

## A APPENDIX

### A.1 NETWORKS

Table 1: Network architectures.

<b>FeatNet</b>
2C9(6C9)-9C16-16C32-32C64-FC256-FC128-L2Norm for the COFW (300W) dataset
<b>CoordNet</b>
2C9(6C9)-9C16-16C32-32C64-FC128-L2Norm-FC2-Tanh for the COFW (300W) dataset
<b>RelCoordNet</b>
4C9-9C16-16C32-32C64-FC128-L2Norm-FC2-2Tanh for the COFW dataset
<b>dirPolNet</b>
FC512-FC256-FC8-Softmax

All convolution layers have the same kernel height and width,  $k_H = k_W = 3$ ,  $s = 2$ ,  $p = 1$ .

Table 2: Hyperparameters used.

	FeatNet	(Rel)CoordNet	dirPolNet
Optimizer	Adam	Adam	Adam
# Epochs	2000	2000	1000
Batch size	256	64	128
Initial lr	1E-2	1E-2 for projection, 1E-3 for head	5E-4
Lr decay	0.1	0.1	-
Decay epoch	1500epochs	1000epochs	-

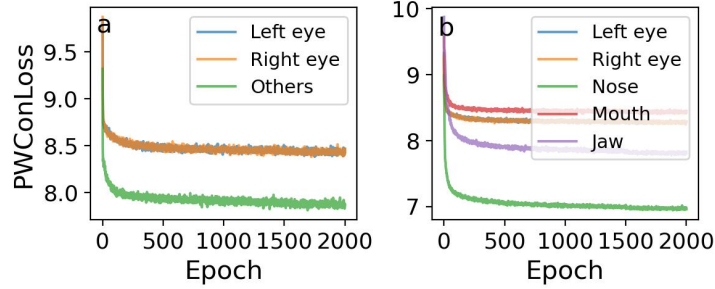


Figure 1: Learning curves of FeatNet for each group on (a) COFW and (b) 300W.

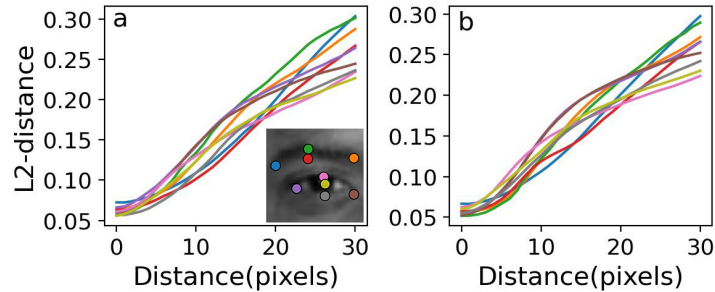


Figure 2: L2 distances between feature embeddings and landmarks at varying spatial distances from the landmarks in the inset for FeatNet (a) with  $o_{0:2C,:}$  and (b) with  $o$ .

**FeatNet.** Fig. 1 represents the learning curves of FeatNet. Due to variations in inter-landmark spatial distances across different groups and differing probabilities that a randomly sampled observation becomes a positive pair for a specific landmark, the absolute values of the PWConLoss vary

accordingly. Fig. 2 represents the results of using  $\mathbf{o}_{0:2C_{:,,:}}$  and the full observation  $\mathbf{o}$  as the input to FeatNet. The results indicate that the difference is negligible.

