On Equivariant and Invariant Learning of Object Landmark Representations — Supplementary Material

Zezhou Cheng Jong-Chyi Su Subhransu Maji
University of Massachusetts Amherst
{zezhoucheng, jcsu, smaji}@cs.umass.edu

The following describes the content in each section in the supplementary material.

- § 1 provides further analysis and discusses the main limitations of the proposed model.
- § 2 shows additional results of landmark matching and regression.
- § 3 describes additional implementation details.
- § 4 details the proposed birds benchmark.
- § 5 demonstrates the effectiveness of the proposed method for the task of figure-ground segmentation.
- § 6 provides the numbers corresponding to Figure 4 in the main paper.

1. Further analysis

1.1. Visualization of feature embeddings

We visualize the first few PCA (uncentered) components of the learned model and a randomly initialized model in Fig. 1. Specifically, we sample hypercolumns from 32 MAFL images using our contrastively pre-trained ResNet50, treat each spatial location separately, and compute the PCA basis vectors. We then project the hypercolumns to each basis and visualize the projection as a spatial map. Observe that the bases encode information about the background, foreground, and landmark regions (*e.g.* eyes, nose, and mouth) of faces. On the other hand, the bases of a randomly initialized model show no such structure.

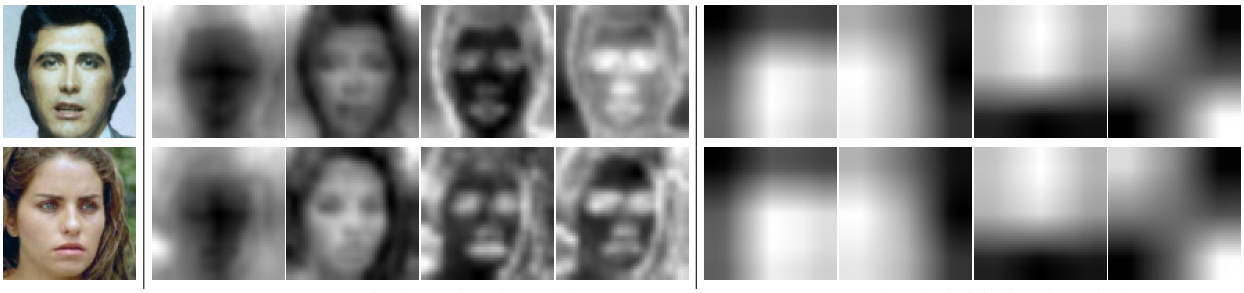

Figure 1. **PCA visualization of the hypercolumn representation.** From left to right: input image, and the projection of hypercolumns on the first four PCA bases from a contrastively trained and a randomly initialized ResNet50.

1.2. Fine-tuning the network

Tab. 1.2 shows the effect of end-to-end fine-tuning of all layers of our model for landmark regression. In this experiment, we use the output of the fourth convolutional block of ResNet50 as the representation, which is the optimal single layer representation. We did *not* find fine-tuning to be uniformly beneficial — fine-tuning is worse than training the linear regressor only on MAFL dataset, while it is better on AFLW dataset. We speculate the reason to be the domain gap between the datasets used for unsupervised learning and supervised learning of the regressor, which is exacerbated by the small training sets. For example, AFLW has a larger domain gap than CelebA to MAFL, which is also noticed in DVE [4].

	MAFL	AFLW$_M$	AFLW$_R$	300W
w/o fine-tuning	2.73	8.83	7.37	6.01
w/ fine-tuning	2.81	7.80	6.99	5.94

Table 1. **Effect of fine-tuning for landmark regression.** The fourth block of a ResNet50 network was used as the representation.

1.3. Memory efficiency

Tab. 1.3 compares the memory efficiency of DVE [4] to ours. DVE maintains high-resolution feature maps across the network hierarchy to compute the equivariance loss. By comparison, our contrastive learning loss is computed on a global image representation which requires less memory, allowing bigger models. Our method with a "ResNet50-half", which halves the layer width of the original network, achieves comparable performance with DVE (see Tab. 4 and Tab. 5) but is more memory-efficient.

