A   Implementation details

We use ResNet-18 as a backbone network. Elsa is pre-trained with the loss objective L_{CL} (Eq. 10) for 500 epochs and fine-tuned with the loss objective L_e (Eq. 13) for 50 epochs. The batch size for the pre-training and fine-tuning step is set to 512 and 64, respectively. τ is set to 0.5. For the optimization, we utilize the LARS optimizer with a learning rate of 0.1 and weight decay 10^{-6} for the pre-training, and utilize the Adam optimizer with a learning rate of 10^{-4} for the fine-tuning step. The number of prototypes is set to 50, and we update prototypes in Elsa every epoch.

In the case of Elsa+, we follow the architecture and training details of CSI [40] for the entire pre-training step. Then, Elsa+ is fine-tuned for 50 epochs. We adopt Adam optimizer with a learning rate of 10^{-4} for the optimization. The batch size is set to 64, and the number of sampling instances for the ensemble technique is set to 10. The number of prototypes is set to 100. We update prototypes for every three epochs.

For the augmentation family $\mathcal{T}_a$, we select the set of transformations from SimCLR [6] in both the pre-training and fine-tuning process (see Table 6). We use rotation by multiples of 90 degrees (e.g., 90°, 180°, 270°, 360°) as shifting transformations $\mathcal{T}_s$. For the strong augmentation, the set of transformations in RandAugment [8] except rotation is excerpted and utilized for the earlystop strategy (see Table 7). We set the number of transformations for RandAugment to 12 ($n = 12$), transform magnitude to 5 ($m = 5$), and modify the probability for applying the transformation from 0.5 to 0.8 to ensure providing anomalies.

Transformation	Parameter	Value
ColorJitter	B,C,s,h,p	0.4, 0.4, 0.4, 0.1, 0.8
RandomGrayScale	P	0.2
RandomResizedCrop	w,h	0.54, 0.54

Table 6: List of transformations used for $\mathcal{T}_a$.

Transformation	Parameter	Range
AutoContrast	-	-
Equalize	-	-
Identity	-	-
Brightness	B	[0.01, 0.99]
Color	C	[0.01, 0.99]
Contrast	C	[0.01, 0.99]
Posterize	B	[1, 8]
Sharpness	S	[0.01, 0.99]
Shear X, Y	R	[-0.3, 0.3]
Solarize	T	[0, 256]
Translate X, Y	λ	[-0.3, 0.3]

Table 7: List of transformations Elsa+ used for strong augmentation. Those are excerpted from RandAugment [8].

