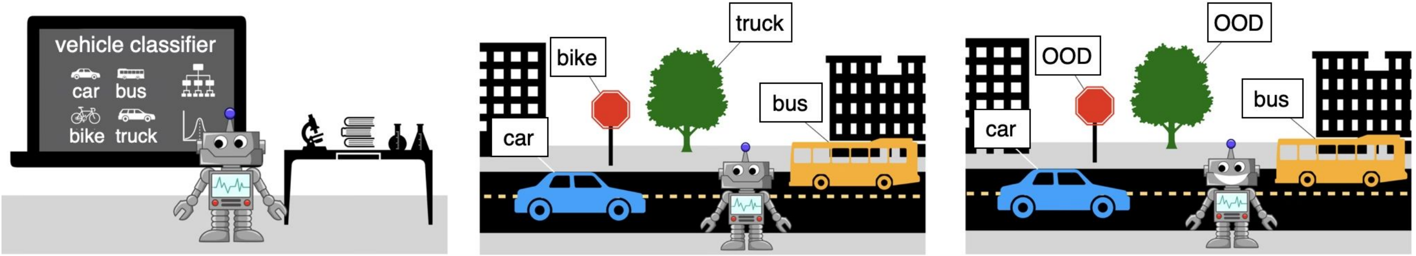
Training OOD Detectors in their Natural Habitats

Julian Katz-Samuels, Julia Nakhleh, Rob Nowak, Yixuan Li

Motivation

- OOD detection is critical for safe deployment of ML models in real-world settings
- ML models deployed in the wild may naturally encounter large quantities of unlabeled data consisting of both ID and OOD examples
- Our work shows that using constrained optimization techniques, this unlabeled “wild” data can be used to train a state-of-the-art OOD detector without sacrificing performance on ID classification



1. Design a classifier

2. Deploy in the wild open world

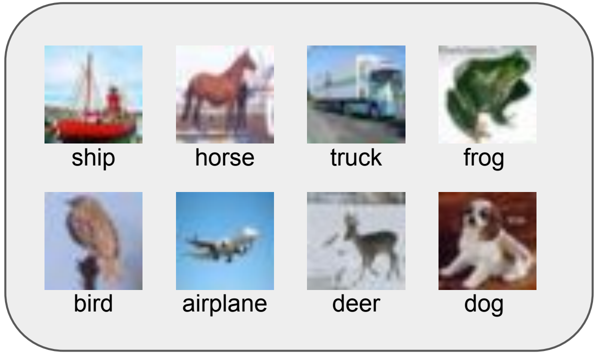
3. Leverage wild data to build classifier and OOD detector

Problem Setup

- Let $\mathbb{P}_{\text{in}}$ and $\mathbb{P}_{\text{out}}$ be two distributions over $\mathbb{R}^d$
- Each in-distribution (ID) sample from $\mathbb{P}_{\text{in}}$ belongs to one of K classes
- When training an OOD detection model, we have access to:

Problem Setup

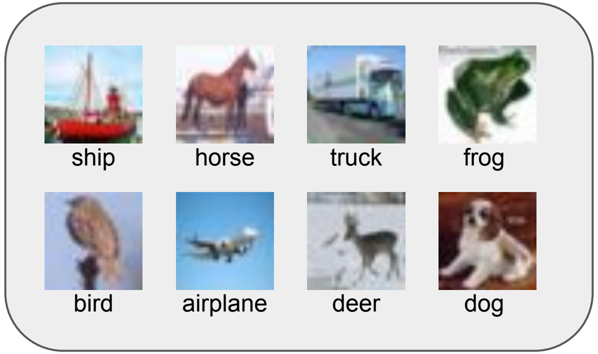
- Let $\mathbb{P}_{\text{in}}$ and $\mathbb{P}_{\text{out}}$ be two distributions over $\mathbb{R}^d$
- Each in-distribution (ID) sample from $\mathbb{P}_{\text{in}}$ belongs to one of K classes
- When training an OOD detection model, we have access to:



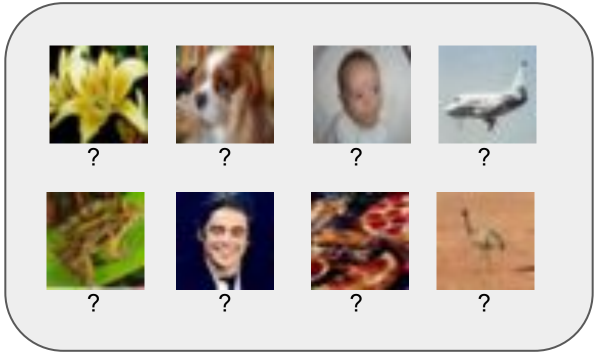
Class-labeled data from $\mathbb{P}_{\text{in}}$

Problem Setup

- Let $\mathbb{P}_{\text{in}}$ and $\mathbb{P}_{\text{out}}$ be two distributions over $\mathbb{R}^d$
- Each in-distribution (ID) sample from $\mathbb{P}_{\text{in}}$ belongs to one of K classes
- When training an OOD detection model, we have access to:



Class-labeled data from $\mathbb{P}_{\text{in}}$



Unlabeled data from $\mathbb{P}_{\text{wild}}$

$$\mathbb{P}_{\text{wild}} := (1-\pi)\mathbb{P}_{\text{in}} + \pi\mathbb{P}_{\text{out}}$$

Learning Objective

$$\begin{aligned}
&\inf_{\theta} \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}\{g_\theta(\tilde{\mathbf{x}}_i) = \text{in}\} \\
\text{s.t. } &\frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{g_\theta(\mathbf{x}_i) = \text{out}\} \le \alpha \\
&\frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{f_\theta(\mathbf{x}_i) \neq y_i)\} \le \tau.
\end{aligned}$$

Learning Objective

$$\inf_{\theta} \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}\{g_\theta(\tilde{\mathbf{x}}_i) = \textbf{in}\}$$

Minimize the proportion of wild samples declared as ID, subject to:

$$\text{s.t. } \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{g_\theta(\mathbf{x}_i) = \textbf{out}\} \leq \alpha$$

$$\frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{f_\theta(\mathbf{x}_i) \neq y_i)\} \leq \tau.$$

Learning Objective

$$\inf_{\theta} \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}\{g_\theta(\tilde{\mathbf{x}}_i) = \text{in}\}$$

Minimize the proportion of wild samples declared as ID, subject to:

$$\text{s.t. } \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{g_\theta(\mathbf{x}_i) = \text{out}\} \leq \alpha$$

No more than 1 - α of the ID samples are declared OOD, and…

$$\frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{f_\theta(\mathbf{x}_i) \neq y_i)\} \leq \tau.$$

Learning Objective

$$\inf_{\theta} \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}\{g_\theta(\tilde{\mathbf{x}}_i) = \textbf{in}\}$$

Minimize the proportion of wild samples declared as ID, subject to:

$$\text{s.t. } \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{g_\theta(\mathbf{x}_i) = \textbf{out}\} \leq \alpha$$

No more than 1 - α of the ID samples are declared OOD, and…

$$\frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{f_\theta(\mathbf{x}_i) \neq y_i)\} \leq \tau.$$

No more than 1 - τ of the ID samples are given the wrong class label.