Method	Network	# Params (M)	Network Size (MB)	Memory (MB)
DVE [4]	Hourglass	12.61	48.10	491.85
Ours	ResNet18	11.24	42.89	11.54
	ResNet50-half	6.03	23.02	28.15
	ResNet50	23.77	90.68	52.65

Table 2. **Memory efficiency.** Comparison of DVE [4] with ours in terms of number of network parameters (# Params), memory required for storing the network (Network Size), and the memory usage of a forward and backward pass for a single 96×96 RGB image (Memory).

1.4. Effect of dimensionality reduction

Fig. 2 presents the landmark matching performance as a function of the projection dimension. Specifically, We evaluate the cross-identity landmark matching on the MAFL test set with different projection dimensions. This usually improves performance across all dimensions as the mean pixel error with hypercolumn representations is 6.16.

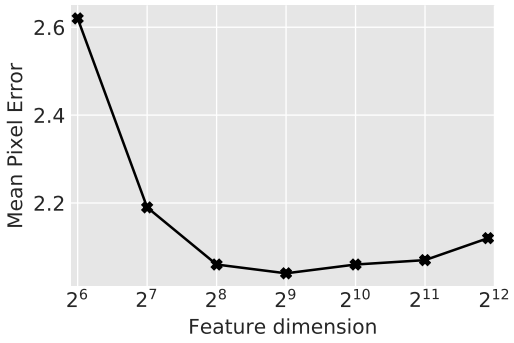

Figure 2. **Landmark matching performance as a function of the projection dimension.** The mean pixel error of the raw hypercolumn representations is 6.16 (not shown), which is higher than the projected representation across all dimensions.

1.5. Higher resolution images

Tab. 3 presents the performance of our model on higher-resolution images. Specially, we train our model on 128×128 CelebA and iNaturalist Aves images (instead of 96×96 used in the experiments in the main paper and DVE [4]), We conduct the linear evaluation with 128×128 images from face and bird benchmarks. However, the memory and the computational requirement is a challenge where our approach provides an advantage. A forward and backward pass on a single 128×128 image takes 874.41 MB for DVE with Hourglass network while it takes only 93.58 MB for the proposed model with ResNet50 as the backbone. While our method could be trained in 3 days on 128×128 CelebA images, we could not finish training DVE with Hourglass or ResNet50 network in two weeks despite our best efforts. We notice that using higher resolution images generally improves the performance across benchmarks.

Resolution	MAFL	AFLW$_M$	AFLW$_R$	300W	CUB
96×96	2.44	6.99	**6.27**	5.22	68.63
128×128	**2.34**	**6.87**	6.41	**4.99**	**72.61**

Table 3. **The effect of image resolution.** We use ResNet50 with hypercolumn representations. We report the error in the percentage of inter-ocular distance on human face dataset (*lower is better*), and the percentage of correct keypoints on CUB dataset (*higher is better*).

1.6. Limitations

Our method shares the limitations of many unsupervised learning approaches. Without supervision the landmark representations are not guaranteed to be interpretable, though we show that semantic parts can be estimated with limited or no supervision in many cases (see Fig. 5 in the main paper); The detailed nature of the emergence of equivariance and invariance and their extension to other tasks (*e.g.* object detection) remains open and active area of research in the community.

2. Additional results of landmark matching and regression

We provide more experimental results of landmark matching and regression in Tab. 4 and Tab. 5 respectively. We report results under various settings of (1) initialization methods (*e.g.*, randomly initialized, ImageNet pre-trained, DVE [4], or the proposed contrastively pre-trained); (2) network architectures (*e.g.*, ResNet18, ResNet50, or ResNet50-half which halves the layer width of ResNet50); (3) training dataset for representation learning (aligned or in-the-wild CelebA dataset); (4) representations (hypercolumns or its projected features (+Proj.)). The proposed method surpasses DVE across different network architectures, projected feature dimensions, and curated or uncurated datasets in both landmark matching and regression tasks.

3. Other implementation details

Training details of unsupervised learning models. We use MoCo [2] as our contrastive learning model. We train MoCo for 800 epochs with a batch size of 256 and a cosine learning rate schedule as proposed in MoCo-v2 [1]. However, we did not observe improvements in our task when using other tricks in MoCo-v2 [1], such as adding Gaussian blur for the data augmentation and using an MLP as the projection network. We use the public implementation[1] of MoCo from [5]. For a comparison with the DVE model [4] on the proposed bird dataset, we use their publicly available implementation[2].