B   Algorithm description

In this section, we provide a PyTorch-style pseudo-code of earlystopping, training, and evaluation of Elsa+. Codes and the downloadable link for all trained models would be released upon the acceptance of the paper via GitHub.

```python
"""
Earlystop score algorithm
"""

def earlystop_score(model, prototypes, valid_set):
    aucscores = []
    for images in valid_set:
        batch_size = image.size(0)
        images1, images2 = Weak_aug(images),
            Weak_aug(Strong_aug(images))
        images1 = Concat([Rotate(images1, 90 * d) for d in range(4)])
        images2 = Concat([Rotate(images2, 90 * d) for d in range(4)])
        label_list = [1] * batch_size + [0] * batch_size

        prob1 = calculate_energy(model, images1, prototypes)
        prob2 = calculate_energy(model, images2, prototypes)

        aucscores.append(roc_auc_score(label_list, prob1 + prob2))
    return mean(aucscores)


def calculate_energy(model, images, prototypes):
    z = model(images)
    logits = Matmul(z, prototypes.t())
    logits_list = logits.chunk(4, dim = 0)
    Px_mean = 0
    for shi in range(4):
        Px_mean += log(sum(exp(logits_list[shi])))
    return Px_mean

"""
Fine tune algorithm
T: Temperature
semi_targets: 0,1,-1 for unlabeled, labeled normal, labeled
    anomalies
"""
def train(model, train_set, prototypes)
    for images, semi_targets in train_set:
        images1, images2 = Weak_aug(images), Weak_aug(images)
        images1 = Concat([Rotate(images1, 90 * d) for d in range(4)])
            # 4B
        images2 = Concat([Rotate(images2, 90 * d) for d in range(4)])
            # 4B
        images_pair = Concat([images1, images2]) # 8B
        num_prototypes = prototypes.size(0) # P
```

```
        shift_labels = Concat([Ones_like(semi_targets) * d for d in
            range(4)])
        semi_targets = semi_targets.repeat(8) # 8B

        z = model(images_pair) # 8B x D
        logits = Matmul(z, prototypes.t()) / T # 8B x P

        C = log(num_prototypes + 1/T)
        scores = log(sum(exp(logits))) # 8B

        Le = mean(Where(semi_targets == -1, (C - scores) ** -1,
            scores ** -1))
        Ls = CEloss(z, shift_labels)
        Loss = Le + Ls

        Loss.backward()
        update(model.params)

"""
Test algorithm
n_samples: number of sampling instances for ensembling
"""
def test(model, test_set, prototypes, n_samples)
    z = 0
    scores = []
    total_semi_targets = []

    # semi_targets: 0,1,-1 for unlabeled, labeled normal, labeled
        anomalies
    for images, semi_targets in test_set:
        for seed in range(n_samples): # ensembling
            images_aug = Weak_aug(images)
            images_aug = Concat([Rotate(images_aug, 90 * d) for d in
                range(4)])
            z += model(images_aug)
        z /= num_samples
        logits = Matmul(z, prototypes.t())
        logits_list = logits.chunk(4, dim = 0)

        Px_mean = 0
        for shi in range(4):
            Px_mean += log(sum(exp(logits_list[shi])))
        total_semi_targets.extend(semi_targets)
        scores.extend(Px_mean)
    auroc = roc_auc_score(total_semi_targets, scores)
    return auroc
```

C   Full results

C.1   OOD detection results – CIFAR-10

	Plane	Car	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck	Mean
Plane	-	94.2	89.3	93.5	92.4	91.6	93.8	94.7	90.3	94.3	92.7
Car	98.9	-	98.2	99.0	99.1	98.6	97.7	99.0	98.8	99.1	98.7
Bird	88.9	92.0	-	93.5	91.9	91.9	83.8	92.6	91.4	91.7	90.9
Cat	89.0	87.8	85.2	-	90.3	76.9	77.5	85.9	87.1	87.8	85.3
Deer	95.5	95.5	94.9	96.0	-	93.3	92.3	92.6	94.6	93.6	94.3
Dog	93.3	95.1	92.7	94.4	95.2	-	90.0	92.6	93.8	94.5	93.5
Frog	98.3	98.1	97.6	98.2	98.0	97.7	-	98.1	96.8	97.8	97.8
Horse	97.0	97.5	98.0	98.0	97.3	95.8	93.7	-	97.1	96.6	96.8
Ship	97.0	96.2	96.4	97.4	97.2	97.1	96.0	97.5	-	97.5	96.9
Truck	97.2	95.6	97.0	97.1	96.3	97.1	95.3	95.9	96.2	-	96.4

Table 8: Confusion matrix of Elsa+'s AUROC on one-class classification over CIFAR-10. Each row and column represents normal and anomaly class respectively. ($\gamma_l = 0.01, \gamma_p = 0.00$)