Learning Objective

$$\inf_{\theta} \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}\{g_\theta(\tilde{\mathbf{x}}_i) = \text{in}\}$$

$$\text{s.t. } \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{g_\theta(\mathbf{x}_i) = \text{out}\} \leq \alpha$$

$$\frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{f_\theta(\mathbf{x}_i) \neq y_i)\} \leq \tau.$$

smooth approx.

$$\text{argmin}_{\theta, w \in \mathbb{R}} \frac{1}{n} \sum_{i=1}^{m} \frac{1}{1 + \exp(-w \cdot E_\theta(\tilde{\mathbf{x}}_i))}$$

$$\text{s.t. } \frac{1}{n} \sum_{j=1}^{n} \frac{1}{1 + \exp(w \cdot E_\theta(\mathbf{x}_i))} \leq \alpha$$

$$\frac{1}{n} \sum_{j=1}^{n} \mathcal{L}_{\text{cls}}(f_\theta(\mathbf{x}_j), y_j) \leq \tau.$$

Learning Objective

Binary-sigmoid loss: distinguish between ID and OOD samples

$$\boxed{\mathcal{L}_{\text{ood}}(g_\theta(\tilde{\mathbf{x}}_i), \text{in}) = \frac{1}{1+\exp(-w \cdot E_\theta(\tilde{\mathbf{x}}_i))}}$$

$$\inf_\theta \frac{1}{m}\sum_{i=1}^{m} \mathbb{1}\{g_\theta(\tilde{\mathbf{x}}_i) = \text{in}\}$$

$$\text{s.t. } \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}\{g_\theta(\mathbf{x}_i) = \text{out}\} \le \alpha$$

$$\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}\{f_\theta(\mathbf{x}_i) \neq y_i)\} \le \tau.$$

smooth approx.

$$\text{argmin}_{\theta, w\in\mathbb{R}} \frac{1}{n}\sum_{i=1}^{m} \boxed{\frac{1}{1+\exp(-w \cdot E_\theta(\tilde{\mathbf{x}}_i))}}$$

$$\text{s.t. } \frac{1}{n}\sum_{j=1}^{n} \frac{1}{1+\exp(w \cdot E_\theta(\mathbf{x}_i))} \le \alpha$$

$$\frac{1}{n}\sum_{j=1}^{n} \mathcal{L}_{\text{cls}}(f_\theta(\mathbf{x}_j), y_j) \le \tau.$$

Learning Objective

Energy-based uncertainty score
(higher for ID samples)

$$E_\theta = \log \sum_{j=1}^{K} e^{f_\theta^{(j)}(\mathbf{x})}$$

Binary-sigmoid loss: distinguish
between ID and OOD samples

$$\mathcal{L}_{\text{ood}}(g_\theta(\tilde{\mathbf{x}}_i), \text{in}) = \frac{1}{1+\exp(-w \cdot \boxed{E_\theta(\tilde{\mathbf{x}}_i)})}$$

$$\inf_\theta \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}\{g_\theta(\tilde{\mathbf{x}}_i) = \text{in}\}$$

$$\text{s.t. } \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{g_\theta(\mathbf{x}_i) = \text{out}\} \leq \alpha$$

$$\frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{f_\theta(\mathbf{x}_i) \neq y_i)\} \leq \tau.$$

smooth approx.

$$\text{argmin}_{\theta, w \in \mathbb{R}} \frac{1}{n} \sum_{i=1}^{m} \boxed{\frac{1}{1+\exp(-w \cdot E_\theta(\tilde{\mathbf{x}}_i))}}$$

$$\text{s.t. } \frac{1}{n} \sum_{j=1}^{n} \frac{1}{1+\exp(w \cdot E_\theta(\mathbf{x}_i))} \leq \alpha$$

$$\frac{1}{n} \sum_{j=1}^{n} \mathcal{L}_{\text{cls}}(f_\theta(\mathbf{x}_j), y_j) \leq \tau.$$

Learning Objective

$$\inf_{\theta} \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}\{g_\theta(\tilde{\mathbf{x}}_i) = \text{in}\}$$

$$\text{s.t. } \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{g_\theta(\mathbf{x}_i) = \text{out}\} \le \alpha$$

$$\frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{f_\theta(\mathbf{x}_i) \ne y_i)\} \le \tau.$$

smooth approx.

Energy-based uncertainty score (higher for ID samples)

$$\boxed{E_\theta = \log \sum_{j=1}^{K} e^{f_\theta^{(j)}(\mathbf{x})}}$$

Binary-sigmoid loss: distinguish between ID and OOD samples

$$\boxed{\mathcal{L}_{\text{ood}}(g_\theta(\tilde{\mathbf{x}}_i), \text{in}) = \frac{1}{1 + \exp(-w \cdot \boxed{E_\theta(\tilde{\mathbf{x}}_i))}}}$$

$$\text{argmin}_{\theta, w \in \mathbb{R}} \frac{1}{n} \sum_{i=1}^{m} \boxed{\frac{1}{1 + \exp(-w \cdot E_\theta(\tilde{\mathbf{x}}_i))}}$$

$$\text{s.t. } \frac{1}{n} \sum_{j=1}^{n} \frac{1}{1 + \exp(w \cdot E_\theta(\mathbf{x}_i))} \le \alpha$$

$$\frac{1}{n} \sum_{j=1}^{n} \boxed{\mathcal{L}_{\text{cls}}(f_\theta(\mathbf{x}_j), y_j) \le \tau.}$$

$$\boxed{\mathcal{L}_{\text{cls}}(f_\theta(\mathbf{x}), y) = -\log \frac{e^{f_\theta^{(y)}(\mathbf{x})}}{\sum_{j=1}^{K} e^{f_\theta^{(j)}(\mathbf{x})}},}$$

Cross-entropy loss: correctly classify ID samples

Augmented Lagrangian Methods (ALM)

- Solve constrained optimization problems of the form:

$$\min_{\theta \in \mathbb{R}^p} f(\theta)$$
$$\text{s.t. } c_i(\theta) \leq 0 \, \forall i \in [q],$$

  as a sequence of unconstrained optimization problems.

- Define the classical augmented Lagrangian function:

$$\mathcal{L}_\beta(\theta, \lambda) = f(\theta) + \sum_{i=1}^{q} \psi_\beta(c_i(\theta), \lambda_i) \;, \quad \text{where} \qquad \psi_\beta(u, v) = \begin{cases} uv + \frac{\beta}{2}u^2 & \beta u + v \geq 0 \\ -\frac{v^2}{2\beta} & \text{o/w} \end{cases}$$

- At iteration k, ALM minimizes $\mathcal{L}_\beta$ w.r.t. θ and then performs the gradient ascent update:

$$1. \;\; \theta^{(k+1)} \longleftarrow \text{argmin}_\theta \mathcal{L}_{\beta_k}(\theta, \lambda^{(k)})$$

$$2. \;\; \lambda^{(k+1)} \longleftarrow \lambda^{(k)} + \rho \nabla_\lambda \mathcal{L}_{\beta_k}(\theta^{(k+1)}, \lambda)$$

Augmented Lagrangian Methods (ALM)

- Solve constrained optimization problems of the form:

$$\min_{\theta \in \mathbb{R}^p} f(\theta)$$

convex

$$\text{s.t. } c_i(\theta) \le 0 \, \forall i \in [q],$$

as a sequence of unconstrained optimization problems.

- Define the classical augmented Lagrangian function:

$$\mathcal{L}_\beta(\theta, \lambda) = f(\theta) + \sum_{i=1}^{q} \psi_\beta(c_i(\theta), \lambda_i) \ , \quad \text{where} \quad \psi_\beta(u, v) = \begin{cases} uv + \frac{\beta}{2}u^2 & \beta u + v \ge 0 \\ -\frac{v^2}{2\beta} & \text{o/w} \end{cases}$$

- At iteration k, ALM minimizes $\mathcal{L}_\beta$ w.r.t. θ and then performs the gradient ascent update:

1. $\theta^{(k+1)} \longleftarrow \text{argmin}_\theta \mathcal{L}_{\beta_k}(\theta, \lambda^{(k)})$
2. $\lambda^{(k+1)} \longleftarrow \lambda^{(k)} + \rho \nabla_\lambda \mathcal{L}_{\beta_k}(\theta^{(k+1)}, \lambda)$

Augmented Lagrangian Methods (ALM)

- Solve constrained optimization problems of the form:

convex

$$\min_{\theta \in \mathbb{R}^p} f(\theta)$$
$$\text{s.t. } c_i(\theta) \le 0 \, \forall i \in [q],$$

as a sequence of unconstrained optimization problems.