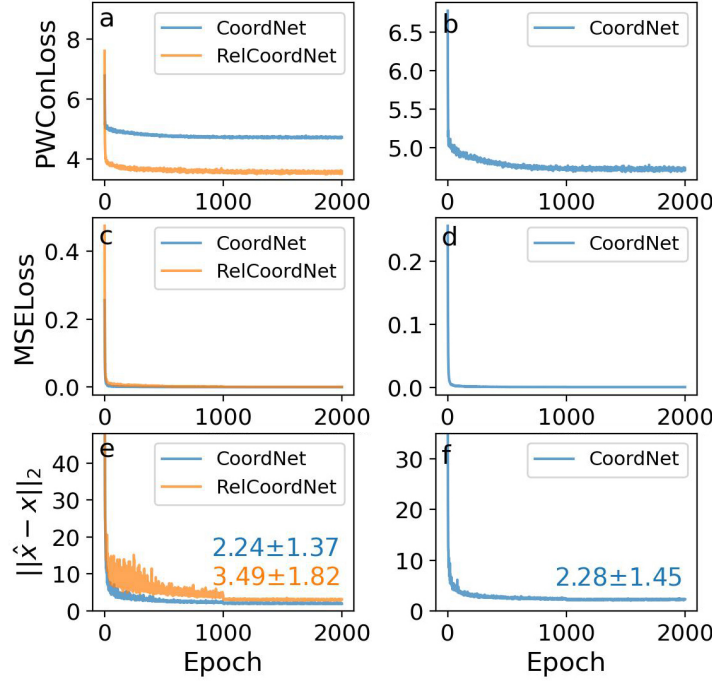


Figure 3: Learning curves of CoordNet and RelCoordNet. PWConLoss on (a) COFW and (b) 300W, MSELoss for regression head on (c) COFW and (d) 300W, coordinate regression error on (e) COFW and (f) 300W.

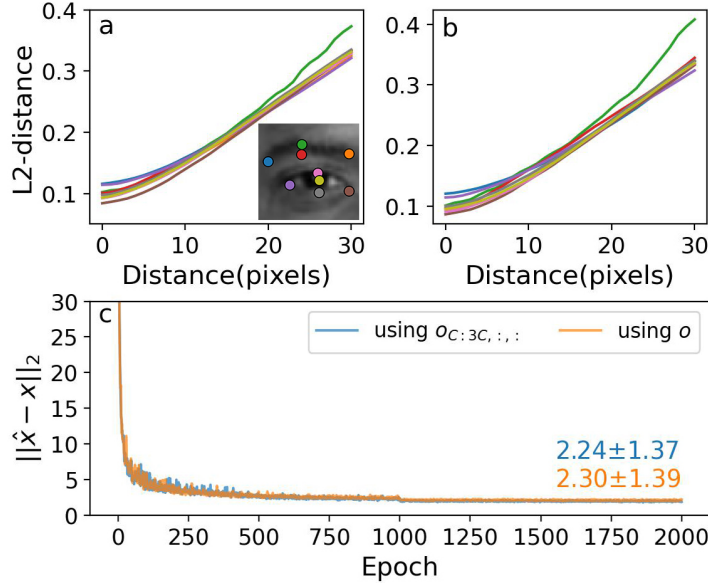


Figure 4: L2 distances between coordinate embeddings and landmarks at varying spatial distances from the landmarks in the inset for CoordNet with (a)  $\mathbf{o}_{C:3C, :, :}$  and (b)  $\mathbf{o}$ . (c) Coordinate regression error for CoordNet with  $\mathbf{o}_{C:3C, :, :}$  and  $\mathbf{o}$ .

**CoordNet/RelCoordNet.** Fig. 3 represents the learning curves of CoordNet and RelCoordNet. Our CoordNet achieves highly accurate coordinate regression with errors of only 2.24 pixels. Although

RelCoordNet shows slightly lower accuracy due to the more complex task, it still maintains high accuracy with an error of 3.49 pixels. Fig. 4 represents the results of using  $\mathbf{o}_{C:3C,\dots}$  and the full observation  $\mathbf{o}$  as the input to CoordNet. The results indicate that the difference is negligible.

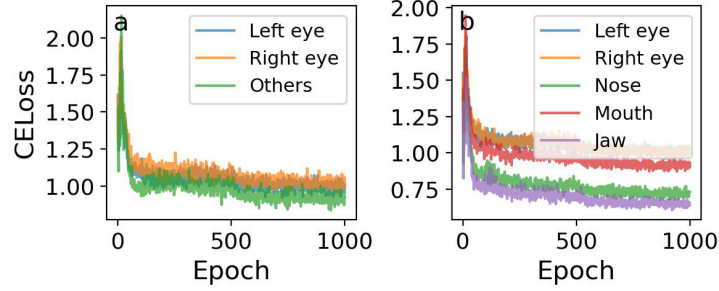


Figure 5: Learning curves of dirPolNet for each group on (a) COFW and (b) 300W.

**dirPolNet.** Fig. 5 represents the learning curves of dirPolNet. The cross-entropy loss is used. Similar to FeatNet, separate dirPolNet with distinct parameters is used for each group in the dataset.

## A.2 ADDITIONAL ABLATION STUDY



Figure 6: Trajectories of agents that fail to detect the target landmark of COFW. The blue circle represents the starting point, the yellow circles represent the intermediate trajectory, the red circle represents the final detection point, and the green circle represent the ground-truth landmark, respectively.

