Landmarks and Head Pose

Supplementary Material

7. Supplementary Contents

In this supplementary document, we provide the following information:

1. Ablation Study on Semantic Guided Landmark Localization (SGLL) (Sec. 8).
2. Analysis of training time and training data used (Sec. 9).
3. Cumulative error distribution (CED) curves for Cat Heads and AFLW datasets (Sec. 12).
4. Additional qualitative comparison for unsupervised landmark detection and visual comparison for head pose estimation (Sec. 13).
5. Details on datasets and implementation, including pseudo-codes (Sec. 14).
6. Qualitative and quantitative results on the AnimalWeb dataset (Sec. 10).
7. Ablation with SD (*Stable-diffusion*) variants (Sec. 11).

8. Ablation Study on Semantic Guided Landmark Localization (SGLL)

Table 4 provides ablation study on two novel components of Semantic Guided Landmark Localization (SGLL). First, we remove the Self-Attention by pixel grouping (*w/o Self-Attention*), and second, we remove the cycle-consistency check and instead pick K landmarks at random (*w/o Cycle-Consistency*). Results show that both cycle-consistency checks and self-attention contribute to the overall performance of our approach, and the former has a relatively greater impact.

Method	F. NME	B. NME
Ours	2.76	3.54
w/o Self-Attention	2.81	3.58
w/o Cycle-Consistency	2.84	3.62

Table 4. Ablation analysis on two components of SGLL (smaller error is better).

9. Training Time and Training Data Usage

The analyses in Table 5 shows that on the average less training time and reduced training data is used compared to the existing methods. Particularly, compared to the existing best performing method [54], our approach provides a significant training speed-up while relying on a relatively smaller fraction of training data.

10. Results on AnimalWeb dataset

To validate our method beyond just human-faces we also show qualitative and quantitative results on the AnimalWeb dataset [22] in Figure 12 and Table 6 respectively.

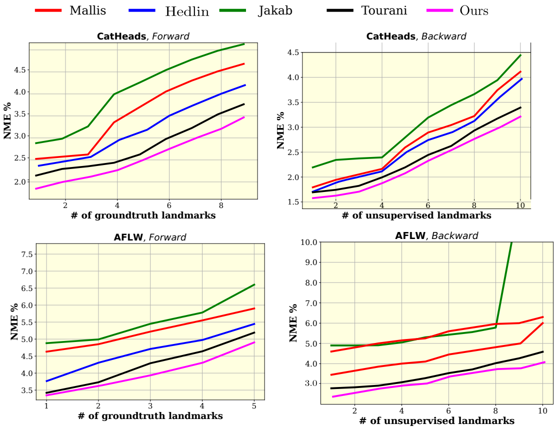


Figure 11. Cumulative Error Distribution (CED) Curves in forward and backward NME for CatHeads and AFLW datasets.

Method	Average Training Time	Training Data Used			
		MAFL	AFLW	LS3D	CatHeads
Hedlin et al. [11]	∼ 1hr 40 mins	29%	52%	93%	84%
Tourani et al. [54]	∼ 8 hours	100 %	100 %	100 %	100 %
Mallis et al. [38]	∼ 6 hours	100%	100%	100%	100%
Ours	∼ 2 hours	27%	49%	87%	76%

Table 5. Comparison of average training time and the fraction of training data used by different methods. Results are averaged over 3 runs.

NME	Method	Alp	Don	Goat	Gir	Monk	Leop	Lemur	Ringt	Seal	Wal	Overall
F	Ours	6.23	4.87	4.84	6.23	4.77	6.23	6.78	6.45	3.45	4.12	**5.56**
	[50]	6.69	5.23	5.76	5.98	5.25	6.19	6.84	6.79	3.79	4.06	6.04
B	Ours	7.87	6.25	6.03	7.89	6.08	7.51	4.78	7.94	4.25	5.23	**6.89**
	[50]	8.69	7.72	6.65	8.31	6.57	8.91	6.84	6.79	5.79	6.06	7.44

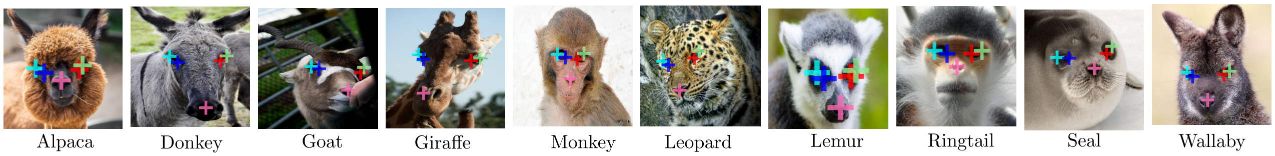
Table 6. Facial landmark comparison on AnimalWeb [22] dataset for ten different species. The overall error is average on all 309 species.