Training details of feature projection. We set the temperature hyperparameter τ to $1/7$ in the equivariance loss (Equation 3 in the main text) for training the linear feature projector. We implement the linear projector as a 1×1 convolutional layer and train the projector for 10 epochs on the CelebA dataset [3] with the equivariance loss. Notice that we do not apply any data augmentations during training, and the training of feature projector does not require any human annotations. We use Adam optimizer with a learning rate of 0.001 and a weight decay of 0.0005.

Training details of linear regression. **(1) Data augmentation**: we do not use any data augmentation when the *entire* annotations of the training data are provided (Tab. 2 in the main paper). However, in the *limited* annotation experiments on human face benchmarks, following DVE [4], we apply thin-plate spline as the data augmentation method with the same deformation hyperparameters as DVE (Fig. 4a in the main paper). We do not apply any augmentations during landmark regression on the CUB dataset (Tab. 2 and Fig. 4b in the main paper). **(2) Validation**: due to the lack of validation set on face benchmarks, we train the linear regressor for 120, 45, and 80 epochs on MAFL, AFLW, and 300W dataset respectively when hypercolumns are used, and we train for 150 epochs uniformly across these benchmarks when the compact representations of the hypercolumn are used. On CUB, the results are reported from the checkpoint selected on the validation set. In the ablation study of the effectiveness of unsupervised learning (Tab. 5 in the main paper), for the ImageNet pre-trained or randomly initialized networks, we report the best performance on the test set within 2000 training epochs. **(3) Learning rate**: on face benchmarks, we use an initial learning rate of 0.01 and a weight decay of 0.05 when only limited annotations are available (Fig. 4a,b in the main paper). The two hyperparameters are 0.001 and 0.0005 respectively when the entire annotations are given (Tab. 2 in the main paper); On CUB, if the number of annotations is smaller or equal to 100 (e.g. 10, 50, 100), we use an initial learning rate of 0.01 and a weight decay of 0.05. The two hyperparameters are 0.01 and 0.005 respectively if more annotations (e.g. 250, 500, 1241) are available. We apply the cosine learning rate schedule in all of our experiments.

[1] https://github.com/HobbitLong/CMC
[2] https://github.com/jamt9000/DVE

Method	Network	# Params (M)	In-the-wild	+Proj.	Dim.	Same identity	Diff. identity
Random	ResNet50	23.77			3840	1.07	10.03
Random	ResNet50	23.77		✓	256	3.68	7.04
Random	ResNet50	23.77		✓	128	3.70	7.03
Random	ResNet50	23.77		✓	64	3.71	7.10
ImageNet	ResNet50	23.77			3840	0.67	6.50
ImageNet	ResNet50	23.77		✓	256	0.82	3.15
ImageNet	ResNet50	23.77		✓	128	1.00	3.39
ImageNet	ResNet50	23.77		✓	64	1.55	4.44
DVE [4]	Smallnet	0.35			64	1.28	2.77
DVE [4]	Hourglass	12.61			64	0.92	2.38
DVE [4]	Hourglass	12.61	✓		64	1.27	3.52
Ours	ResNet50	23.77			3840	**0.73**	6.16
Ours	ResNet50	23.77		✓	256	**0.71**	**2.06**
Ours	ResNet50	23.77		✓	128	**0.82**	**2.19**
Ours	ResNet50	23.77		✓	64	**0.92**	2.62
Ours	ResNet50	23.77	✓		3840	**0.78**	5.58
Ours	ResNet50	23.77	✓	✓	256	**0.96**	**3.03**
Ours	ResNet50	23.77	✓	✓	128	**0.98**	**3.05**
Ours	ResNet50	23.77	✓	✓	64	**0.99**	**3.06**
Ours	ResNet50-half	6.03			3840	**0.74**	5.84
Ours	ResNet50-half	6.03		✓	256	**0.76**	**2.18**
Ours	ResNet50-half	6.03		✓	128	**0.88**	**2.38**
Ours	ResNet50-half	6.03		✓	64	1.05	2.85
Ours	ResNet18	11.24			3840	**0.64**	4.95
Ours	ResNet18	11.24		✓	256	**0.71**	**2.20**
Ours	ResNet18	11.24		✓	128	**0.82**	**2.31**
Ours	ResNet18	11.24		✓	64	1.00	2.74

Table 4. **Landmark matching**. The mean pixel error between the predicted landmarks and the ground-truth (*lower is better*). Results better than DVE's are in **bold**.