	Plane	Car	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck	Mean
Plane	-	95.0	89.9	94.8	94.2	94.6	93.4	95.0	85.5	94.4	93.0
Car	98.8	-	98.6	98.7	99.0	98.8	97.9	98.8	98.6	99.2	98.7
Bird	89.0	91.8	-	93.7	93.1	92.1	89.2	94.1	90.9	94.4	92.0
Cat	91.0	87.8	89.8	-	90.6	81.6	82.7	90.3	89.2	88.8	88.0
Deer	95.5	95.7	96.1	95.9	-	95.1	93.5	95.8	94.7	95.7	95.3
Dog	93.4	94.8	96.1	93.8	96.1	-	93.7	95.5	94.3	95.3	94.8
Frog	97.8	97.9	98.3	98.4	98.5	98.6	-	98.6	98.0	98.4	98.3
Horse	97.9	97.3	98.5	98.2	98.5	97.4	97.6	-	97.6	97.1	97.8
Ship	97.8	97.8	97.8	97.4	98.0	96.1	97.9	98.0	-	97.4	97.6
Truck	97.3	97.0	97.1	96.6	96.1	97.0	96.0	96.0	95.9	-	96.6

Table 9: Confusion matrix of Elsa+'s AUROC on one-class classification over CIFAR-10. Each row and column represents normal and anomaly class respectively. ($\gamma_l = 0.05, \gamma_p = 0.00$)

	Plane	Car	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck	Mean
Plane	-	95.4	92.2	96.2	94.1	92.3	94.0	95.8	92.1	94.8	94.1
Car	98.8	-	98.8	98.6	99.1	98.7	98.2	98.6	98.8	99.2	98.7
Bird	90.5	93.6	-	93.8	94.3	92.2	90.5	94.2	93.6	93.6	92.9
Cat	90.5	89.7	89.2	-	91.4	84.2	84.3	90.7	90.4	88.5	88.8
Deer	95.7	95.7	96.3	95.6	-	94.9	94.2	96.7	94.7	95.7	95.5
Dog	94.5	94.1	95.9	94.1	96.0	-	93.4	96.2	94.4	94.6	94.8
Frog	98.1	97.7	98.5	98.7	98.5	98.4	-	98.7	98.1	97.8	98.3
Horse	97.4	97.8	98.6	98.4	98.5	97.2	97.7	-	97.3	97.0	97.8
Ship	98.5	97.8	97.7	97.6	98.3	97.2	96.4	98.2	-	97.6	97.7
Truck	97.7	97.0	97.0	96.6	96.7	96.5	96.5	96.3	96.3	-	96.7

Table 10: Confusion matrix of Elsa+'s AUROC on one-class classification over CIFAR-10. Each row and column represents normal and anomaly class respectively. ($\gamma_l = 0.1, \gamma_p = 0.00$)

	Plane	Car	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck	Mean
Plane	-	92.1	88.4	93.8	89.2	91.1	88.9	92.3	80.6	88.0	89.4
Car	97.5	-	97.9	98.2	98.7	97.0	97.7	98.1	95.0	95.8	97.3
Bird	87.2	92.3	-	83.3	84.5	86.1	79.7	92.0	88.4	92.3	87.3
Cat	86.8	88.0	80.7	-	82.8	69.9	68.6	85.9	86.6	86.0	81.7
Deer	93.8	94.7	92.2	92.5	-	91.1	87.6	91.9	93.3	91.9	92.1
Dog	92.2	94.9	89.6	86.2	92.9	-	89.7	92.3	93.9	92.6	91.6
Frog	96.5	97.2	95.7	93.8	95.9	95.5	-	98.2	96.4	97.6	96.3
Horse	96.1	95.5	95.8	94.1	93.3	93.5	93.4	-	94.3	96.0	94.7
Ship	93.6	94.8	96.4	96.7	97.0	96.5	96.7	97.1	-	91.0	95.5
Truck	92.9	87.4	96.6	95.7	95.8	95.4	95.8	96.1	94.5	-	94.5

Table 11: Confusion matrix of Elsa+'s AUROC on one-class classification over CIFAR-10. Each row and column represents normal and anomaly class respectively. ($\gamma_l = 0.01, \gamma_p = 0.05$)