11. Ablation with different SD variants

Table 7 shows the comparisons between different SD (*Stable-diffusion*) variants. Performance using SD-XL and SD 2.1 is slightly better than SD 1.5 used in the main paper.

Datasets	NME	SD-1.2	SD-1.3	SD-XL	SD-1.5 (default)	SD-2.1
AFLW	F	2.88	2.78	2.73	2.76	2.68
	B	3.72	3.56	3.55	3.54	3.48
LS3D	F	2.16	2.10	2.06	2.08	2.03
	B	2.78	2.75	2.72	2.73	2.69

Table 7. Error comparisons across different SD Variants.



Alpaca Donkey Goat Giraffe Monkey Leopard Lemur Ringtail Seal Wallaby

Figure 12. Results on 10 diverse animal species from AnimalWeb dataset. Landmarks are shown as detected by our proposed method for eye corners and nose.

12. Additional CED Curves

Figure 11 shows the cumulative error distribution (CED) curves for the CatHeads and AFLW datasets. Our method shows a consistently lower increase in forward and backward errors with an increasing number of ground truth landmarks than all compared unsupervised landmark detection methods.

13. Qualitative Results

We show additional qualitative results for unsupervised landmark detection on CatHeads (Figure 13), AFLW (Figure 14), MAFL (Figure 15), and LS3D (Figure 16) datasets. We compare with Hedlin et al. [11], Mallis et al. [38], and Tourani et al. [54]. Existing methods struggle with facial images containing smooth regions and/or extreme rotations, while our method can detect semantically relevant landmarks on all such challenging images.

Figure 17 displays visual comparison in the unsupervised head pose estimation results. Compared to [15] and [40], our approach is more robust to extreme facial orientations when detecting head pose.

14. Dataset and Implementation Details

For all datasets, we use the train and test splits, preprocessing, and evaluation protocols, from [38] for a fair comparison. See the pseudo-codes for our overall approach, semantic-guided landmark localization (SGLL), and rotate-and-render augmentation in Algorithm 1, Algorithm 2, and Algorithm 3, respectively.

14.1. Datasets for Landmark Estimation

For landmark estimation, we perform experiments on five different datasets: AFLW [25], LS3D [2], CatHeads [66], MAFL [34], and 300VW dataset. We use the same data set split and evaluation protocol as used in [38].

MAFL: MAFL is a subset of the CelebA [34] dataset and it offers manually annotated facial landmark locations for 19,000 training images and 1,000 test images.

AFLW: The AFLW dataset comprises 10,112 training images and 2,991 test images, where each image is annotated with 21 landmarks.

Cat Heads: This dataset has 9000 images of cat heads annotated with 9 different landmarks. We used 7747 images for training and 1257 images for testing.

LS3D: The LS3D dataset offers images of faces with large pose variations. It is developed by re-annotating the images from the following datasets: 300W-LP [71], AFLW [25], 300VW [48], 300W [46], and FDDB [16] in a consistent manner with 68 points. Each image in the LS3D dataset is annotated with 3D points. Evaluation is performed on the balanced LS3D-W test set, comprising 7200 images, including an equal number of images for each of the range of yaw angles $0° - 30°$, $30° - 60°$, $60° - 90°$.

300-VW: is a facial landmark tracking dataset [48] to evaluate landmark tracking algorithms in the wild. The duration of each video is ~ 1 minute (at 25-30 fps). All frames are annotated with 68 landmarks configuration and this markup is consistent with the one used in 300W competition. The dataset offers 114 videos categorized into three subsets A, B, C in the order of increasing difficulty/challenges.