**Failure case analysis.** The hopping policy does not always guarantee convergence. When an agent fails to detection within a predefined number of steps, it invokes the delayed decision mechanism to select the most plausible point as the final landmark. Fig. 6 illustrates the trajectories of agents across stages and timesteps in failure cases. Although occlusion prevents the agent from reaching the true landmark, it ultimately chooses the optimal location based on its prior knowledge using the delayed-decision algorithm with SHT.

Table 3: Detection performance on COFW under ablation of the hyperparameter scheduling.

	$\lambda_{ft}$ scheduling	$\theta_d$ scheduling	NME	$T_d$	FLOPs
Baseline	✓	✓	8.28	10.42	21.1M
Case 1	✗	✓	8.92	11.40	23.1M
Case 2	✓	✗	8.77	28.00	56.7M
Case 3	✗	✗	8.76	34.04	68.9M

**Hyperparameter scheduling.** Our method employs hyperparameter scheduling for accurate detection. The ablation results of hyperparameter scheduling on COFW are provided in Table 3.

**Robustness to sample degradations.** To validate robustness of proposed method, we evaluate the detection performance under various degradation conditions. The results are summarized in Table 4.

Table 4: Detection performance on COFW under various degradations.

Degradation mode	NME	$T_d$	FLOPs
Baseline	8.28	10.42	21.1M
Blur ( $\sigma = 1$ )	8.64	14.51	29.4M
Blur ( $\sigma = 2$ )	8.72	14.87	30.1M
Blur ( $\sigma = 3$ )	8.90	15.72	31.8M
JPEG ( $Q = 80$ )	8.59	14.47	29.3M
JPEG ( $Q = 60$ )	8.65	14.42	29.3M
JPEG ( $Q = 40$ )	8.59	14.43	29.3M
JPEG ( $Q = 20$ )	8.64	14.46	29.3M
Motion blur ( $k = 5$ )	8.71	14.62	29.6M
Motion blur ( $k = 10$ )	8.86	15.29	31.0M
Occlusion (size=(20, 40))	8.74	14.78	29.9M

## A.3 DETECTION HYPERPARAMETERS

Table 5: Detection hyperparameters.

Symbol	Definition
$\theta_{d,\min}$	Initial threshold
$\Delta\theta_d$	Increase of threshold
$T_{d,\text{up}}$	Threshold increase starting time
$\lambda_{\text{ft},\max}$	Initial balance parameter
$\Delta\lambda_{\text{ft}}$	Decrease of balance parameter
$T_{d,\text{down}}$	Balance parameter decrease starting time

Table 6: Landmark-wise, stage-wise hyperparameter configuration on COFW.