4. Birds benchmark

The Birds benchmark consists of unsupervised learning on the images from the iNaturalist Aves taxa and evaluating them on the landmarks in the CUB dataset. Specially, we randomly sample 100K images of birds from the iNaturalist 2017 dataset [6] under "Aves" class. Fig. 3 top row shows images from iNaturalist Aves dataset. The dataset contains objects in significant clutter, occlusion, and with a wider range of pose, viewpoint, and size variations than those in face benchmarks. Some images even contain multiple objects. To test the performance in the few-shot setting, we sample a subset of the CUB dataset which contains similar species to iNaturalist. Specifically, we sample 35 species of *Passeroidea* superfamily, each annotated with 15 landmarks. Fig. 3 bottom row shows images from the CUB dataset. We sample at most 60 images per class and conduct the splitting of training, validation, and test set on the samples of each species in a ratio of 3:1:1. These splits are then combined, which results in 1241 training images, 382 validation images, and 383 test images.

5. Figure-ground Segmentation

We extend the model to tackle a figure-ground segmentation task. Specifically, we train a 1×1 convolutional layer on the top of the learned pixel representations to predict the foreground mask. We evaluate the segmentation model on the CUB dataset described in Sec. 4, which comes with annotated masks. Tab. 6 compares representations from randomly initialized, ImageNet pre-trained, and our contrastively learned networks under different sizes of the training set. The contrastive model is trained on iNaturalist Aves dataset, as described in the main text. Under the linear evaluation setting where the backbone is fixed,

Method	Network	# Params (M)	In-the-wild	+Proj.	Dim.	MAFL	AFLW$_M$	AFLW$_R$	300W
Random	ResNet50	23.77			3840	4.72	16.74	11.23	11.70
Random	ResNet50	23.77		✓	256	6.56	20.33	13.50	11.67
Random	ResNet50	23.77		✓	128	6.21	20.25	13.99	17.12
Random	ResNet50	23.77		✓	64	6.28	19.76	13.53	16.93
ImageNet	ResNet50	23.77			3840	2.98	8.88	7.34	6.88
ImageNet	ResNet50	23.77		✓	256	3.51	9.69	8.02	7.02
ImageNet	ResNet50	23.77		✓	128	3.36	9.11	7.68	6.55
ImageNet	ResNet50	23.77		✓	64	3.50	9.43	7.63	6.51
DVE	Smallnet	0.35			64	3.42	8.60	7.79	5.75
DVE	Hourglass	12.61			64	2.86	7.53	6.54	4.65
DVE	Hourglass	12.61	✓		64	3.23	8.52	7.38	5.05
Ours	ResNet50	23.77			3840	**2.44**	**6.99**	**6.27**	5.22
Ours	ResNet50	23.77		✓	256	**2.64**	**7.17**	**6.14**	4.99
Ours	ResNet50	23.77		✓	128	**2.71**	**7.14**	**6.14**	5.09
Ours	ResNet50	23.77		✓	64	**2.77**	**7.21**	**6.22**	5.19
Ours	ResNet50	23.77	✓		3840	**2.46**	**7.57**	**6.29**	**5.04**
Ours	ResNet50	23.77	✓	✓	256	**2.82**	**7.69**	**6.67**	5.27
Ours	ResNet50	23.77	✓	✓	128	**2.88**	**7.81**	**6.79**	5.37
Ours	ResNet50	23.77	✓	✓	64	**3.00**	**7.87**	**6.92**	5.59
Ours	ResNet50-half	6.03			3840	**2.46**	**7.37**	6.71	5.33
Ours	ResNet50-half	6.03		✓	256	**2.66**	**7.21**	**6.32**	5.20
Ours	ResNet50-half	6.03		✓	128	**2.75**	**7.26**	**6.32**	5.26
Ours	ResNet50-half	6.03		✓	64	**2.85**	**7.42**	**6.45**	5.42
Ours	ResNet18	11.24			3840	**2.57**	8.59	7.38	5.78
Ours	ResNet18	11.24		✓	256	**2.71**	**7.23**	**6.30**	5.20
Ours	ResNet18	11.24		✓	128	**2.81**	**7.30**	**6.32**	5.30
Ours	ResNet18	11.24		✓	64	2.89	**7.48**	**6.43**	5.42

Table 5. **Landmark regression.** The error in the percentage of inner-ocular distance (*lower is better*). Results better than DVE's are in **bold**.