14.2. Datasets for Head Pose Estimation

BIWI: The BIWI [7] dataset includes 15,678 images that were captured in a lab environment with 20 participants. The head occupies only a small area in the images.

AFLW2000: The AFLW2000 [70] dataset offers the first 2,000 images from the ALFW dataset. These 2K images are annotated with the ground-truth 3D faces and the corresponding 68 landmarks. The images in this dataset display large variations, including different illumination conditions and occlusions.

14.3. Implementation Details

Adapter Γ architecture: Γ consists of a sequence of transformer blocks. The input to gamma is $\mathbf{F} \in \mathbb{R}^{150 \times 768}$ and 150 is the number of tokens. The architecture of Γ consists of 2 transformer blocks. Each block is a $\{$`self`, `cross`, `self`$\}$-attention sequence. Cross attention is performed between latent $\mathbf{z}$ and unified embedding $\mathbf{F}$ processed by self-attention. We write the flow of information as follows:

$$F_s = \texttt{SA}(\mathbf{F}) \tag{11}$$

$$G_c = \texttt{CA}(\mathbf{F}_s, \mathbf{z}) \tag{12}$$

$$H_s = \texttt{SA}(G_c) \tag{13}$$

$$A_s = MLP(H_s) \tag{14}$$

Here, `SA` is the self-attention operation. `CA` is the cross-attention operation. The identical flow of operations is repeated for the next set of blocks. Additional details can be found in the codebase.

Algorithm 1 Main Training Loop

```
Input: Dataset images $\{\mathbf{I}\}$, $\mathcal{P}$ set of consistent landmarks, $\mathbf{I}_o$ origin image (part of dataset)
  Initialization
  1. Learn the initial unified embedding $\mathbf{F}$ by minimizing Eq. (3).
  2. Estimate the initial poses via Eq. (10). Split into pose buckets.
  3. Store the landmarks for each of the images with their poses.
  4. Add $\Gamma$ adapter to SD network.
  for epoch in 1 to N do
      Landmark Learning: Minimize Eq. (3) to train the network for the pose batched image-landmark pairs ($\mathbf{I}$, $\mathbf{C}$).
      Pose Estimation Given the updated landmarks, update the poses via Eq. (10) w.r.t. $\mathbf{I}_o$.
      Perform Rotate-and-Render Augmentation to create new batch of images.
  end for
```

Algorithm 2 Semantic Guided Landmark Localization (SGLL)

```
Input: $\mathcal{I} = \{\mathbf{I}_j \mid j \in \text{images}\}$ and $\mathcal{Y} = \{$“eye”, “nose”, $\ldots\}$ the set of prompts.
  1. $\mathbf{p}_i = \arg\max_{\mathbf{q}\in\mathbb{R}^2} \mathcal{C}(\tau_\theta(\mathbf{y}_i), \mathbf{I})[\mathbf{q}]$        ▷ Extract landmarks from CA maps for image with prompt $\mathbf{y}_i$.
  2. Repeat process by masking out square regions around previous landmarks.
  3. $\mathcal{P}_{\mathbf{I}} = \{\mathbf{q}_i = \arg\max_{\mathbf{q}\in\mathbb{R}^2} \mathcal{S}[\mathbf{p}, \mathbf{q}] \mid \mathbf{p} \in \text{landmarks from } \mathbf{I}\}$.        ▷ Get the landmarks for $\mathbf{I}$ by querying SA map.
  4. Pick a random pair of images and perform cycle-consistency checks. Do so for a random pairs for $3 \times |\mathcal{I}|$ iterations
  5. Pick $K$ landmarks that passed cycle-consistency check.
Output: Set of consistent landmarks.
```