	1	2	3	4	5	6	7	8
$\theta_{d,\min}(\times 10^{-3})$	9/2	3/1	6/2	10/4	6/2	6/2	4/2	8/3
$\Delta\theta_d(\times 10^{-3})$	4/2	3/2	3/1	3/1	3/1	3/1	4/1	3/1
$T_{d,\text{up}}$	7/1	1/2	1/2	9/1	1/2	1/2	1/3	4/3
$\lambda_{\text{ft},\max}$	0.89/0.90	0.99/0.99	0.79/0.81	0.78/0.82	0.79/0.81	0.79/0.81	0.73/0.90	0.91/0.97
$\Delta\lambda_{\text{ft}}$	0.18/0.08	0.16/0.09	0.18/0.10	0.19/0.07	0.18/0.10	0.18/0.10	0.18/0.08	0.15/0.09
$T_{d,\text{down}}$	6/6	9/7	5/5	9/9	5/5	5/5	9/10	7/8
	9	10	11	12	13	14	15	16
$\theta_{d,\min}(\times 10^{-3})$	6/2	9/3	6/2	5/2	10/3	6/2	8/2	9/2
$\Delta\theta_d(\times 10^{-3})$	3/1	4/1	3/1	5/1	4/2	3/1	4/1	4/1
$T_{d,\text{up}}$	1/2	1/3	1/2	3/3	2/1	1/2	5/4	4/3
$\lambda_{\text{ft},\max}$	0.79/0.81	0.94/0.98	0.79/0.81	0.78/0.84	0.83/0.84	0.79/0.81	0.93/0.99	0.89/0.98
$\Delta\lambda_{\text{ft}}$	0.18/0.10	0.19/0.09	0.18/0.10	0.16/0.07	0.14/0.08	0.18/0.10	0.12/0.08	0.19/0.09
$T_{d,\text{down}}$	5/5	9/5	5/5	7/7	8/7	5/5	9/5	8/5
	17	18	19	20	21	22	23	24
$\theta_{d,\min}(\times 10^{-3})$	6/2	5/2	8/3	9/4	7/3	10/3	9/2	8/3
$\Delta\theta_d(\times 10^{-3})$	3/1	5/1	4/2	4/2	2/1	3/1	4/2	3/1
$T_{d,\text{up}}$	1/2	3/3	6/1	8/1	1/1	5/1	2/2	3/2
$\lambda_{\text{ft},\max}$	0.79/0.81	0.78/0.84	0.74/0.76	0.73/0.74	0.77/0.91	0.96/0.99	0.76/0.99	0.94/0.96
$\Delta\lambda_{\text{ft}}$	0.18/0.10	0.16/0.07	0.16/0.07	0.11/0.06	0.16/0.09	0.14/0.10	0.18/0.06	0.13/0.10
$T_{d,\text{down}}$	5/5	7/7	8/7	6/8	8/7	7/7	7/6	5/9
	25	26	27	28	29			
$\theta_{d,\min}(\times 10^{-3})$	8/3	10/3	6/2	10/3	9/3			
$\Delta\theta_d(\times 10^{-3})$	4/2	3/1	4/2	3/1	4/2			
$T_{d,\text{up}}$	4/2	5/1	3/2	5/1	3/4			
$\lambda_{\text{ft},\max}$	0.77/0.93	0.96/0.99	0.90/0.96	0.96/0.99	0.87/0.91			
$\Delta\lambda_{\text{ft}}$	0.15/0.08	0.14/0.10	0.14/0.07	0.14/0.10	0.14/0.07			
$T_{d,\text{down}}$	6/7	7/7	9/5	7/7	6/6			

The detection hyperparameters are described in Table 5. We use two-stage detection technique. The landmark-wise, stage-wise hyperparameter configuration are detailed in Table 6 (COFW) and Table 7 (300W).

Table 7: Landmark-wise, stage-wise hyperparameter configuration on 300W.