	Plane	Car	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck	Mean
Plane	-	91.3	90.4	93.5	92.6	93.7	90.6	94.0	84.6	92.0	91.4
Car	97.6	-	98.6	98.1	98.9	98.3	98.0	98.7	97.3	96.7	98.0
Bird	88.6	90.3	-	86.5	89.3	83.2	82.2	90.7	90.7	92.4	88.2
Cat	88.7	87.0	86.1	-	88.0	66.4	79.6	85.6	87.0	87.2	84.0
Deer	94.4	94.8	93.7	92.1	-	93.8	90.8	91.0	93.6	94.1	93.1
Dog	94.1	94.8	90.9	79.5	93.1	-	89.9	93.5	93.6	93.8	91.5
Frog	97.5	96.6	93.7	95.6	97.8	95.3	-	98.1	97.6	97.5	96.7
Horse	96.7	96.3	96.7	94.0	96.1	92.4	96.7	-	96.3	96.5	95.8
Ship	91.1	92.7	96.9	97.3	97.6	96.7	95.0	97.8	-	92.6	95.3
Truck	96.0	94.9	97.0	96.3	96.3	96.7	96.1	96.1	94.1	-	96.0

Table 12: Confusion matrix of Elsa+'s AUROC on one-class classification over CIFAR-10. Each row and column represents normal and anomaly class respectively. ($\gamma_l = 0.05, \gamma_p = 0.05$)

	Plane	Car	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck	Mean
Plane	-	91.6	88.6	94.3	90.7	93.6	92.7	94.5	86.0	93.1	91.7
Car	98.4	-	98.8	97.8	98.6	98.1	97.1	98.3	97.8	97.0	98.0
Bird	90.4	91.2	-	89.7	86.3	85.3	84.8	91.6	89.5	92.7	89.0
Cat	89.3	86.0	85.0	-	86.5	70.9	78.3	86.9	88.3	87.0	84.2
Deer	94.9	94.6	89.4	90.4	-	92.8	90.7	93.8	95.1	94.4	92.9
Dog	93.2	93.5	91.7	82.5	92.6	-	88.6	91.6	93.8	93.5	91.2
Frog	97.4	97.0	95.5	92.7	96.4	95.6	-	97.7	97.5	97.7	96.4
Horse	96.4	96.5	95.8	95.3	95.8	94.1	97.2	-	96.0	96.0	95.9
Ship	94.9	96.2	96.3	97.3	97.5	96.8	95.7	97.8	-	93.2	96.2
Truck	97.1	88.5	97.2	96.3	97.1	95.5	96.2	96.2	93.3	-	95.3

Table 13: Confusion matrix of Elsa+'s AUROC on one-class classification over CIFAR-10. Each row and column represents normal and anomaly class respectively. ($\gamma_l = 0.1, \gamma_p = 0.05$)

	Plane	Car	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck	Mean
Plane	-	87.7	88.8	90.9	87.1	86.1	85.9	89.4	78.7	81.4	86.2
Car	97.1	-	98.8	98.2	99.0	98.0	98.0	98.0	95.1	90.5	97.0
Bird	87.8	91.1	-	86.2	81.8	80.2	76.8	85.6	90.2	91.7	85.7
Cat	83.1	79.9	77.3	-	75.3	64.7	68.0	81.4	85.3	85.9	77.9
Deer	93.6	93.2	91.1	84.7	-	88.7	86.4	86.9	93.6	92.7	90.1
Dog	89.0	92.6	88.1	82.2	88.8	-	87.6	92.7	90.8	90.8	89.2
Frog	96.4	96.2	89.0	88.3	89.5	93.4	-	97.9	95.2	97.3	93.7
Horse	96.2	96.3	94.7	93.3	85.5	91.9	94.6	-	96.0	93.8	93.6
Ship	86.9	93.5	95.4	95.3	95.0	95.5	96.2	96.1	-	90.2	93.8
Truck	93.7	84.5	96.4	95.9	95.9	95.9	96.3	95.3	93.2	-	94.1

Table 14: Confusion matrix of Elsa+'s AUROC on one-class classification over CIFAR-10. Each row and column represents normal and anomaly class respectively. ($\gamma_l = 0.01, \gamma_p = 0.1$)