Algorithm 3 Rotate-and-Render Augmentation

```
Input: $\{(\mathbf{I}_i, \mathbf{C}_i) \mid i \in \text{dataset image}\}$        ▷ Front-facing image and landmark pairs.
  Notation $\Pi(.,.)$ orthographic projection. $\odot$ is matrix multiplication. $\mathcal{R}$ is the rendering function.
  for each image $\mathbf{I}'$ and landmark $\mathbf{C}'$ do
      a. Compute the image point cloud via orthographic projection. i.e. $\mathbf{I}'_{pc} = \Pi(\mathbf{I}', \mathbf{D}')$
      b. Compute the landmark point cloud via orthographic projection., i.e. $\mathbf{L}'_{pc} = \Pi(\mathbf{C}', \mathbf{D}')$ .
      c. $\mathbf{I}_R = \mathcal{R}(\mathbf{R}_\Delta, \mathbf{I}'_{pc})$        ▷ Apply rotation to image point cloud and render.
      d. $\mathbf{L}_R = \mathcal{R}(\mathbf{R}_\Delta, \mathbf{L}'_{pc})$        ▷ Apply rotation to landmark point cloud and projection via orthographic-projection.
      e. Compute pose for rotated image point cloud $\mathbf{R}_\Delta \odot \mathbf{I}'_{pc}$ w.r.t. origin image.
  end for
  3. Sort augmented images w.r.t yaw angle and put in pose buckets.        ▷ Pose based batching
Output: Return the pose batched image-landmark pairs.
```

Training details: Reported results are averaged over 5 runs. The following are the training details.

- **Learning rate for Γ and F:** $5 \times e^{-4}$.
- **Learning rate for head-pose network:** $5 \times e^{-5}$.
- **Noise level**: Randomly chosen within the range t = 1 to t = 10, where T = 50.
- **Iterations:** We optimize the embeddings for 10k iterations.
- **Optimizer:** We use the Adam Optimizer [23].
- **Batch Size:** 1