	1	2	3	4	5	6	7	8
$\theta_{d,\min} (\times 10^{-3})$	5/2	8/8	5/2	9/5	6/1	4/2	3/1	10/2
$\Delta\theta_d (\times 10^{-3})$	5/2	5/5	5/2	5/2	4/2	4/2	5/1	4/2
$T_{d,\text{up}}$	4/6	1/1	1/1	1/4	5/3	5/2	3/1	5/1
$\lambda_{\text{ft,max}}$	0.78/0.99	0.80/0.80	0.85/0.97	0.79/0.91	0.95/0.96	0.95/0.95	0.97/0.99	0.88/0.97
$\Delta\lambda_{\text{ft}}$	0.11/0.10	0.11/0.11	0.10/0.07	0.10/0.10	0.18/0.10	0.11/0.07	0.15/0.07	0.14/0.06
$T_{d,\text{down}}$	5/9	8/8	10/7	10/9	6/5	10/8	10/8	6/10
	9	10	11	12	13	14	15	16
$\theta_{d,\min} (\times 10^{-3})$	5/2	4/4	5/2	3/1	3/1	10/4	10/4	10/5
$\Delta\theta_d (\times 10^{-3})$	3/1	3/3	4/2	5/1	4/2	2/1	4/2	5/2
$T_{d,\text{up}}$	3/2	4/4	4/2	1/1	1/2	5/1	9/1	1/3
$\lambda_{\text{ft,max}}$	0.83/0.98	0.86/0.86	0.94/0.99	0.95/0.98	0.90/0.91	0.94/0.95	0.78/0.88	0.73/0.80
$\Delta\lambda_{\text{ft}}$	0.12/0.06	0.18/0.18	0.17/0.07	0.15/0.10	0.16/0.08	0.14/0.07	0.13/0.07	0.18/0.09
$T_{d,\text{down}}$	6/9	7/7	8/7	5/7	7/6	7/5	8/8	10/6
	17	18	19	20	21	22	23	24
$\theta_{d,\min} (\times 10^{-3})$	3/1	9/9	6/1	9/3	3/1	5/1	10/1	10/1
$\Delta\theta_d (\times 10^{-3})$	5/2	3/3	5/2	4/2	4/2	4/2	2/1	2/1
$T_{d,\text{up}}$	5/1	8/8	1/5	2/2	3/7	1/5	5/5	5/5
$\lambda_{\text{ft,max}}$	0.91/0.92	0.99/0.99	1.00/1.00	0.93/1.00	0.99/0.99	0.88/1.00	0.90/1.00	0.90/1.00
$\Delta\lambda_{\text{ft}}$	0.14/0.07	0.20/0.20	0.12/0.07	0.11/0.06	0.14/0.08	0.13/0.06	0.20/0.05	0.20/0.05
$T_{d,\text{down}}$	5/9	7/7	6/5	10/7	7/6	5/6	10/10	10/10
	25	26	27	28	29	30	31	32
$\theta_{d,\min} (\times 10^{-3})$	9/4	5/5	9/3	10/1	10/1	9/2	10/1	10/1
$\Delta\theta_d (\times 10^{-3})$	4/1	5/5	3/1	2/1	2/1	4/2	2/1	2/1
$T_{d,\text{up}}$	5/8	4/4	6/3	5/5	5/5	1/2	5/5	5/5
$\lambda_{\text{ft,max}}$	0.86/0.97	0.95/0.95	0.97/0.98	0.90/1.00	0.90/1.00	0.98/0.99	0.90/1.00	0.90/1.00
$\Delta\lambda_{\text{ft}}$	0.11/0.08	0.12/0.12	0.16/0.09	0.20/0.05	0.20/0.05	0.14/0.09	0.20/0.05	0.20/0.05
$T_{d,\text{down}}$	9/7	6/6	10/9	10/10	10/10	5/8	10/10	10/10
	33	34	35	36	37	38	39	40
$\theta_{d,\min} (\times 10^{-3})$	10/1	10/10	10/1	10/1	3/1	9/2	9/2	9/2
$\Delta\theta_d (\times 10^{-3})$	2/1	2/2	2/1	2/1	2/1	3/1	3/1	3/1
$T_{d,\text{up}}$	5/5	5/5	5/5	5/5	5/2	8/1	8/1	8/1
$\lambda_{\text{ft,max}}$	0.90/1.00	0.90/0.90	0.90/1.00	0.90/1.00	0.85/0.99	0.99/1.00	0.99/1.00	0.90/0.99
$\Delta\lambda_{\text{ft}}$	0.20/0.05	0.20/0.20	0.20/0.05	0.20/0.05	0.20/0.07	0.20/0.08	0.20/0.08	0.20/0.08
$T_{d,\text{down}}$	10/10	10/10	10/10	10/10	10/6	7/6	7/6	7/6
	41	42	43	44	45	46	47	48
$\theta_{d,\min} (\times 10^{-3})$	6/3	3/3	10/1	5/2	7/2	10/4	7/3	10/1
$\Delta\theta_d (\times 10^{-3})$	4/2	2/2	2/1	3/1	4/1	2/1	5/1	2/1
$T_{d,\text{up}}$	1/10	6/6	5/5	1/3	2/2	2/5	4/3	5/5
$\lambda_{\text{ft,max}}$	0.91/1.00	1.00/1.00	0.90/1.00	0.99/0.99	0.92/0.95	0.91/0.96	0.96/0.99	0.90/1.00
$\Delta\lambda_{\text{ft}}$	0.15/0.05	0.19/0.19	0.20/0.05	0.16/0.06	0.12/0.06	0.13/0.06	0.19/0.08	0.20/0.05
$T_{d,\text{down}}$	10/6	7/7	10/10	6/8	8/7	6/6	6/6	10/10
	49	50	51	52	53	54	55	56
$\theta_{d,\min} (\times 10^{-3})$	1/1	4/4	7/3	10/1	10/3	9/4	7/3	7/3
$\Delta\theta_d (\times 10^{-3})$	4/2	4/4	3/1	2/1	3/1	4/1	2/1	4/2
$T_{d,\text{up}}$	1/3	1/1	10/2	5/5	10/2	8/6	1/1	4/2
$\lambda_{\text{ft,max}}$	0.95/0.98	0.91/0.91	0.98/0.99	0.90/1.00	0.99/1.00	0.99/1.00	0.99/0.99	0.93/0.95
$\Delta\lambda_{\text{ft}}$	0.14/0.06	0.15/0.15	0.11/0.08	0.20/0.05	0.15/0.05	0.18/0.09	0.19/0.08	0.12/0.09
$T_{d,\text{down}}$	7/7	9/9	6/6	10/10	8/10	6/9	6/10	7/8
	57	58	59	60	61	62	63	64
$\theta_{d,\min} (\times 10^{-3})$	4/2	9/9	9/4	5/2	8/4	7/2	2/1	3/1
$\Delta\theta_d (\times 10^{-3})$	4/1	5/5	4/1	3/1	4/2	3/2	3/1	4/2
$T_{d,\text{up}}$	1/3	2/2	1/3	5/3	7/2	3/3	1/2	6/2
$\lambda_{\text{ft,max}}$	0.93/1.00	0.98/0.98	0.96/0.98	0.98/0.99	0.99/0.99	1.00/1.00	0.96/1.00	0.99/1.00
$\Delta\lambda_{\text{ft}}$	0.11/0.10	0.13/0.13	0.17/0.06	0.15/0.09	0.15/0.09	0.16/0.06	0.12/0.05	0.12/0.07
$T_{d,\text{down}}$	9/7	9/9	9/7	10/9	7/5	5/8	8/9	10/6
	65	66	67	68				
$\theta_{d,\min} (\times 10^{-3})$	9/4	10/10	10/1	7/2				
$\Delta\theta_d (\times 10^{-3})$	3/2	3/3	2/1	4/1				
$T_{d,\text{up}}$	9/2	7/7	5/5	10/1				
$\lambda_{\text{ft,max}}$	0.97/0.98	1.00/1.00	0.90/1.00	0.98/1.00				
$\Delta\lambda_{\text{ft}}$	0.13/0.07	0.17/0.17	0.20/0.05	0.14/0.08				
$T_{d,\text{down}}$	9/10	9/9	10/10	5/10				