$\beta > 0$

- Define the classical augmented Lagrangian function:

$$\mathcal{L}_\beta(\theta, \lambda) = f(\theta) + \sum_{i=1}^{q} \psi_\beta(c_i(\theta), \lambda_i) \ , \quad \text{where} \quad \psi_\beta(u, v) = \begin{cases} uv + \frac{\beta}{2}u^2 & \beta u + v \ge 0 \\ -\frac{v^2}{2\beta} & \text{o/w} \end{cases}$$

$\lambda = (\lambda_1, \ldots, \lambda_q)^\top$

- At iteration k, ALM minimizes $\mathcal{L}_\beta$ w.r.t. θ and then performs the gradient ascent update:

1. $\theta^{(k+1)} \longleftarrow \text{argmin}_\theta \mathcal{L}_{\beta_k}(\theta, \lambda^{(k)})$

2. $\lambda^{(k+1)} \longleftarrow \lambda^{(k)} + \rho \nabla_\lambda \mathcal{L}_{\beta_k}(\theta^{(k+1)}, \lambda)$

Augmented Lagrangian Methods (ALM)

- Solve constrained optimization problems of the form:

convex

$$\min_{\theta \in \mathbb{R}^p} f(\theta)$$
$$\text{s.t. } c_i(\theta) \le 0 \, \forall i \in [q],$$

as a sequence of unconstrained optimization problems.

- Define the classical augmented Lagrangian function:

$\beta > 0$

$$\mathcal{L}_\beta(\theta, \lambda) = f(\theta) + \sum_{i=1}^{q} \psi_\beta(c_i(\theta), \lambda_i) \ , \quad \text{where} \quad \psi_\beta(u, v) = \begin{cases} uv + \frac{\beta}{2}u^2 & \beta u + v \ge 0 \\ -\frac{v^2}{2\beta} & \text{o/w} \end{cases}$$

$\lambda = (\lambda_1, \ldots, \lambda_q)^\top$

- At iteration k, ALM minimizes $\mathcal{L}_\beta$ w.r.t. θ and then performs the gradient ascent update:

learning rate

1. $\theta^{(k+1)} \longleftarrow \text{argmin}_\theta \mathcal{L}_{\beta_k}(\theta, \lambda^{(k)})$

penalty parameter (fixed beforehand or adapted during training)

2. $\lambda^{(k+1)} \longleftarrow \lambda^{(k)} + \rho \nabla_\lambda \mathcal{L}_{\beta_k}(\theta^{(k+1)}, \lambda)$

Implementing ALM with neural networks

Algorithm 1 WOODS (Wild OOD detection sans-Supervision)

1: **Input:** $\theta^{(1)}_{(1)}$, $\lambda^{(1)}_{(1)}$ β_1, β_2, epoch length T, batch size B, learning rate μ_1, learning rate μ_2, penalty multiplier γ, `tol`
2: **for** epoch $= 1, 2, \ldots$ **do**
3: **for** $t = 1, 2, \ldots, T-1$ **do**
4: Sample a batch of data, calculate $\mathcal{L}^{\text{batch}}_{\beta}(\theta, \lambda)$
5: $\theta^{(t+1)}_{(\text{epoch})} \longleftarrow \theta^{(t)}_{(\text{epoch})} - \mu_1 \nabla_\theta \mathcal{L}^{\text{batch}}_{\beta}(\theta, \lambda)$
6: **end for**
7: $\lambda^{(\text{epoch}+1)} \leftarrow \lambda^{(\text{epoch})} + \mu_2 \nabla_\theta \mathcal{L}_\beta(\theta^{(T)}_{(\text{epoch})}, \lambda^{(\text{epoch})})$
8: **if** $\frac{1}{n}\sum_{i=1}^{n} \mathcal{L}_{\text{ood}}(g_{\theta^{(T)}_{(\text{epoch})}}(\mathbf{x}_i), \text{out}) > \alpha + \texttt{tol}$ **then**
9: $\beta_1 \longleftarrow \gamma \beta_1$
10: **end if**
11: **if** $\frac{1}{n}\sum_{i=1}^{n} \mathcal{L}_{\text{cls}}(f_{\theta^{(T)}_{(\text{epoch})}}(\mathbf{x}_i), y_i) > \tau + \texttt{tol}$ **then**
12: $\beta_2 \longleftarrow \gamma \beta_2$
13: **end if**
14: $\theta^{(1)}_{(\text{epoch}+1)} \longleftarrow \theta^{(T)}_{(\text{epoch})}$
15: **end for**

Overview of our training procedure.

$$\mathcal{L}^{\text{batch}}_{\beta}(\theta, \lambda) = \frac{1}{B}\sum_{i \in I} \mathcal{L}_{\text{ood}}(g_\theta(\tilde{\mathbf{x}}_i), \text{in}) + \psi_{\beta_1}\Big(\frac{1}{B}\sum_{j \in J} \mathcal{L}_{\text{ood}}(g_\theta(\mathbf{x}_j), \text{out}) - \alpha, \lambda_1^{(\text{epoch})}\Big) + \psi_{\beta_2}\Big(\frac{1}{B}\sum_{j \in J} \mathcal{L}_{\text{cls}}(f_\theta(\mathbf{x}_j), y_j) - \tau, \lambda_2^{(\text{epoch})}\Big)],$$

$$\psi_\beta(u, v) = \begin{cases} uv + \frac{\beta}{2}u^2 & \beta u + v \geq 0 \\ -\frac{v^2}{2\beta} & \text{o/w} \end{cases}$$

I and J are mini-batches of size B sampled randomly from the wild and ID data, respectively.

Because ψ is convex in u, the function $\mathcal{L}^{\text{batch}}_{\beta}$ is an upper bound on $\mathcal{L}_\beta$ at each epoch (via Jensen's inequality).

Experimental setup

- ID datasets: CIFAR-10 and CIFAR-100
- OOD datasets: SVHN, Textures, Places, LSUN-Crop, LSUN-Resize, and 300K Random Images (cleaned subset of 80 Million TinyImages)
- Models are initialized using a WideResNet architecture pre-trained on CIFAR-10/100 and trained for 100 epochs
    - Architecture: 40 layers, widen factor = 2, weight decay = 0.0005, momentum = 0.09
    - Optimization: SGD with Nesterov momentum
- Metrics: FPR@95, AUROC, accuracy (on ID classification)

Main results (CIFAR-100)

Method	OOD Dataset: SVHN		LSUN-R		LSUN-C		Textures		Places365		Average		Acc.
	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	
With $\mathbb{P}_{\text{in}}$ only													
MSP	84.59	71.44	82.42	75.38	66.54	83.79	83.29	73.34	82.84	73.78	79.94	75.55	**75.96**
ODIN	84.66	67.26	71.96	81.82	55.55	87.73	79.27	73.45	87.88	71.63	75.86	76.38	**75.96**
Energy	85.82	73.99	79.47	79.23	35.32	93.53	79.41	76.28	80.56	75.44	72.12	79.69	**75.96**
Mahalanobis	57.52	86.01	21.23	96.00	91.18	69.69	39.39	90.57	88.83	67.87	59.63	82.03	**75.96**
GODIN	83.38	84.05	62.24	88.22	72.86	83.84	83.83	78.91	80.56	76.14	76.57	82.23	75.33
CSI	64.70	84.97	91.55	63.42	38.10	92.52	74.70	92.66	82.25	73.63	70.26	81.44	69.90
With $\mathbb{P}_{\text{in}}$ and $\mathbb{P}_{\text{wild}}$													
OE	$1.57^{\pm0.1}$	$99.63^{\pm0.0}$	$0.93^{\pm0.2}$	$99.79^{\pm0.0}$	$3.83^{\pm0.4}$	$99.26^{\pm0.1}$	$27.89^{\pm0.5}$	$93.35^{\pm0.2}$	$60.24^{\pm0.6}$	$83.43^{\pm0.6}$	$18.89^{\pm0.4}$	$95.09^{\pm0.2}$	$71.65^{\pm0.4}$
Energy (w/ OE)	$1.47^{\pm0.3}$	$99.68^{\pm0.0}$	$2.68^{\pm1.9}$	$99.50^{\pm0.3}$	$2.52^{\pm0.4}$	$99.44^{\pm0.1}$	$37.26^{\pm9.1}$	$91.26^{\pm2.5}$	$54.67^{\pm1.0}$	$86.09^{\pm0.4}$	$19.72^{\pm2.5}$	$95.19^{\pm0.7}$	$73.46^{\pm0.8}$
WOODS (ours)	$0.52^{\pm0.1}$	$99.88^{\pm0.0}$	$0.38^{\pm0.1}$	$99.92^{\pm0.0}$	$0.93^{\pm0.2}$	$99.77^{\pm0.0}$	$17.92^{\pm0.5}$	$96.44^{\pm0.2}$	$37.90^{\pm0.6}$	**91.22**$^{\pm0.3}$	$11.53^{\pm0.3}$	$97.45^{\pm0.1}$	$74.79^{\pm0.2}$
WOODS-alt (ours)	**0.12**$^{\pm0.0}$	**99.96**$^{\pm0.0}$	**0.07**$^{\pm0.1}$	**99.96**$^{\pm0.0}$	**0.11**$^{\pm0.0}$	**99.96**$^{\pm0.0}$	**9.12**$^{\pm0.3}$	**96.65**$^{\pm0.1}$	**29.58**$^{\pm0.4}$	$90.60^{\pm0.3}$	**7.80**$^{\pm0.5}$	**97.43**$^{\pm0.5}$	**75.22**$^{\pm0.2}$

