Robust Perception through Equivariance

Chengzhi Mao [1] Lingyu Zhang [1] Abhishek Vaibhav Joshi [1] Junfeng Yang [1] Hao Wang [2] Carl Vondrick [1]

Abstract

Deep networks for computer vision are not reliable when they encounter adversarial examples. In this paper, we introduce a framework that uses the dense intrinsic constraints in natural images to robustify inference. By introducing constraints at inference time, we can shift the burden of robustness from training to testing, thereby allowing the model to dynamically adjust to each individual image's unique and potentially novel characteristics at inference time. Our theoretical results show the importance of having dense constraints at inference time. In contrast to existing single-constraint methods, we propose to use equivariance, which naturally allows dense constraints at a fine-grained level in the feature space. Our empirical experiments show that restoring feature equivariance at inference time defends against worst-case adversarial perturbations. The method obtains improved adversarial robustness on four datasets (ImageNet, Cityscapes, PASCAL VOC, and MS-COCO) on image recognition, semantic segmentation, and instance segmentation tasks.

1. Introduction

Despite the strong performance of deep networks on computer vision benchmarks (He et al., 2016; 2017; Yu et al., 2017), state-of-the-art systems are not reliable when evaluated in open-world settings (Geirhos et al., 2019; Hendrycks et al., 2021; Szegedy et al., 2013; Hendrycks & Dietterich, 2019; Croce & Hein, 2020; Carlini & Wagner, 2017). However, the robustness against a large number of adversarial cases remains a prerequisite necessary to deploy models in real-world applications, such as in medical imaging, healthcare, and robotics.

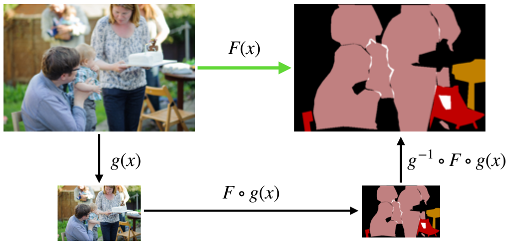

Figure 1. Equivariance is shared across the input images (left) and the output labels (right), providing a dense constraint. The predictions from a model $F(x)$ should be identical to performing a spatial transformation on x, a forward pass of F, and undoing that spatial transformation on the output space (black).

Due to the importance of this problem, there has been a large number of investigations aiming to improve the training algorithm to establish reliability. For example, data augmentation (Yun et al., 2019; Hendrycks et al., 2021) and adversarial training (Madry et al., 2017; Carmon et al., 2019) improve robustness by training the model on anticipated distribution shifts and worst-case images. However, placing the burden of robustness on the training algorithm means that the model can only be robust to the corruptions that are anticipated ahead of time, which is an unrealistic assumption in an open world. In addition, retraining the model on new distributions each time can be expensive.

To address this challenge, we propose to robustify the model at inference time. Specifically, instead of retraining the whole model on the new distribution, our inference-time defense shifts the burden to test time with our robust inference algorithm without updating the model. Prior work (Mao et al., 2021; Shi et al., 2020; Wang et al., 2021) is limited to a single constraint at inference time and hence may not provide the model with enough information to dynamically adjust to the unique and potentially novel characteristics of the corruption in the testing image. We therefore ask the natural question: *Can we further improve the robustness through increasing the number of constraints?*

We start with theoretical analysis and prove that applying more constraints at inference time strictly improves the model's robustness. The next question is then: *How to efficiently apply multiple constraints at inference time?* One approach is to directly apply multiple feature invariance

[1] Department of Computer Science, Columbia University, New York, USA [2] Department of Computer Science, Rutgers University, New Jersey, USA. Correspondence to: Chengzhi Mao <cm3797@columbia.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

constraints to the defense. While this defense is effective, we find the resulting representation can be limited by the invariance property, therefore harming robust accuracy. For example, after resizing, the representations of the segmentation models are not the same, and it is unclear which part should be invariant.

Our further study with empirical results suggest that a better approach is to use dense equivariance constraints. Our main hypothesis is that visual representations must be equivariant under spatial transformations, which is a dense property that should hold for all natural images (equivariance consistency in Figure 1). This property holds when the test data are from the same distribution that the model has been trained on. However, once there has been an adversarial corruption, the equivariance is often broken (Figure 2). Therefore our key insight is that we can repair the model's prediction on corrupted data by restoring the equivariance.