#### A.4 LANDMARK-WISE PERFORMANCE

Table 8: Landmark-wise detection performance on COFW.

	1	2	3	4	5	6	7	8	9	10	11	12
NME	10.42	9.37	8.24	8.34	8.70	8.40	9.10	8.45	8.26	7.34	6.53	6.49
$T_d$	6.98	7.15	10.96	8.30	11.21	10.99	10.42	8.90	11.26	8.46	11.85	11.74
	13	14	15	16	17	18	19	20	21	22	23	24
NME	7.70	6.81	6.18	6.06	7.29	6.07	8.04	7.09	8.42	7.80	10.26	9.66
$T_d$	7.89	9.33	9.95	8.54	11.43	12.43	10.45	10.50	8.91	9.44	10.40	11.93
	25	26	27	28	29	mean						
NME	8.67	8.62	9.47	10.45	12.02	<b>8.28</b>						
$T_d$	10.11	11.63	12.04	10.15	18.36	<b>10.42</b>						

Table 9: Landmark-wise detection performance on 300W.

	1	2	3	4	5	6	7	8	9	10	11	12
NME	13.68	12.68	13.01	14.51	13.38	13.03	11.97	10.73	9.74	10.28	11.00	12.07
$T_d$	23.80	18.83	19.67	17.83	19.50	16.92	14.56	11.92	13.67	15.09	14.54	16.09
	13	14	15	16	17	18	19	20	21	22	23	24
NME	13.56	13.07	12.83	12.13	14.06	10.62	9.49	10.33	9.87	10.70	9.44	9.49
$T_d$	18.78	26.02	24.22	17.37	21.61	8.57	10.11	6.50	13.94	10.13	11.92	13.07
	25	26	27	28	29	30	31	32	33	34	35	36
NME	9.97	9.49	10.67	8.25	9.53	13.00	15.99	11.52	10.85	10.92	10.96	11.00
$T_d$	10.88	11.47	9.63	14.60	12.26	6.81	14.23	20.55	19.95	19.24	19.66	21.35
	37	38	39	40	41	42	43	44	45	46	47	48
NME	6.43	6.11	6.31	8.02	6.69	6.61	7.08	5.85	6.13	6.11	5.93	6.10
$T_d$	13.48	8.53	9.19	10.61	7.35	9.39	11.79	7.19	8.88	10.97	7.51	10.56
	49	50	51	52	53	54	55	56	57	58	59	60
NME	8.25	7.14	7.54	7.41	7.72	6.81	8.61	7.28	7.41	6.55	7.09	7.29
$T_d$	9.70	8.64	8.98	14.70	7.92	8.48	7.51	9.77	9.22	6.97	7.14	12.01
	61	62	63	64	65	66	67	68	mean			
NME	7.62	6.71	6.45	7.00	7.67	7.26	6.80	7.26	<b>9.36</b>			
$T_d$	7.28	8.08	9.13	11.54	7.45	8.92	13.94	10.31	<b>12.77</b>			

The landmark-wise detection performance is detailed in Table 8 (COFW) and Table 9 (300W).