Table 1. **Main results when $\mathbb{P}_{\text{out}}^{\text{test}} = \mathbb{P}_{\text{out}}$.** Comparison with competitive OOD detection methods on `CIFAR-100`. For methods using $\mathbb{P}_{\text{wild}}$, we train under the same dataset and same $\pi = 0.1$. For each dataset, we create corresponding wild mixture distribution $\mathbb{P}_{\text{wild}} := (1-\pi)\mathbb{P}_{\text{in}} + \pi\mathbb{P}_{\text{out}}$ for training and test on the corresponding OOD dataset. ↑ indicates larger values are better and vice versa. $\pm x$ denotes the standard error, rounded to the first decimal point.

Main results (CIFAR-100)

- 48%
(avg
FPR)

Method	OOD Dataset: SVHN		LSUN-R		LSUN-C		Textures		Places365		Average		Acc.
	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	
With $\mathbb{P}_{\text{in}}$ only													
MSP	84.59	71.44	82.42	75.38	66.54	83.79	83.29	73.34	82.84	73.78	79.94	75.55	**75.96**
ODIN	84.66	67.26	71.96	81.82	55.55	87.73	79.27	73.45	87.88	71.63	75.86	76.38	**75.96**
Energy	85.82	73.99	79.47	79.23	35.32	93.53	79.41	76.28	80.56	75.44	72.12	79.69	**75.96**
Mahalanobis	57.52	86.01	21.23	96.00	91.18	69.69	39.39	90.57	88.83	67.87	59.63	82.03	**75.96**
GODIN	83.38	84.05	62.24	88.22	72.86	83.84	83.83	78.91	80.56	76.14	76.57	82.23	75.33
CSI	64.70	84.97	91.55	63.42	38.10	92.52	74.70	92.66	82.25	73.63	70.26	81.44	69.90
With $\mathbb{P}_{\text{in}}$ and $\mathbb{P}_{\text{wild}}$													
OE	$1.57^{\pm 0.1}$	$99.63^{\pm 0.0}$	$0.93^{\pm 0.2}$	$99.79^{\pm 0.0}$	$3.83^{\pm 0.4}$	$99.26^{\pm 0.1}$	$27.89^{\pm 0.5}$	$93.35^{\pm 0.2}$	$60.24^{\pm 0.6}$	$83.43^{\pm 0.6}$	$18.89^{\pm 0.4}$	$95.09^{\pm 0.2}$	$71.65^{\pm 0.4}$
Energy (w/ OE)	$1.47^{\pm 0.3}$	$99.68^{\pm 0.0}$	$2.68^{\pm 1.9}$	$99.50^{\pm 0.3}$	$2.52^{\pm 0.4}$	$99.44^{\pm 0.1}$	$37.26^{\pm 9.1}$	$91.26^{\pm 2.5}$	$54.67^{\pm 1.0}$	$86.09^{\pm 0.4}$	$19.72^{\pm 2.5}$	$95.19^{\pm 0.7}$	$73.46^{\pm 0.8}$
WOODS (ours)	$0.52^{\pm 0.1}$	$99.88^{\pm 0.0}$	$0.38^{\pm 0.1}$	$99.92^{\pm 0.0}$	$0.93^{\pm 0.2}$	$99.77^{\pm 0.0}$	$17.92^{\pm 0.5}$	$96.44^{\pm 0.2}$	$37.90^{\pm 0.6}$	$\mathbf{91.22}^{\pm 0.3}$	$11.53^{\pm 0.3}$	$97.45^{\pm 0.1}$	$74.79^{\pm 0.2}$
WOODS-alt (ours)	$\mathbf{0.12}^{\pm 0.0}$	$\mathbf{99.96}^{\pm 0.0}$	$\mathbf{0.07}^{\pm 0.1}$	$\mathbf{99.96}^{\pm 0.0}$	$\mathbf{0.11}^{\pm 0.0}$	$\mathbf{99.96}^{\pm 0.0}$	$\mathbf{9.12}^{\pm 0.3}$	$\mathbf{96.65}^{\pm 0.1}$	$\mathbf{29.58}^{\pm 0.4}$	$90.60^{\pm 0.3}$	$\mathbf{7.80}^{\pm 0.5}$	$\mathbf{97.43}^{\pm 0.5}$	$\mathbf{75.22}^{\pm 0.2}$

Table 1. **Main results when $\mathbb{P}_{\text{out}}^{\text{test}} = \mathbb{P}_{\text{out}}$.** Comparison with competitive OOD detection methods on `CIFAR-100`. For methods using $\mathbb{P}_{\text{wild}}$, we train under the same dataset and same $\pi = 0.1$. For each dataset, we create corresponding wild mixture distribution $\mathbb{P}_{\text{wild}} := (1-\pi)\mathbb{P}_{\text{in}} + \pi\mathbb{P}_{\text{out}}$ for training and test on the corresponding OOD dataset. ↑ indicates larger values are better and vice versa. $\pm x$ denotes the standard error, rounded to the first decimal point.

Main results (CIFAR-100)

- 48% (avg FPR)

- 7.3% (avg FPR)