	Plane	Car	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck	Mean
Plane	-	88.3	85.6	92.5	89.8	91.3	91.4	92.6	83.9	92.2	89.7
Car	97.5	-	98.6	97.6	98.7	98.3	98.0	97.2	96.4	94.4	97.4
Bird	83.5	90.6	-	82.6	83.3	79.7	82.1	87.2	87.4	92.2	85.4
Cat	86.8	85.6	83.5	-	80.9	60.9	69.7	86.7	84.8	85.5	80.5
Deer	93.3	94.9	89.7	88.4	-	88.3	86.0	88.4	93.4	93.3	90.6
Dog	93.2	93.5	92.4	80.1	91.0	-	88.0	91.4	92.3	92.2	90.5
Frog	96.5	96.2	93.9	90.7	92.1	90.6	-	97.5	96.5	97.4	94.6
Horse	94.5	96.3	95.9	92.1	91.0	89.2	94.5	-	96.2	95.8	93.9
Ship	93.2	89.6	96.4	96.3	96.9	96.5	95.3	96.5	-	89.9	94.5
Truck	91.7	87.9	95.9	96.6	95.3	96.2	96.2	95.8	89.4	-	93.9

Table 15: Confusion matrix of Elsa+'s AUROC on one-class classification over CIFAR-10. Each row and column represents normal and anomaly class respectively. ($\gamma_l = 0.05, \gamma_p = 0.1$)

	Plane	Car	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck	Mean
Plane	-	90.2	84.4	92.1	91.8	92.4	89.0	92.3	80.7	86.2	88.8
Car	97.8	-	98.5	97.8	98.4	98.4	97.4	98.2	95.4	88.1	96.7
Bird	89.3	91.6	-	86.9	80.3	83.9	79.0	88.2	89.5	91.4	86.7
Cat	86.8	86.4	81.0	-	81.6	61.1	75.3	81.3	87.1	86.7	80.8
Deer	94.4	94.6	92.3	87.9	-	85.8	85.9	91.8	93.7	93.7	91.1
Dog	92.0	93.1	91.0	79.2	89.9	-	88.6	90.9	91.4	91.4	89.7
Frog	97.2	95.4	93.5	93.8	90.4	93.3	-	96.4	96.1	97.4	94.8
Horse	95.3	96.2	96.3	95.1	93.5	91.1	95.2	-	96.4	94.9	94.9
Ship	93.0	92.5	95.6	96.5	96.6	96.4	95.3	97.2	-	89.5	94.7
Truck	92.9	86.7	97.0	94.7	95.9	95.2	96.4	94.9	92.5	-	94.0

Table 16: Confusion matrix of Elsa+'s AUROC on one-class classification over CIFAR-10. Each row and column represents normal and anomaly class respectively. ($\gamma_l = 0.1, \gamma_p = 0.1$)

C.2 OOD detection results – ImageNet-10

We evaluate Elsa+ on ImageNet-10, which is a 10-class subset of the original ImageNet dataset. The total number of ImageNet-10 samples is 13,000, and they are separated into train and test set with a ratio of 10-to-3. To adjust the prototype count according to the sample size (i.e., 1000 images per class), we set a prototype count to 20. Then, the batch size is set to 32, and an Adam optimizer with a learning rate of $5e^{-5}$ is utilized during the fine-tuning process. All other remaining parameter-settings are the same as settings from CIFAR-10 experiments.

Table 17 and Table 18 describe the anomaly detection results over the ImageNet-10 dataset. For both scenarios, Elsa+ achieves comparable performance against CSI. Even though Elsa+ shows a small drop in the detection performance when it is trained on the clean training set, we find that Elsa+ is much robust to the contamination.