#### A.5 RESULTS ON WFLW

Table 10: Comparison of our method with SoTA approaches on WFLF.

Method	NME-ocular	# Params (M)	FLOPs	Duration $T_d$
LAB (Wu et al., 2018)	5.27	25.1	18.9G	-
AWing (Wang et al., 2019)	4.36	24.2	26.8G	-
AVS (Qian et al., 2019)	4.39	28.3	2.40G	-
DAG (Li et al., 2020)	4.21	21.0	-	-
HRNet (Wang et al., 2020)	4.60	9.66	4.75G	-
PIP (Jin et al., 2021)	4.57	12.0	2.40G	-
ADNet (Huang et al., 2021)	4.14	13.4	17.0G	-
HIH (Lan et al., 2021)	4.18	22.7	17.2G	-
SDFL (Lin et al., 2021)	4.35	-	5.17G	-
HIH (Lan et al., 2021)	4.08	22.7	17.2G	-
SLPT (Xia et al., 2022)	4.20	13.2	6.12G	-
STARLoss (Zhou et al., 2023)	4.02	13.4	-	-
D-ViT (Dang et al., 2025)	3.75	67.3	21.8G	-
PoPos (Xiang et al., 2025)	3.95	9.70	1.20G	-
<b>Ours</b>	<b>13.60</b>	<b>0.577</b>	<b>44.8M</b>	<b>19.67</b>

We evaluated the performance of our method on WFLW dataset. Table 10 presents the comparison of our method with SoTA approaches.

## A.6 PSEUDOCODE

---

### Algorithm 1 Operation of proposed agent.

---

**Input:**  $c, \text{Image}, \mathbf{x}, t_{\max} \theta_{d,\min} \in \mathbb{R}^2, \Delta\theta_d \in \mathbb{R}^2, T_{d,\text{up}} \in \mathbb{R}^2, \lambda_{\text{ft},\max} \in \mathbb{R}^2, \Delta\lambda_{\text{ft}} \in \mathbb{R}^2, T_{d,\text{down}} \in \mathbb{R}^2$

**Output:**  $\hat{\mathbf{x}}_c$

```

1:  $z_{c,\text{ft}}^*, z_{c,\text{cd}}^* \leftarrow \text{Prior knowledge modeling}(c)$ 
2:  $\hat{\mathbf{x}}_c \leftarrow \text{NULL}$ 
3: for stage = 0 to 1 do
4:    $\theta_d \leftarrow \theta_{d,\min}[\text{stage}]$ 
5:    $\Delta\theta_d \leftarrow \Delta\theta_d[\text{stage}]$ 
6:    $T_{d,\text{up}} \leftarrow T_{d,\text{up}}[\text{stage}]$ 
7:    $\lambda_{\text{ft}} \leftarrow \lambda_{\text{ft},\max}[\text{stage}]$ 
8:    $\lambda_{\text{cd}} \leftarrow 1 - \lambda_{\text{ft}}$ 
9:    $\Delta\lambda_{\text{ft}} \leftarrow \Delta\lambda_{\text{ft}}[\text{stage}]$ 
10:   $T_{d,\text{down}} \leftarrow T_{d,\text{down}}[\text{stage}]$ 
11:   $\hat{\Lambda} \leftarrow \{\lambda_{\text{ft}}[t] | \forall t \in [0, t_{\max}]\}$ 
12:  for  $t = 0$  to  $t_{\max}$  do
13:     $\mathbf{o} \leftarrow \text{Observation by agent}(\text{Image}, \mathbf{x})$ 
14:     $z_{\text{ft}}, z_{\text{cd}}, \hat{\mathbf{x}} \leftarrow \text{Data extraction}(\mathbf{o})$ 
15:     $D_{\text{ft}}, D_{\text{cd}}, D \leftarrow \text{DistComp}(z(\cdot), z_{c,\cdot}^*, \lambda(\cdot)), \quad \text{where } (\cdot) \in \{\text{ft}, \text{cd}\}$ 
16:     $\hat{\mathbf{x}}_c \leftarrow \text{Delayed decision}(\hat{\Lambda}, D_{\text{ft}}, D_{\text{cd}}, \theta_d, \lambda_{\text{ft}}, \hat{\mathbf{x}})$ 
17:    if  $\hat{\mathbf{x}}_c \neq \text{NULL}$  then
18:       $\mathbf{x} \leftarrow \hat{\mathbf{x}}_c$ 
19:      Terminate.
20:    end if
21:    if  $t \geq T_{d,\text{up}}$  then
22:       $\theta_d \leftarrow \theta_d + \Delta\theta_d$ 
23:    end if
24:    if  $t \geq T_{d,\text{down}}$  and  $t - T_{d,\text{down}} = \text{even}$  then
25:       $\lambda_{\text{ft}} \leftarrow \lambda_{\text{ft}} - \Delta\lambda_{\text{ft}}$ 
26:    end if
27:     $s \cdot \mathbf{u}^* \leftarrow \text{Hopping policy}(D, z(\cdot), z_{c,\cdot}^*, \lambda(\cdot)), \quad \text{where } (\cdot) \in \{\text{ft}, \text{cd}\}$ 
28:     $\mathbf{x} \leftarrow \mathbf{x} + s \cdot \mathbf{u}^*$ 
29:  end for
30: end for