Method	SVHN FPR↓	SVHN AUROC↑	LSUN-R FPR↓	LSUN-R AUROC↑	LSUN-C FPR↓	LSUN-C AUROC↑	Textures FPR↓	Textures AUROC↑	Places365 FPR↓	Places365 AUROC↑	Average FPR↓	Average AUROC↑	Acc.
With $\mathbb{P}_{\text{in}}$ only													
MSP	84.59	71.44	82.42	75.38	66.54	83.79	83.29	73.34	82.84	73.78	79.94	75.55	**75.96**
ODIN	84.66	67.26	71.96	81.82	55.55	87.73	79.27	73.45	87.88	71.63	75.86	76.38	**75.96**
Energy	85.82	73.99	79.47	79.23	35.32	93.53	79.41	76.28	80.56	75.44	72.12	79.69	**75.96**
Mahalanobis	57.52	86.01	21.23	96.00	91.18	69.69	39.39	90.57	88.83	67.87	59.63	82.03	**75.96**
GODIN	83.38	84.05	62.24	88.22	72.86	83.84	83.83	78.91	80.56	76.14	76.57	82.23	75.33
CSI	64.70	84.97	91.55	63.42	38.10	92.52	74.70	92.66	82.25	73.63	70.26	81.44	69.90
With $\mathbb{P}_{\text{in}}$ and $\mathbb{P}_{\text{wild}}$													
OE	1.57$^{\pm0.1}$	99.63$^{\pm0.0}$	0.93$^{\pm0.2}$	99.79$^{\pm0.0}$	3.83$^{\pm0.4}$	99.26$^{\pm0.1}$	27.89$^{\pm0.5}$	93.35$^{\pm0.2}$	60.24$^{\pm0.6}$	83.43$^{\pm0.6}$	18.89$^{\pm0.4}$	95.09$^{\pm0.2}$	71.65$^{\pm0.4}$
Energy (w/ OE)	1.47$^{\pm0.3}$	99.68$^{\pm0.0}$	2.68$^{\pm1.9}$	99.50$^{\pm0.3}$	2.52$^{\pm0.4}$	99.44$^{\pm0.1}$	37.26$^{\pm9.1}$	91.26$^{\pm2.5}$	54.67$^{\pm1.0}$	86.09$^{\pm0.4}$	19.72$^{\pm2.5}$	95.19$^{\pm0.7}$	73.46$^{\pm0.8}$
WOODS (ours)	0.52$^{\pm0.1}$	99.88$^{\pm0.0}$	0.38$^{\pm0.1}$	99.92$^{\pm0.0}$	0.93$^{\pm0.2}$	99.77$^{\pm0.0}$	17.92$^{\pm0.5}$	96.44$^{\pm0.2}$	37.90$^{\pm0.6}$	**91.22**$^{\pm0.3}$	11.53$^{\pm0.3}$	97.45$^{\pm0.1}$	74.79$^{\pm0.2}$
WOODS-alt (ours)	**0.12**$^{\pm0.0}$	**99.96**$^{\pm0.0}$	**0.07**$^{\pm0.1}$	**99.96**$^{\pm0.0}$	**0.11**$^{\pm0.0}$	**99.96**$^{\pm0.0}$	**9.12**$^{\pm0.3}$	**96.65**$^{\pm0.1}$	**29.58**$^{\pm0.4}$	90.60$^{\pm0.3}$	**7.80**$^{\pm0.5}$	**97.43**$^{\pm0.5}$	**75.22**$^{\pm0.2}$

Table 1. **Main results when $\mathbb{P}_{\text{out}}^{\text{test}} = \mathbb{P}_{\text{out}}$.** Comparison with competitive OOD detection methods on CIFAR-100. For methods using $\mathbb{P}_{\text{wild}}$, we train under the same dataset and same $\pi = 0.1$. For each dataset, we create corresponding wild mixture distribution $\mathbb{P}_{\text{wild}} := (1-\pi)\mathbb{P}_{\text{in}} + \pi\mathbb{P}_{\text{out}}$ for training and test on the corresponding OOD dataset. ↑ indicates larger values are better and vice versa. $\pm x$ denotes the standard error, rounded to the first decimal point.

Main results (CIFAR-100)

Method	SVHN		LSUN-R		LSUN-C		Textures		Places365		Average		Acc.
	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	
With $\mathbb{P}_{\text{in}}$ only													
MSP	84.59	71.44	82.42	75.38	66.54	83.79	83.29	73.34	82.84	73.78	79.94	75.55	**75.96**
ODIN	84.66	67.26	71.96	81.82	55.55	87.73	79.27	73.45	87.88	71.63	75.86	76.38	**75.96**
Energy	85.82	73.99	79.47	79.23	35.32	93.53	79.41	76.28	80.56	75.44	72.12	79.69	**75.96**
Mahalanobis	57.52	86.01	21.23	96.00	91.18	69.69	39.39	90.57	88.83	67.87	$\boxed{59.63}$	82.03	**75.96**
GODIN	83.38	84.05	62.24	88.22	72.86	83.84	83.83	78.91	80.56	76.14	76.57	82.23	75.33
CSI	64.70	84.97	91.55	63.42	38.10	92.52	74.70	92.66	82.25	73.63	70.26	81.44	69.90
With $\mathbb{P}_{\text{in}}$ and $\mathbb{P}_{\text{wild}}$													
OE	$1.57^{\pm 0.1}$	$99.63^{\pm 0.0}$	$0.93^{\pm 0.2}$	$99.79^{\pm 0.0}$	$3.83^{\pm 0.4}$	$99.26^{\pm 0.1}$	$27.89^{\pm 0.5}$	$93.35^{\pm 0.2}$	$60.24^{\pm 0.6}$	$83.43^{\pm 0.6}$	$\boxed{18.89^{\pm 0.4}}$	$95.09^{\pm 0.2}$	$71.65^{\pm 0.4}$
Energy (w/ OE)	$1.47^{\pm 0.3}$	$99.68^{\pm 0.0}$	$2.68^{\pm 1.9}$	$99.50^{\pm 0.3}$	$2.52^{\pm 0.4}$	$99.44^{\pm 0.1}$	$37.26^{\pm 9.1}$	$91.26^{\pm 2.5}$	$54.67^{\pm 1.0}$	$86.09^{\pm 0.4}$	$19.72^{\pm 2.5}$	$95.19^{\pm 0.7}$	$73.46^{\pm 0.8}$
WOODS (ours)	$0.52^{\pm 0.1}$	$99.88^{\pm 0.0}$	$0.38^{\pm 0.1}$	$99.92^{\pm 0.0}$	$0.93^{\pm 0.2}$	$99.77^{\pm 0.0}$	$17.92^{\pm 0.5}$	$96.44^{\pm 0.2}$	$37.90^{\pm 0.6}$	$\mathbf{91.22}^{\pm 0.3}$	$\boxed{11.53^{\pm 0.3}}$	$97.45^{\pm 0.1}$	$74.79^{\pm 0.2}$
WOODS-alt (ours)	$\mathbf{0.12}^{\pm 0.0}$	$\mathbf{99.96}^{\pm 0.0}$	$\mathbf{0.07}^{\pm 0.1}$	$\mathbf{99.96}^{\pm 0.0}$	$\mathbf{0.11}^{\pm 0.0}$	$\mathbf{99.96}^{\pm 0.0}$	$\mathbf{9.12}^{\pm 0.3}$	$\mathbf{96.65}^{\pm 0.1}$	$\mathbf{29.58}^{\pm 0.4}$	$90.60^{\pm 0.3}$	$\mathbf{7.80}^{\pm 0.5}$	$\mathbf{97.43}^{\pm 0.5}$	$\mathbf{75.22}^{\pm 0.2}$

- 48% (avg FPR)

- 7.3% (avg FPR)

Table 1. **Main results when $\mathbb{P}_{\text{out}}^{\text{test}} = \mathbb{P}_{\text{out}}$.** Comparison with competitive OOD detection methods on CIFAR-100. For methods using $\mathbb{P}_{\text{wild}}$, we train under the same dataset and same $\pi = 0.1$. For each dataset, we create corresponding wild mixture distribution $\mathbb{P}_{\text{wild}} := (1-\pi)\mathbb{P}_{\text{in}} + \pi\mathbb{P}_{\text{out}}$ for training and test on the corresponding OOD dataset. ↑ indicates larger values are better and vice versa. $\pm x$ denotes the standard error, rounded to the first decimal point.