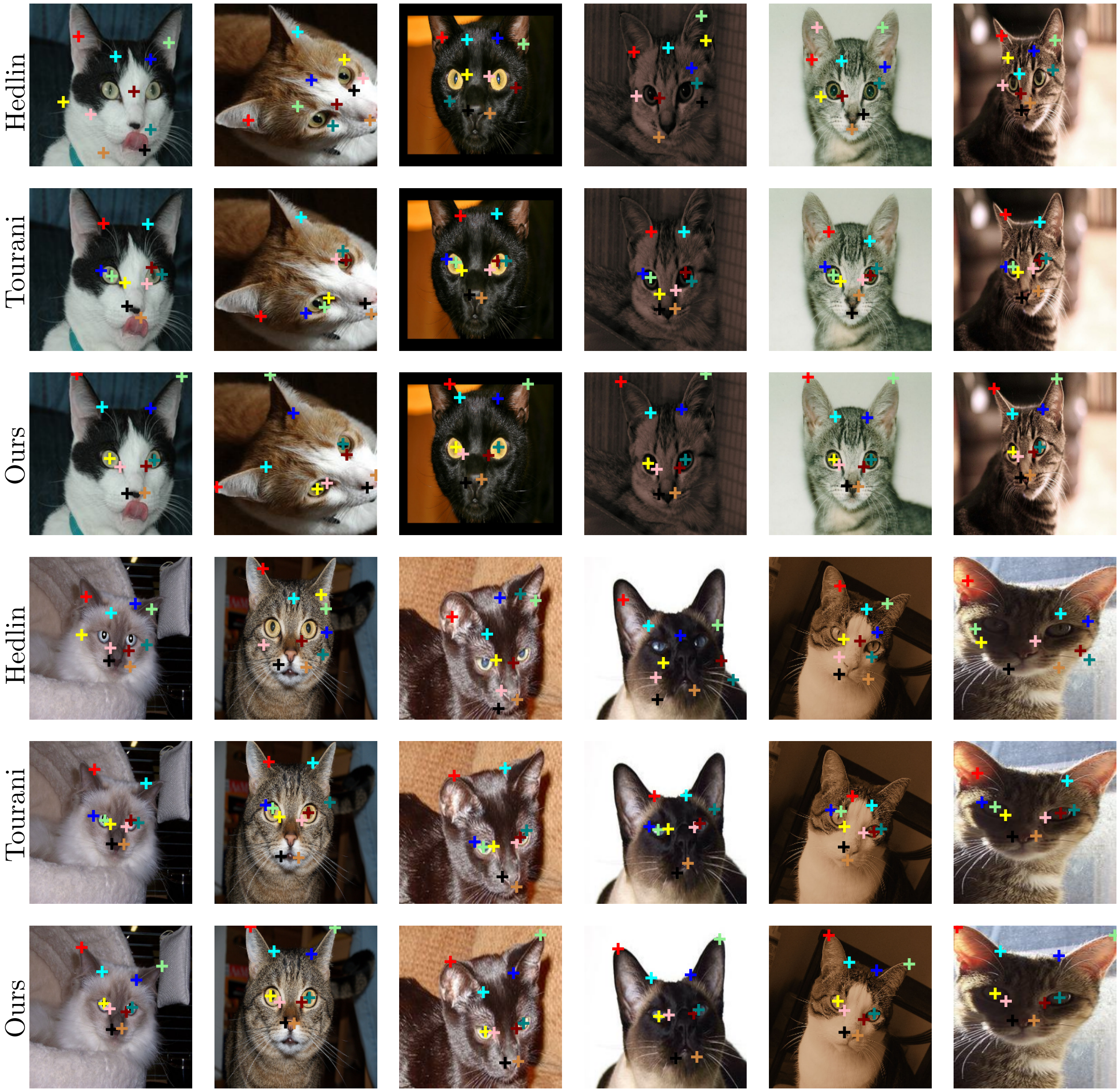


Figure 13. Comparison of CatHeads Results.