	0	1	2	3	4	5	6	7	8	9	Mean
Ours	0.95	0.99	0.99	0.90	0.86	0.89	0.84	0.95	0.90	0.97	0.93
CSI	0.98	0.98	0.99	0.97	0.93	0.99	0.88	0.99	0.96	0.88	0.95

Table 17: One-class classification over ImageNet-10 in the semi-supervised scenario. ($\gamma_l = 0.1, \gamma_p = 0.0$)

	0	1	2	3	4	5	6	7	8	9	Mean
Ours	0.93	0.97	0.99	0.89	0.78	0.89	0.80	0.93	0.85	0.95	0.90
CSI	0.88	0.86	0.83	0.93	0.86	0.97	0.79	0.94	0.92	0.62	0.86

Table 18: One-class classification over ImageNet-10 in the contamination scenario. ($\gamma_l = 0.1, \gamma_p = 0.1$)

C.3   Qualitative analysis

We now examine the inner workings of Elsa+ in detail. Figure 5 shows the energy score distribution at epoch 1 and epoch 2, which shows separation to already begin in epoch 2. At around epoch 30, the two distributions are nearly separated (omitted due to space). The fine-tuning process of Elsa+ separates normal and anomaly samples so that the distribution of normal is sharpened nearby the prototypes and the distribution of anomaly is spread out. Figure 4 shows examples of which training samples are positioned to the same prototypes after training. Images in the same row (i.e., belong to the same prototype) exhibit a common shape that is distinguishable in human eyes. These examples confirm that Elsa+'s flexible and superior embedding effectively explains the distribution of normal samples.
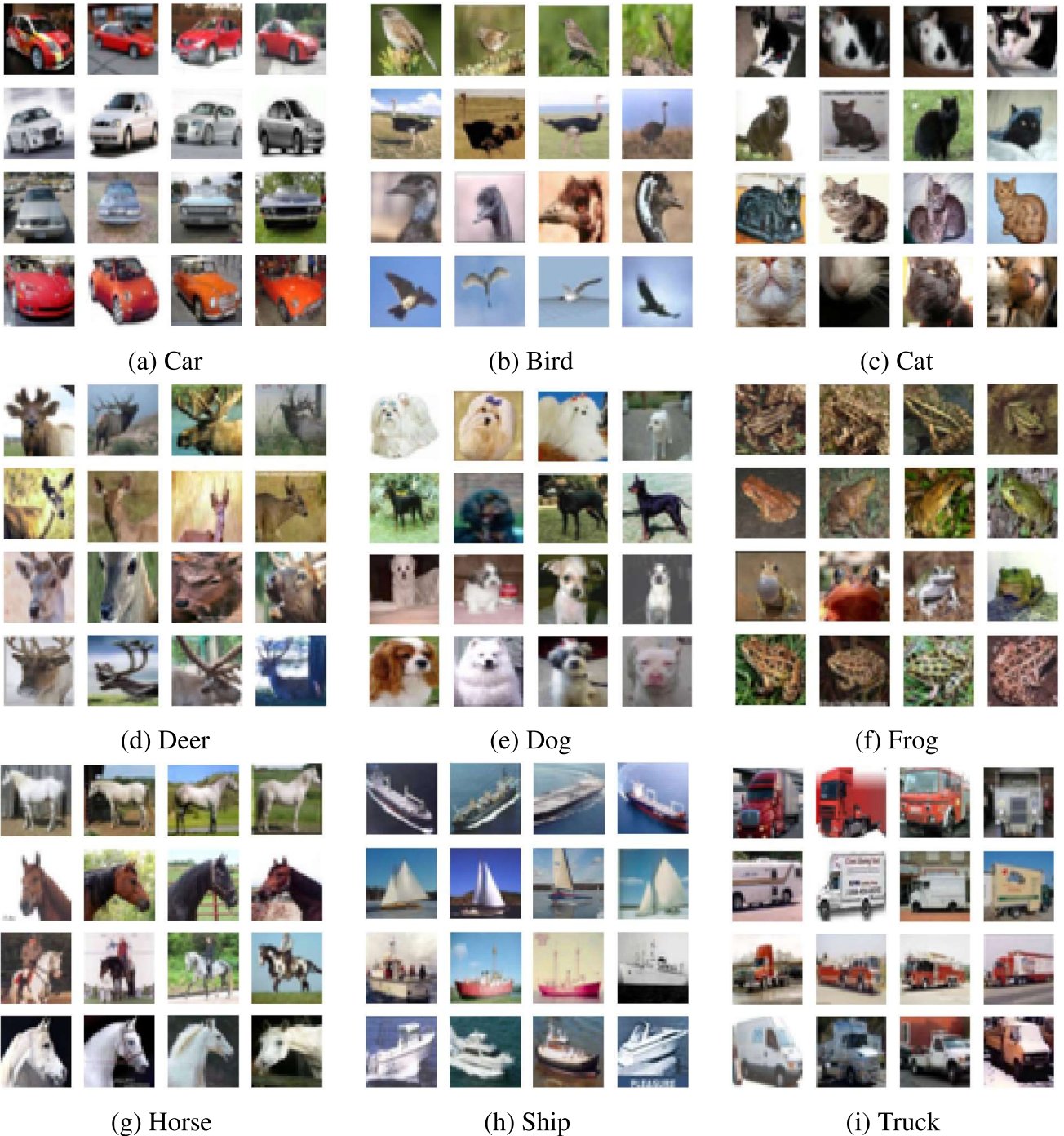