Ablation on π

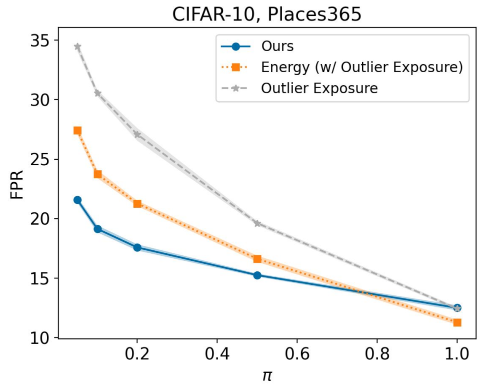
Method	OOD Dataset												
	SVHN		LSUN-R		LSUN-C		Textures		Places365		300K Rand. Img.		Acc.
	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	
$\pi = 0.05$													
OE	80.21$^{\pm1.7}$	77.47$^{\pm1.8}$	77.97$^{\pm2.3}$	78.68$^{\pm1.7}$	61.27$^{\pm1.4}$	86.27$^{\pm0.4}$	77.15$^{\pm1.2}$	77.94$^{\pm0.5}$	80.24$^{\pm0.3}$	74.86$^{\pm0.2}$	75.33$^{\pm0.3}$	77.16$^{\pm0.3}$	73.63$^{\pm0.3}$
Energy (w/ OE)	77.47$^{\pm2.0}$	80.48$^{\pm1.2}$	70.83$^{\pm3.1}$	82.86$^{\pm2.0}$	29.42$^{\pm4.3}$	94.61$^{\pm0.8}$	72.05$^{\pm0.8}$	80.73$^{\pm0.5}$	74.69$^{\pm0.6}$	78.60$^{\pm0.4}$	66.91$^{\pm0.7}$	80.44$^{\pm0.5}$	75.77$^{\pm0.1}$
WOODS (ours)	**74.54**$^{\pm1.7}$	**82.01**$^{\pm1.3}$	**66.29**$^{\pm3.9}$	**84.46**$^{\pm2.3}$	**19.07**$^{\pm1.6}$	**96.48**$^{\pm0.3}$	**65.75**$^{\pm0.6}$	**83.71**$^{\pm0.2}$	**69.97**$^{\pm1.1}$	**80.82**$^{\pm0.5}$	**62.48**$^{\pm1.1}$	**82.92**$^{\pm0.5}$	**75.92**$^{\pm0.1}$
$\pi = 0.1$													
OE	79.56$^{\pm1.6}$	77.00$^{\pm1.2}$	76.86$^{\pm2.1}$	78.75$^{\pm1.2}$	58.53$^{\pm2.8}$	86.92$^{\pm0.8}$	74.63$^{\pm1.2}$	79.13$^{\pm0.5}$	78.52$^{\pm0.3}$	75.68$^{\pm0.1}$	72.18$^{\pm0.2}$	78.48$^{\pm0.3}$	73.53$^{\pm0.4}$
Energy (w/ OE)	77.45$^{\pm2.1}$	80.94$^{\pm1.4}$	67.13$^{\pm3.6}$	83.68$^{\pm2.4}$	27.08$^{\pm2.1}$	94.97$^{\pm0.4}$	70.15$^{\pm1.0}$	81.59$^{\pm0.6}$	71.71$^{\pm1.1}$	79.89$^{\pm0.6}$	64.24$^{\pm2.3}$	82.28$^{\pm1.1}$	75.27$^{\pm0.2}$
WOODS (ours)	**71.67**$^{\pm1.9}$	**84.11**$^{\pm1.4}$	**59.27**$^{\pm3.9}$	**86.80**$^{\pm1.9}$	**15.03**$^{\pm1.4}$	**97.24**$^{\pm0.3}$	**61.38**$^{\pm0.7}$	**85.57**$^{\pm0.2}$	**64.19**$^{\pm1.0}$	**83.12**$^{\pm0.5}$	**55.51**$^{\pm1.3}$	**85.72**$^{\pm0.4}$	**75.64**$^{\pm0.3}$
$\pi = 0.2$													
OE	72.59$^{\pm3.9}$	81.38$^{\pm1.9}$	65.04$^{\pm3.8}$	82.65$^{\pm1.8}$	48.62$^{\pm3.1}$	89.52$^{\pm0.8}$	65.95$^{\pm1.2}$	82.43$^{\pm0.3}$	71.29$^{\pm0.7}$	78.71$^{\pm0.4}$	65.40$^{\pm0.8}$	81.99$^{\pm0.1}$	72.89$^{\pm0.3}$
Energy (w/ OE)	72.76$^{\pm2.5}$	83.48$^{\pm1.2}$	62.53$^{\pm5.7}$	84.46$^{\pm2.8}$	22.49$^{\pm1.2}$	95.84$^{\pm0.2}$	64.93$^{\pm0.5}$	83.87$^{\pm0.4}$	64.62$^{\pm0.2}$	82.72$^{\pm0.2}$	56.07$^{\pm1.2}$	85.50$^{\pm0.4}$	75.00$^{\pm0.3}$
WOODS (ours)	**71.61**$^{\pm2.3}$	**84.99**$^{\pm1.2}$	**51.66**$^{\pm2.8}$	**89.68**$^{\pm1.2}$	**12.63**$^{\pm0.6}$	**97.67**$^{\pm0.1}$	**59.77**$^{\pm0.5}$	**86.74**$^{\pm0.1}$	**58.29**$^{\pm0.4}$	**85.22**$^{\pm0.1}$	**49.87**$^{\pm1.8}$	**88.25**$^{\pm0.2}$	**75.26**$^{\pm0.2}$
$\pi = 0.5$													
OE	**68.80**$^{\pm2.8}$	82.89$^{\pm1.1}$	47.64$^{\pm4.7}$	88.84$^{\pm1.8}$	30.86$^{\pm1.9}$	93.91$^{\pm0.4}$	**56.18**$^{\pm1.6}$	86.11$^{\pm0.4}$	62.24$^{\pm0.5}$	82.53$^{\pm0.3}$	53.70$^{\pm1.6}$	86.58$^{\pm0.2}$	73.00$^{\pm0.3}$
Energy (w/ OE)	69.81$^{\pm2.4}$	85.59$^{\pm1.0}$	56.11$^{\pm3.1}$	87.41$^{\pm1.5}$	16.23$^{\pm0.6}$	97.02$^{\pm0.1}$	58.41$^{\pm0.9}$	86.70$^{\pm0.1}$	58.31$^{\pm0.5}$	85.36$^{\pm0.4}$	48.12$^{\pm1.3}$	88.76$^{\pm0.3}$	74.87$^{\pm0.4}$
WOODS (ours)	69.41$^{\pm2.7}$	**86.76**$^{\pm0.8}$	**44.60**$^{\pm2.6}$	**91.72**$^{\pm0.7}$	**12.70**$^{\pm0.4}$	**97.71**$^{\pm0.1}$	57.60$^{\pm0.6}$	**87.74**$^{\pm0.1}$	**55.03**$^{\pm0.3}$	**86.82**$^{\pm0.1}$	**45.00**$^{\pm0.7}$	**89.85**$^{\pm0.2}$	**75.72**$^{\pm0.0}$
$\pi = 1.0$													
OE	**46.45**$^{\pm2.7}$	**91.82**$^{\pm0.5}$	51.26$^{\pm3.6}$	88.47$^{\pm1.2}$	20.08$^{\pm0.7}$	96.42$^{\pm0.1}$	**51.31**$^{\pm0.8}$	88.81$^{\pm0.2}$	55.66$^{\pm0.4}$	87.28$^{\pm0.1}$	44.29$^{\pm0.6}$	90.44$^{\pm0.1}$	74.99$^{\pm0.1}$
Energy (w/ OE)	56.40$^{\pm4.0}$	89.48$^{\pm1.2}$	54.41$^{\pm2.5}$	88.77$^{\pm0.8}$	17.14$^{\pm0.9}$	96.91$^{\pm0.1}$	52.36$^{\pm1.3}$	**89.38**$^{\pm0.3}$	**54.11**$^{\pm0.9}$	**88.35**$^{\pm0.2}$	**43.42**$^{\pm1.0}$	**90.88**$^{\pm0.1}$	74.85$^{\pm0.2}$
WOODS (ours)	62.13$^{\pm4.4}$	88.89$^{\pm1.4}$	**45.87**$^{\pm1.1}$	**91.64**$^{\pm0.2}$	**13.48**$^{\pm1.1}$	**97.58**$^{\pm0.2}$	56.83$^{\pm0.7}$	88.19$^{\pm0.3}$	54.57$^{\pm0.3}$	87.43$^{\pm0.3}$	45.61$^{\pm3.0}$	89.78$^{\pm1.0}$	**75.60**$^{\pm0.2}$

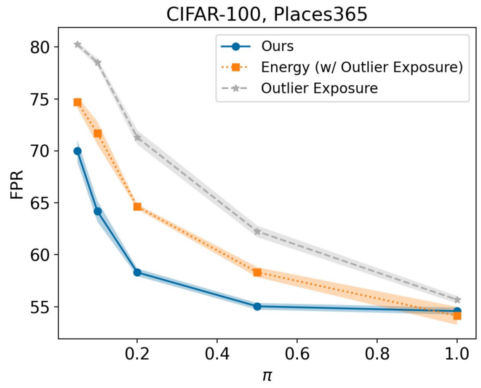
Table 2. **Effect of π.** ID dataset is `CIFAR-100`, and the auxiliary outlier training data is `300K Random Images`. ↑ indicates larger values are better and vice versa. $\pm x$ denotes the standard error, rounded to the first decimal point.