```

---

Algorithm 1 summarizes the overall procedure of the two-stage detection process for landmark  $c$ .

---

## REFERENCES

- Ziqiang Dang, Jianfang Li, and Lin Liu. Cascaded dual vision transformer for accurate facial landmark detection. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 5884–5894. IEEE, 2025.
- Yangyu Huang, Hao Yang, Chong Li, Jongyoo Kim, and Fangyun Wei. Adnet: Leveraging error-bias towards normal direction in face alignment. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 3080–3090, 2021.
- Haibo Jin, Shengcai Liao, and Ling Shao. Pixel-in-pixel net: Towards efficient facial landmark detection in the wild. *International Journal of Computer Vision*, 129:3174–3194, 2021.
- Xing Lan, Qinghao Hu, Qiang Chen, Jian Xue, and Jian Cheng. Hih: Towards more accurate face alignment via heatmap in heatmap. *arXiv preprint arXiv:2104.03100*, 2021.
- Weijian Li, Yuhang Lu, Kang Zheng, Haofu Liao, Chihung Lin, Jiebo Luo, Chi-Tung Cheng, Jing Xiao, Le Lu, Chang-Fu Kuo, et al. Structured landmark detection via topology-adapting deep graph learning. In *European Conference on Computer Vision*, pp. 266–283. Springer, 2020.
- Chunze Lin, Beier Zhu, Quan Wang, Renjie Liao, Chen Qian, Jiwen Lu, and Jie Zhou. Structure-coherent deep feature learning for robust face alignment. *IEEE Transactions on Image Processing*, 30:5313–5326, 2021.
- Shengju Qian, Keqiang Sun, Wayne Wu, Chen Qian, and Jiaya Jia. Aggregation via separation: Boosting facial landmark detector with semi-supervised style translation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10153–10163, 2019.
- Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Minghui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 43: 3349–3364, 2020.
- Xinyao Wang, Liefeng Bo, and Li Fuxin. Adaptive wing loss for robust face alignment via heatmap regression. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6971–6981, 2019.
- Wayne Wu, Chen Qian, Shuo Yang, Quan Wang, Yici Cai, and Qiang Zhou. Look at boundary: A boundary-aware face alignment algorithm. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2129–2138, 2018.
- Jiahao Xia, Weiwei Qu, Wenjian Huang, Jianguo Zhang, Xi Wang, and Min Xu. Sparse local patch transformer for robust face alignment and landmarks inherent relation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4052–4061, 2022.
- Chong-Yang Xiang, Jun-Yan He, Zhi-Qi Cheng, Xiao Wu, and Xian-Sheng Hua. Popos: Improving efficient and robust facial landmark detection with parallel optimal position search. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 8602–8610, 2025.
- Zhenglin Zhou, Huaxia Li, Hong Liu, Nanyang Wang, Gang Yu, and Rongrong Ji. Star loss: Reducing semantic ambiguity in facial landmark detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15475–15484, 2023.