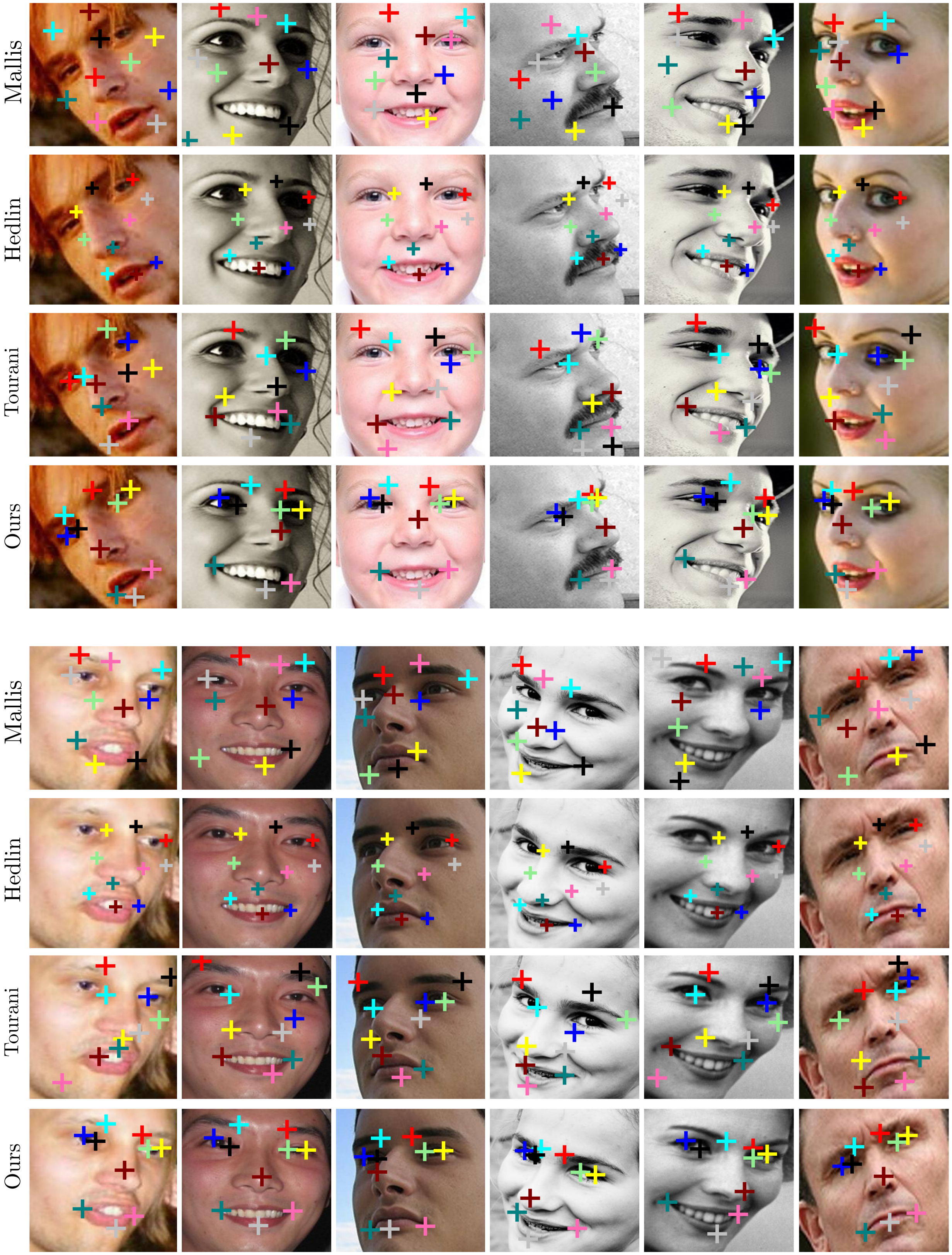


Figure 14. Comparison of AFLW Results.

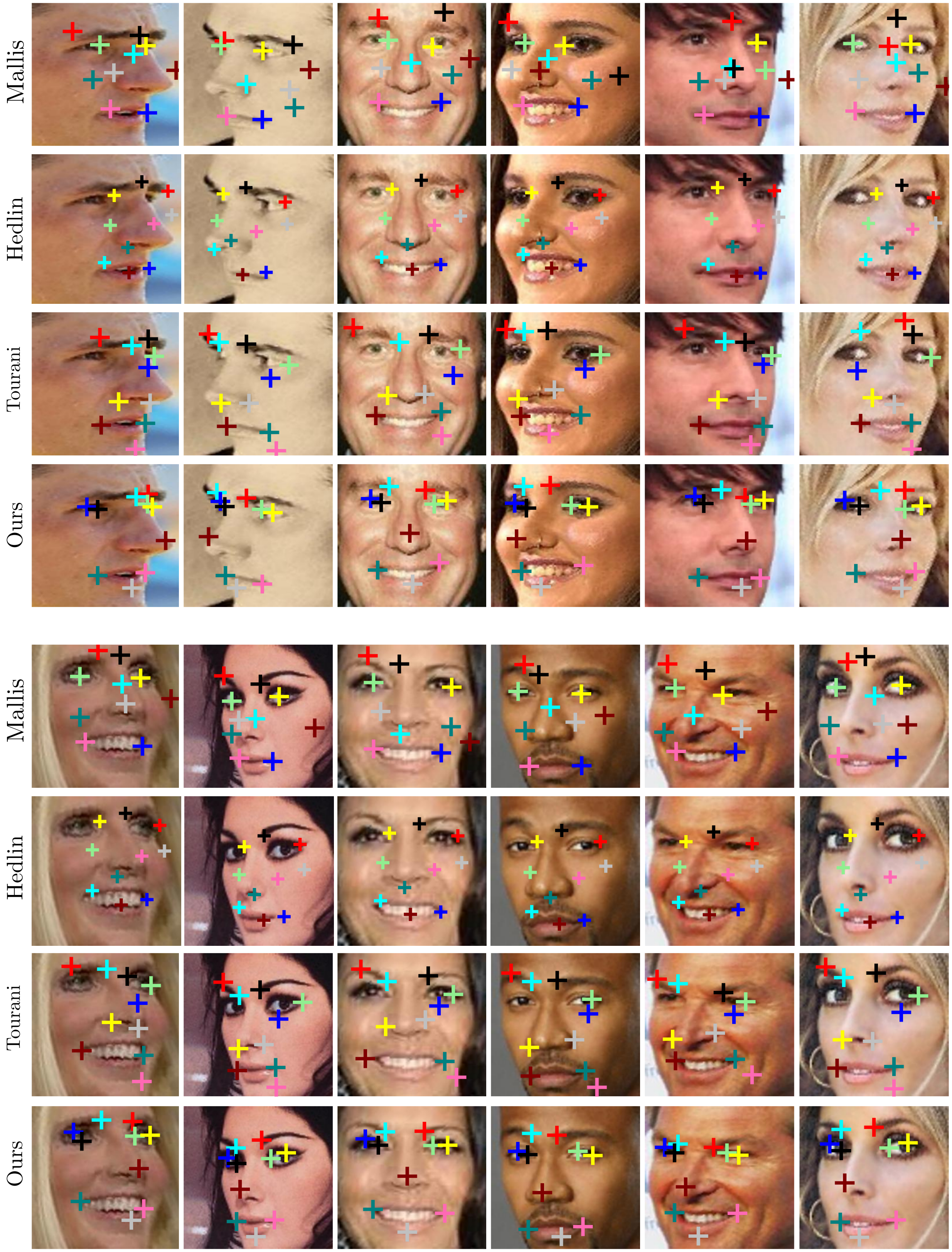


Figure 15. Comparison of MAFL Results.