Ablation on π

Method	SVHN		LSUN-R		LSUN-C		Textures		Places365		300K Rand. Img.		Acc.
	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	FPR↓	AUROC↑	
$\pi = 0.05$													
OE	80.21$^{\pm1.7}$	77.47$^{\pm1.8}$	77.97$^{\pm2.3}$	78.68$^{\pm1.7}$	61.27$^{\pm1.4}$	86.27$^{\pm0.4}$	77.15$^{\pm1.2}$	77.94$^{\pm0.5}$	80.24$^{\pm0.3}$	74.86$^{\pm0.2}$	75.33$^{\pm0.3}$	77.16$^{\pm0.3}$	73.63$^{\pm0.3}$
Energy (w/ OE)	77.47$^{\pm2.0}$	80.48$^{\pm1.2}$	70.83$^{\pm3.1}$	82.86$^{\pm2.0}$	29.42$^{\pm4.3}$	94.61$^{\pm0.8}$	72.05$^{\pm0.8}$	80.73$^{\pm0.5}$	74.69$^{\pm0.6}$	78.60$^{\pm0.4}$	66.91$^{\pm0.7}$	80.44$^{\pm0.5}$	75.77$^{\pm0.1}$
WOODS (ours)	**74.54**$^{\pm1.7}$	**82.01**$^{\pm1.3}$	**66.29**$^{\pm3.9}$	**84.46**$^{\pm2.3}$	**19.07**$^{\pm1.6}$	**96.48**$^{\pm0.3}$	**65.75**$^{\pm0.6}$	**83.71**$^{\pm0.2}$	**69.97**$^{\pm1.1}$	**80.82**$^{\pm0.5}$	**62.48**$^{\pm1.1}$	**82.92**$^{\pm0.5}$	**75.92**$^{\pm0.1}$
$\pi = 0.1$													
OE	79.56$^{\pm1.6}$	77.00$^{\pm1.2}$	76.86$^{\pm2.1}$	78.75$^{\pm1.2}$	58.53$^{\pm2.8}$	86.92$^{\pm0.8}$	74.63$^{\pm1.2}$	79.13$^{\pm0.5}$	78.52$^{\pm0.3}$	75.68$^{\pm0.1}$	72.18$^{\pm0.2}$	78.48$^{\pm0.3}$	73.53$^{\pm0.4}$
Energy (w/ OE)	77.45$^{\pm2.1}$	80.94$^{\pm1.4}$	67.13$^{\pm3.6}$	83.68$^{\pm2.4}$	27.08$^{\pm2.1}$	94.97$^{\pm0.4}$	70.15$^{\pm1.0}$	81.59$^{\pm0.6}$	71.71$^{\pm1.1}$	79.89$^{\pm0.6}$	64.24$^{\pm2.3}$	82.28$^{\pm1.1}$	75.27$^{\pm0.2}$
WOODS (ours)	**71.67**$^{\pm1.9}$	**84.11**$^{\pm1.4}$	**59.27**$^{\pm3.9}$	**86.80**$^{\pm1.9}$	**15.03**$^{\pm1.4}$	**97.24**$^{\pm0.3}$	**61.38**$^{\pm0.7}$	**85.57**$^{\pm0.2}$	**64.19**$^{\pm1.0}$	**83.12**$^{\pm0.5}$	**55.51**$^{\pm1.3}$	**85.72**$^{\pm0.4}$	**75.64**$^{\pm0.3}$
$\pi = 0.2$													
OE	72.59$^{\pm3.9}$	81.38$^{\pm1.9}$	65.04$^{\pm3.8}$	82.65$^{\pm1.8}$	48.62$^{\pm3.1}$	89.52$^{\pm0.8}$	65.95$^{\pm1.2}$	82.43$^{\pm0.3}$	71.29$^{\pm0.7}$	78.71$^{\pm0.4}$	65.40$^{\pm0.8}$	81.99$^{\pm0.1}$	72.89$^{\pm0.3}$
Energy (w/ OE)	72.76$^{\pm2.5}$	83.48$^{\pm1.2}$	62.53$^{\pm5.7}$	84.46$^{\pm2.8}$	22.49$^{\pm1.2}$	95.84$^{\pm0.2}$	64.93$^{\pm0.5}$	83.87$^{\pm0.4}$	64.62$^{\pm0.2}$	82.72$^{\pm0.2}$	56.07$^{\pm1.2}$	85.50$^{\pm0.4}$	75.00$^{\pm0.3}$
WOODS (ours)	**71.61**$^{\pm2.3}$	**84.99**$^{\pm1.2}$	**51.66**$^{\pm2.8}$	**89.68**$^{\pm1.2}$	**12.63**$^{\pm0.6}$	**97.67**$^{\pm0.1}$	**59.77**$^{\pm0.5}$	**86.74**$^{\pm0.1}$	**58.29**$^{\pm0.4}$	**85.22**$^{\pm0.1}$	**49.87**$^{\pm1.8}$	**88.25**$^{\pm0.2}$	**75.26**$^{\pm0.2}$
$\pi = 0.5$													
OE	**68.80**$^{\pm2.8}$	82.89$^{\pm1.1}$	47.64$^{\pm4.7}$	88.84$^{\pm1.8}$	30.86$^{\pm1.9}$	93.91$^{\pm0.4}$	**56.18**$^{\pm1.6}$	86.11$^{\pm0.4}$	62.24$^{\pm0.5}$	82.53$^{\pm0.3}$	53.70$^{\pm1.6}$	86.58$^{\pm0.2}$	73.00$^{\pm0.3}$
Energy (w/ OE)	69.81$^{\pm2.4}$	85.59$^{\pm1.0}$	56.11$^{\pm3.1}$	87.41$^{\pm1.5}$	16.23$^{\pm0.6}$	97.02$^{\pm0.1}$	58.41$^{\pm0.9}$	86.70$^{\pm0.1}$	58.31$^{\pm0.5}$	85.36$^{\pm0.4}$	48.12$^{\pm1.3}$	88.76$^{\pm0.3}$	74.87$^{\pm0.4}$
WOODS (ours)	69.41$^{\pm2.7}$	**86.76**$^{\pm0.8}$	**44.60**$^{\pm2.6}$	**91.72**$^{\pm0.7}$	**12.70**$^{\pm0.4}$	**97.71**$^{\pm0.1}$	57.60$^{\pm0.6}$	**87.74**$^{\pm0.1}$	**55.03**$^{\pm0.3}$	**86.82**$^{\pm0.1}$	**45.00**$^{\pm0.7}$	**89.85**$^{\pm0.2}$	**75.72**$^{\pm[illegible]}$
$\pi = 1.0$													
OE	**46.45**$^{\pm2.7}$	**91.82**$^{\pm0.5}$	51.26$^{\pm3.6}$	88.47$^{\pm1.2}$	20.08$^{\pm0.7}$	96.42$^{\pm0.1}$	**51.31**$^{\pm0.8}$	88.81$^{\pm0.2}$	55.66$^{\pm0.4}$	87.28$^{\pm0.1}$	44.29$^{\pm0.6}$	90.44$^{\pm0.1}$	74.99$^{\pm0.1}$
Energy (w/ OE)	56.40$^{\pm4.0}$	89.48$^{\pm1.2}$	54.41$^{\pm2.5}$	88.77$^{\pm0.8}$	17.14$^{\pm0.9}$	96.91$^{\pm0.1}$	52.36$^{\pm1.3}$	**89.38**$^{\pm0.3}$	**54.11**$^{\pm0.9}$	**88.35**$^{\pm0.2}$	**43.42**$^{\pm1.0}$	**90.88**$^{\pm0.1}$	74.85$^{\pm0.2}$
WOODS (ours)	62.13$^{\pm4.4}$	88.89$^{\pm1.4}$	**45.87**$^{\pm1.1}$	**91.64**$^{\pm0.2}$	**13.48**$^{\pm1.1}$	**97.58**$^{\pm0.2}$	56.83$^{\pm0.7}$	88.19$^{\pm0.3}$	54.57$^{\pm0.3}$	87.43$^{\pm0.3}$	45.61$^{\pm3.0}$	89.78$^{\pm1.0}$	**75.60**$^{\pm0.2}$

Table 2. **Effect of π.** ID dataset is `CIFAR-100`, and the auxiliary outlier training data is `300K Random Images`. ↑ indicates larger values are better and vice versa. $\pm x$ denotes the standard error, rounded to the first decimal point.