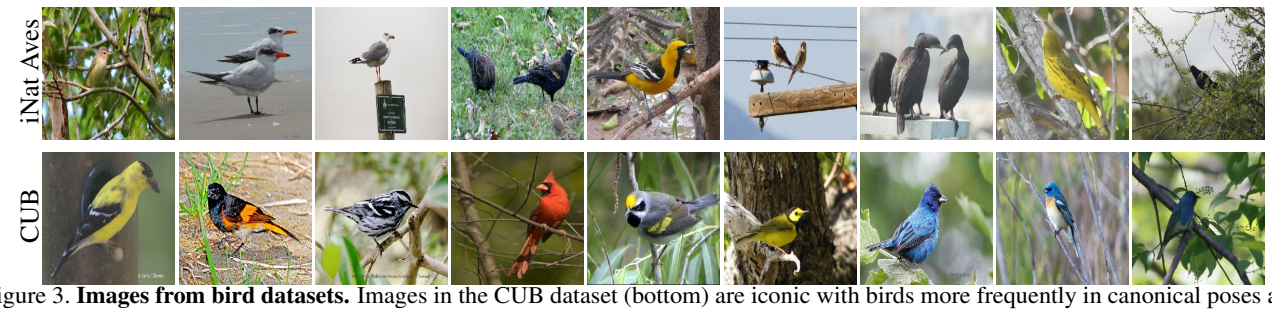

Figure 3. **Images from bird datasets.** Images in the CUB dataset (bottom) are iconic with birds more frequently in canonical poses and contain a single instance. On the other hand, iNaturalist images (top) are community driven and less curated. Often multiple birds are in a single image and are far away. This makes learning and transfer more challenging.

the contrastive model with hypercolumn representation outperforms both the randomly initialized and ImageNet pre-trained models. We are unable to get meaningful results for the ImageNet pre-trained model with fewer than 100 annotations and for a randomly initialized network with the entire dataset. Fine-tuning the network end-to-end outperforms training only the linear layer across different representations. Hypercolumns are more effective than the activations from the fourth convolutional block with fine-tuning and linear evaluation (Tab. 6).

One observation is fine-tuning a randomly initialized network achieves good quantitative performance on this task. A closer inspection, as presented in Figure 4, reveals that this is because the randomly initialized network simply generates a fixed

mask at the center of each test which results in high intersection-over-union with the object mask as most images from the CUB dataset are object-centric (as shown in Fig. 3). Evaluation with a boundary-metric might reveal this difference. Our model on the other hand achieves meaningful and highly accurate masks with as few as 10 training images, as shown in Fig. 4.

Self-supervision	Backbone Fixed?	Hypercolumn	# of annotation				
			10	50	100	250	1241
Random	✓	✓	0.00	0.14	0.01	0.00	0.12
ImageNet	✓	✓	0.12	0.08	0.22	0.51	**0.66**
Contrastive	✓	✓	0.36	**0.52**	**0.59**	**0.61**	0.62
Contrastive	✓	×	**0.37**	0.45	0.51	0.52	0.53
Random	×	✓	0.41	0.40	0.48	0.55	0.71
ImageNet	×	✓	0.38	**0.56**	**0.58**	**0.63**	0.73
Contrastive	×	✓	**0.46**	0.48	0.54	**0.63**	**0.74**
Contrastive	×	×	0.39	0.43	0.45	0.51	0.59

Table 6. **Figure-ground segmentation on CUB dataset.** We report the mean Intersection-over-Union (IoU) performance (*higher is better*) using a ResNet50 network.