(a) Car   (b) Bird   (c) Cat

(d) Deer   (e) Dog   (f) Frog

(g) Horse   (h) Ship   (i) Truck

Figure 4: Example images sampled from each of prototype classes (class #1 to #9 in CIFAR-10). Images in a row are from the same prototype class.

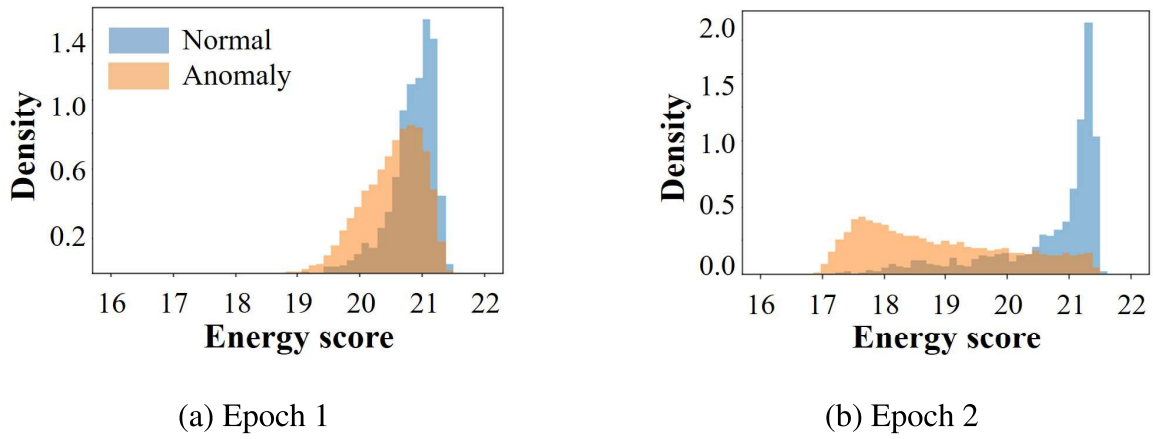(a) Epoch 1   (b) Epoch 2

Figure 5: Histogram of the distributions of scores across training epochs.

C.4   Computational complexity.

We measure the relative computational cost of the model by reporting the elapsed running time in Table 19. Each of the four models is trained with four Tesla V100 GPUs. The table shows that the fine-tuning step in Step-2 and Step-3 increases the model latency marginally, with at most 7% increment compared to the original CSI.

Model	Time
CSI	1.00
Elsa+	×1.07
SimCLR	×0.27
Elsa	×0.31

Table 19: Comparisons of the elapsed time over CIFAR-10.