Conclusion

- We propose a novel framework for OOD detection using unlabeled "wild" data, which occurs abundantly in the open world and can be easily collected by deployed systems
- Augmented Lagrangian methods for constrained optimization problems can be incorporated into the training process of a neural network, achieving state-of-the-art OOD detection performance and without sacrificing ID classification accuracy
- This framework may dramatically improve real-world OOD detection, enhancing the reliability of deployed ML systems

References

Bendale, A. and Boult, T. Towards open world recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1893–1902, 2015.

Bevandić, P., Krešo, I., Oršić, M., and Šegvić, S. Discriminative out-of-distribution detection for semantic segmentation. *arXiv preprint arXiv:1808.07703*, 2018.

Blanchard, G., Lee, G., and Scott, C. Semi-supervised novelty detection. *The Journal of Machine Learning Research*, 11:2973–3009, 2010.

Chalapathy, R. and Chawla, S. Deep learning for anomaly detection: A survey. *arXiv preprint arXiv:1901.03407*, 2019.

Chalapathy, R., Menon, A. K., and Chawla, S. Anomaly detection using one-class neural networks. *arXiv preprint arXiv:1802.06360*, 2018.

Chen, J., Li, Y., Wu, X., Liang, Y., and Jha, S. Atom: Robustifying out-of-distribution detection using outlier mining. *In Proceedings of European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD)*, 2021.

Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., and Vedaldi, a. A. Describing Textures in the Wild. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.

Daniel, T., Kurutach, T., and Tamar, A. Deep variational semi-supervised novelty detection. *arXiv preprint arXiv:1911.04971*, 2019.

Du, X., Wang, X., Gozum, G., and Li, Y. Unknown-aware object detection: Learning what you don't know from videos in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022a.

Duchi, J., Hazan, E., and Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

Ergen, T. and Kozat, S. S. Unsupervised anomaly detection with lstm neural networks. *IEEE transactions on neural networks and learning systems*, 31(8):3127–3141, 2019.

Hendrycks, D. and Gimpel, K. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *Proceedings of International Conference on Learning Representations*, 2017.

Hendrycks, D., Mazeika, M., and Dietterich, T. Deep anomaly detection with outlier exposure. In *International Conference on Learning Representations*, 2019.

Hestenes, M. R. Multiplier and gradient methods. *Journal of optimization theory and applications*, 4(5):303–320, 1969.

Hsu, Y.-C., Shen, Y., Jin, H., and Kira, Z. Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10951–10960, 2020.

Huang, R., Geng, A., and Li, Y. On the importance of gradients for detecting distributional shifts in the wild. In *Advances in Neural Information Processing Systems*, 2021.

Huber, P. J. Robust estimation of a location parameter. *Annals of Mathematical Statistics*, 35:73–101, March 1964.

Krizhevsky, A., Hinton, G., and others. Learning multiple layers of features from tiny images. 2009. Publisher: Citeseer.

Lee, K., Lee, H., Lee, K., and Shin, J. Training confidence-calibrated classifiers for detecting out-of-distribution samples. *International Conference on Learning Representations (ICLR)*, 2018a.

Lee, K., Lee, K., Lee, H., and Shin, J. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Advances in Neural Information Processing Systems*, pp. 7167–7177, 2018b.

Liang, S., Li, Y., and Srikant, R. Enhancing the reliability of out-of-distribution image detection in neural networks. In *6th International Conference on Learning Representations, ICLR 2018*, 2018.

Liu, W., Wang, X., Owens, J., and Li, Y. Energy-based out-of-distribution detection. *Advances in Neural Information Processing Systems*, 2020.

Malinin, A. and Gales, M. Predictive uncertainty estimation via prior networks. *arXiv preprint arXiv:1802.10501*, 2018.

Ming, Y., Fan, Y., and Li, Y. Poem: Out-of-distribution detection with posterior sampling. In *International Conference on Machine Learning (ICML)*. PMLR, 2022.

Morteza, P. and Li, Y. Provable guarantees for understanding out-of-distribution detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading Digits in Natural Images with Unsupervised Feature Learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011.

Nguyen, A., Yosinski, J., and Clune, J. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 427–436, 2015.

Nocedal, J. and Wright, S. *Numerical optimization*. Springer Science & Business Media, 2006.

Perera, P. and Patel, V. M. Learning deep features for one-class classification. *IEEE Transactions on Image Processing*, 28(11):5450–5463, 2019.

Rockafellar, R. T. A dual approach to solving nonlinear programming problems by unconstrained optimization. *Mathematical programming*, 5(1):354–373, 1973.

Ruff, L., Vandermeulen, R., Goernitz, N., Deecke, L., Siddiqui, S. A., Binder, A., Müller, E., and Kloft, M. Deep one-class classification. In *International conference on machine learning*, pp. 4393–4402. PMLR, 2018.

Ruff, L., Vandermeulen, R. A., Görnitz, N., Binder, A., Müller, E., Müller, K.-R., and Kloft, M. Deep semi-supervised anomaly detection. In *International Conference on Learning Representations*, 2020.

Ruff, L., Kauffmann, J. R., Vandermeulen, R. A., Montavon, G., Samek, W., Kloft, M., Dietterich, T. G., and Müller, K.-R. A unifying review of deep and shallow anomaly detection. *Proceedings of the IEEE*, 2021.

Sangalli, S., Erdil, E., Hötker, A., Donati, O., and Konukoglu, E. Constrained optimization to train neural networks on critical and under-represented classes. *Advances in Neural Information Processing Systems*, 34, 2021.

Song, H., Jiang, Z., Men, A., and Yang, B. A hybrid semi-supervised anomaly detection model for high-dimensional data. *Computational intelligence and neuroscience*, 2017, 2017.

Sun, Y., Guo, C., and Li, Y. React: Out-of-distribution detection with rectified activations. In *Advances in Neural Information Processing Systems*, 2021.

Sun, Y., Ming, Y., Zhu, X., and Li, Y. Out-of-distribution detection with deep nearest neighbors. In *International Conference on Machine Learning (ICML)*. PMLR, 2022.

References (cont.)

Tack, J., Mo, S., Jeong, J., and Shin, J. Csi: Novelty detection via contrastive learning on distributionally shifted instances. In *Advances in Neural Information Processing Systems*, 2020.

Wang, H., Liu, W., Bocchieri, A., and Li, Y. Can multi-label classification networks know what they don't know? *Advances in Neural Information Processing Systems*, 34, 2021.

Xu, Y. First-order methods for constrained convex programming based on linearized augmented lagrangian function. *Informs Journal on Optimization*, 3(1):89–117, 2021a.

Xu, Y. Iteration complexity of inexact augmented lagrangian methods for constrained convex programming. *Mathematical Programming*, 185(1):199–244, 2021b.

Yan, Y. and Xu, Y. Adaptive primal-dual stochastic gradient method for expectation-constrained convex stochastic programs. *arXiv preprint arXiv:2012.14943*, 2020.

Yu, F., Seff, A., Zhang, Y., Song, S., Funkhouser, T., and Xiao, J. LSUN: Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop. *arXiv:1506.03365 [cs]*, June 2016. arXiv: 1506.03365.

Zagoruyko, S. and Komodakis, N. Wide Residual Networks. In *Procedings of the British Machine Vision Conference*, 2016.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., and Torralba, A. Places: A 10 Million Image Database for Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6):1452–1464, June 2018.