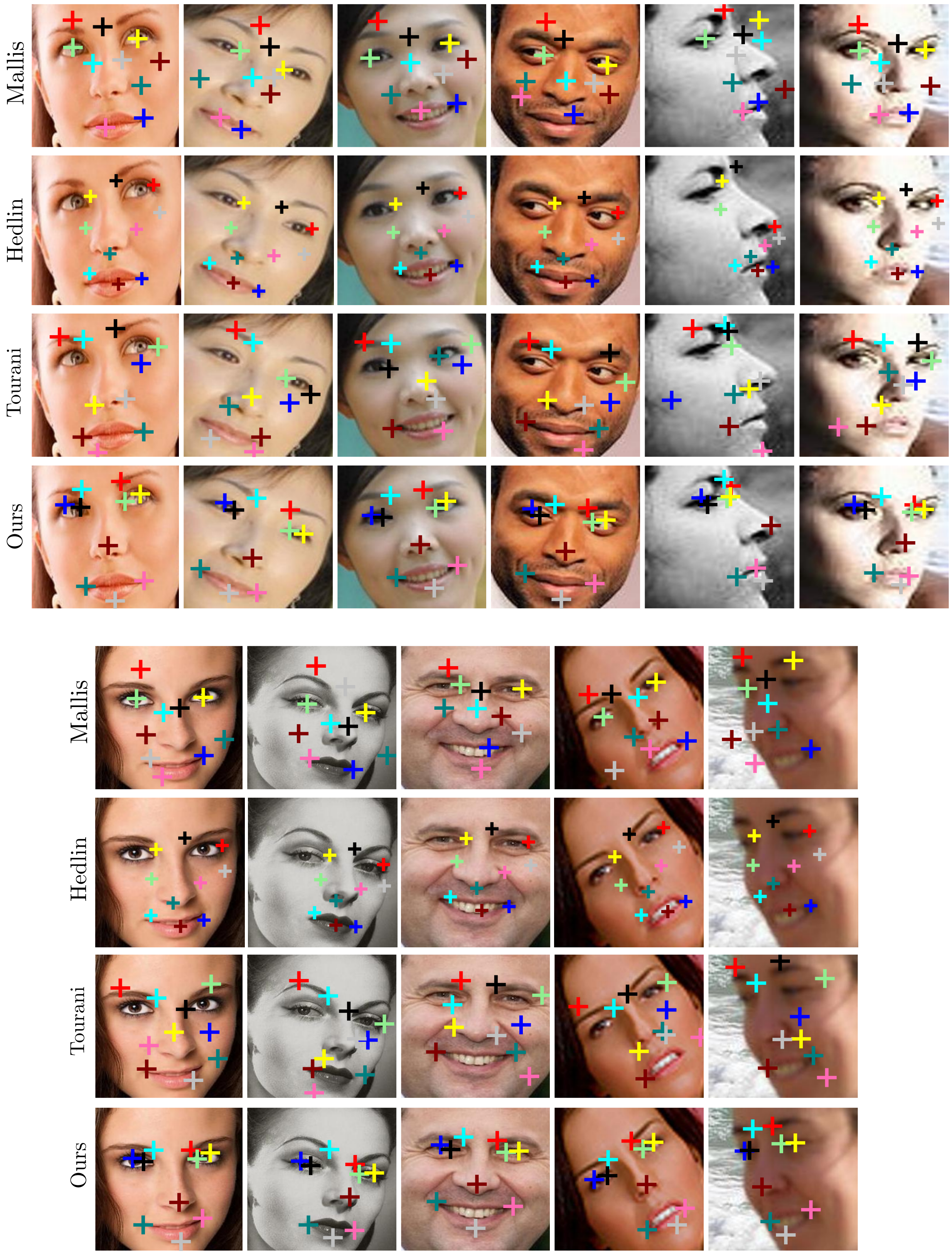


Figure 16. Comparison of LS3D Results.