Figure 4. **Figure-ground segmentation on CUB dataset with 10 annotated images as training data.** We fine-tune the network end-to-end using the hypercolumn representation.

6. Tables for Figure 4 in the main paper

Tab. 7, 8, and 9 present the numbers corresponding to Fig. 4a, b, and c in the main paper respectively. These describe the effect of dataset size for landmark regression and unsupervised learning.

References

[1] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 3

[2] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 3

[3] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015. 3

[4] James Thewlis, Samuel Albanie, Hakan Bilen, and Andrea Vedaldi. Unsupervised learning of landmarks by descriptor vector exchange. In *ICCV*, 2019. 1, 2, 3, 4, 7

[5] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *ECCV*, 2020. 3

[6] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The iNaturalist species classification and detection dataset. In *CVPR*, 2018. 4

Self-supervision	# of annotations						
	1	5	10	20	50	100	10122
None (SmallNet) [4]	28.87	32.85	22.31	21.13	–	–	14.25
DVE (Hourglass) [4]	14.23 ±1.54	12.04 ±2.03	12.25 ±2.42	11.46 ±0.83	12.76 ±0.53	11.88 ± 0.16	7.53
Ours (ResNet50 + hypercol.)	42.69 ±5.10	25.74 ±2.33	17.61 ±0.75	13.35 ±0.33	10.67 ±0.35	9.24 ± 0.35	6.99
Ours (ResNet50 + conv4)	43.74 ± 2.78	21.25 ±1.14	16.51 ±1.43	12.45 ±0.66	10.03 ±0.21	9.95 ± 0.17	8.05
Ours (ResNet50 + 256D proj.)	28.00 ±1.39	15.85 ±0.86	12.98 ±0.16	11.18 ±0.19	9.56 ±0.44	9.30 ± 0.20	7.17
Ours (ResNet50 + 128D proj.)	27.31 ± 1.39	18.66 ±4.59	13.39 ±0.30	11.77 ±0.85	10.25 ±0.22	9.46 ± 0.05	7.14
Ours (ResNet50 + 64D proj.)	24.87 ± 2.67	15.15 ±0.53	13.62 ±1.08	11.77 ±0.68	11.57 ±0.03	10.06 ± 0.45	7.21
Ours (ResNet18 + hypercol.)	47.15 ± 6.88	24.99 ±3.21	17.40 ±0.37	13.87 ±0.66	11.04 ±0.92	9.93 ±0.39	8.59
Ours (ResNet18 + conv4)	38.05 ±5.25	21.71 ±1.57	16.60 ±0.61	14.48 ±0.69	12.20 ±0.36	11.02 ±0.06	10.61

Table 7. **Landmark regression with limited annotations on AFLW$_M$.** The results are reported as the error in percentage of inter-ocular distance (*lower is better*).

Self-supervision	# of annotation					
	10	50	100	250	500	1241
None (ResNet18)	2.97	10.07	11.31	24.82	38.86	52.64
DVE (Hourglass) [4]	37.82	51.64	54.58	56.78	58.64	61.91
Ours (ResNet18 + hypercol.)	13.41	25.91	34.02	51.70	56.77	62.24
Ours (ResNet50 + hypercol.)	13.87	29.28	40.86	57.96	64.55	68.63
Ours (ResNet50 + 256D proj.)	16.32	38.70	48.75	56.04	57.74	61.22
Ours (ResNet50 + 512D proj.)	17.29	43.90	49.91	57.96	58.93	62.55
Ours (ResNet50 + 1280D proj.)	18.94	47.02	50.75	57.24	59.89	63.25

Table 8. **Landmark regression on bird dataset.** The results are reported as percentage of correct keypoints (PCK). (*higher is better*).

Methods	Dimension	Training set size				
		5%	10%	25%	50%	100%
DVE	64	–	–	–	–	7.53
Ours	3840	13.26	9.12	7.82	7.22	6.99
Ours+proj.	256	8.88	8.20	7.54	7.32	7.17
Ours+proj.	128	9.31	8.50	7.64	7.29	7.14
Ours+proj.	64	9.41	9.69	8.27	7.60	7.21

Table 9. **The effect of training set size on unsupervised learning models.** The results are reported as percentage of inter-ocular distance on AFLW$_M$ benchmark (*lower is better*).