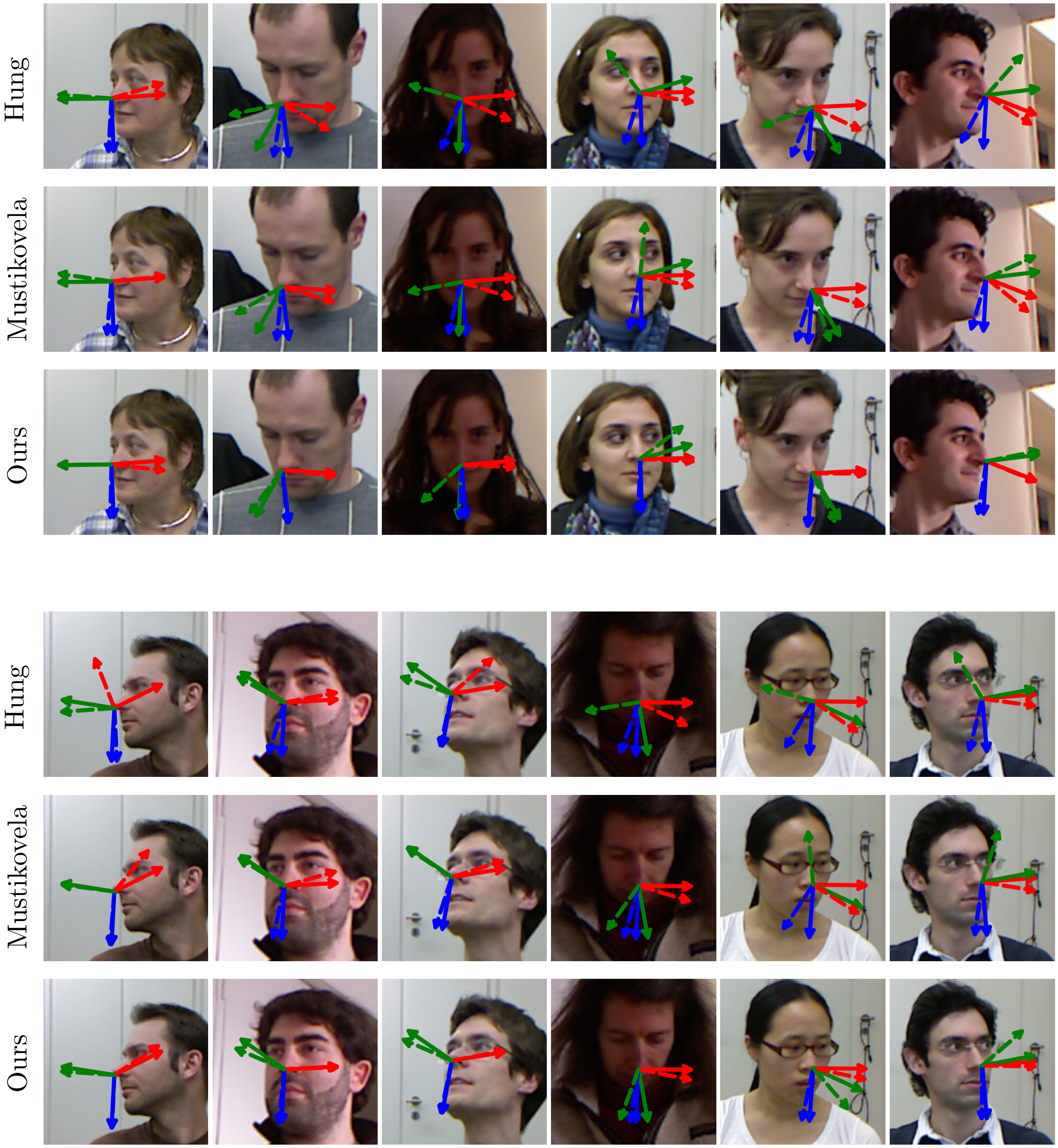


Figure 17. Comparison of Head-Pose Results.