Empirical experiments, theoretical analysis, and visualizations highlight that equivariance significantly improves model robustness over other methods (Mao et al., 2021; Shi et al., 2020; Wang et al., 2021). On four large datasets (ImageNet (Deng et al., 2009), Cityscapes (Cordts et al., 2016), PASCAL-VOC (Everingham et al., 2010), and MS-COCO (Lin et al., 2014)), our approach improves adversarial robust accuracy by up to 15 points. Our study shows that equivariance can efficiently improve robustness by increasing the number of constraints (Figure 3). Even under two adaptive adversarial attacks where the attacker knows our defense (Athalye et al., 2018; Mao et al., 2021), adding our method improves robustness. In addition, since equivariance is an intrinsic property of visual models, we do not need to train a separate model to predict equivariance (Shi et al., 2020; Mao et al., 2021). Our code is available at `https://github.com/cvlab-columbia/Equi4Rob`.

2. Related Work

Equivariance. Equivariance benefits a number of visual tasks (Dieleman et al., 2016; Cohen & Welling, 2016b; Gupta et al., 2021; Zhang, 2019; Chaman & Dokmanic, 2021; Chaman & Dokmanić, 2021). Cohen & Welling (2016a) proposed the first group-convolutional operation that produces equivariant features to symmetry-group. However, it can only be equivariant on a discrete subset of transformation (Sosnovik et al., 2019). Steerable equivariance achieves continuous equivariant transformation (Cohen & Welling, 2016b; Weiler et al., 2018) on the defined set of basis, but they cannot be applied to arbitrary convolution filters due to the requirement of an equivariant basis. Besides architecture design (Weiler & Cesa, 2019), adding regularization (Barnard & Casasent, 1991) can improve equivariance in the network. (Kamath et al., 2021) shows that training equivariance at training time decreases adversarial robustness. Our method sidesteps this issue by promoting equivariance for attacked images at test time, improving equivariance when it is most needed.

Adversarial Attack and Defense. Adversarial attacks (Szegedy et al., 2013; Madry et al., 2017; Cisse et al., 2017a; Dong et al., 2018; Carlini & Wagner, 2017; Croce & Hein, 2020; Arnab et al., 2018) are perturbations optimized to change the prediction of deep networks. Adversarial training (Madry et al., 2017; Rice et al., 2020; Carmon et al., 2019) and its variants (Mao et al., 2019; 2022; Zhang et al., 2019) are the standard way to defend adversarial examples. The matching algorithm to produce features invariant to adversarial perturbations has been shown to produce robust models (Mahajan et al., 2021; Zhang et al., 2019). However, training time defense can only be robust to the attack that it has been trained on. Multitask learning (Mao et al., 2020; Zamir et al., 2020) and regularization (Cisse et al., 2017b) can improve adversarial robustness. However, they did not consider the spatial equivariance in their task. Recently, inference time defense using contrastive invariance (Mao et al., 2021) and rotation (Shi et al., 2020) has been shown to improve adversarial robustness without retraining the model on unforeseen attacks. However, they only apply a single constraint, which may not provide enough information.

Test Test Adaptation. Berthelot et al. (2019); Pastore et al. (2021) perform test-time training on the entire test set for many iterations, our method only assumes seeing one example at a time and performs test-time adaptation on a single image. Tsai et al. (2023) adapts the model with convolutional prompt, but only works for a large batchsize. Test-time adaption is also useful in language domain (McDermott et al.). By leveraging equivariance, we can efficiently incorporate dense constraints into our framework, which can be orders of magnitude more effective than adding constraints individually (Sun et al., 2020; Lawhon et al., 2022).

3. Method

In this section, we first introduce equivariance for visual representation, present algorithms to improve adversarial robustness using equivariance, and then provide theoretical insight into why the multiple constraints can lead to such improvement.

3.1. Equivariance in Vision Representation

Let $\mathbf{x}$ be an input image. A neural network produces a representation $\mathbf{h} = F_\theta(\mathbf{x})$ for the input image. Assume there is a transformation g for the input image. A neural network is equivariant only when:

$$F_\theta(\mathbf{x}) = g^{-1} \circ F_\theta \circ g(\mathbf{x}), \tag{1}$$

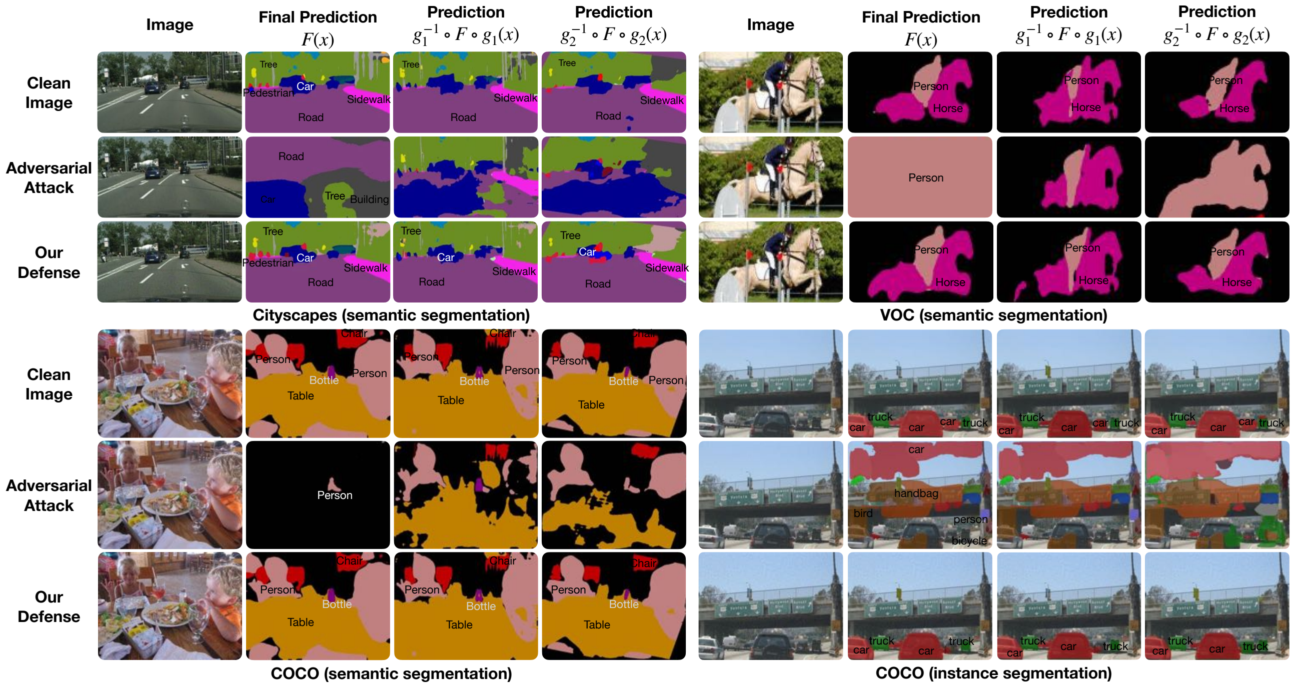
Image
Final Prediction $F(x)$
Prediction $g_1^{-1} \circ F \circ g_1(x)$
Prediction $g_2^{-1} \circ F \circ g_2(x)$
Clean Image
Adversarial Attack
Our Defense
Tree
Car
Pedestrian
Sidewalk
Road
Building
Person
Horse
Cityscapes (semantic segmentation)
VOC (semantic segmentation)
Chair
Bottle
Table
truck
car
handbag
bird
bicycle
COCO (semantic segmentation)
COCO (instance segmentation)

Figure 2. Random examples showing equivariance on clean images and non-equivariance on attacked images in Cityscapes, PASCAL VOC, and COCO. The representation is equivariant when the predicted images (2nd column) and the reversed prediction of transformed images (3rd, 4th column) are the same. By restoring equivariance, our method corrects the prediction.

where $g^{-1}(\cdot)$ denotes the inverse transformation for $g(\cdot)$, and $\circ$ denotes function composition. Equivariant representations will change symmetrically as the input transformation. This means applying the transformation to the input image and undoing the transformation in the representation space, should result in the same representation as fed in the original image. Equivariance provides a meta property that can be applied to dense feature maps, and generalized to most existing vision tasks (Gupta et al., 2021; Laptev et al., 2016; Marcos et al., 2016).

In contrast, invariance is defined as $F_\theta(\mathbf{x}) = F_\theta \circ g(\mathbf{x})$, which requires the model to produce the same representation after different transformations, such as texture augmentation (Geirhos et al., 2019) and color jittering (Mao et al., 2021; Chen et al., 2020). Without performing transformation in the same way as the input, invariance removes all the information related to the transformation, which can hurt the final task if the transformation is crucial to the final task (Lee et al., 2021). On the contrary, equivariant models maintain the covariance of the transformations (Gupta et al., 2021; Laptev et al., 2016; Marcos et al., 2016).

Transformation for Equivariance. We use spatial transformation, such as flip, resizing, and rotation, in our experiments. Assume we apply k different transformations g_i where $i = 1, ..., k$. We denote the cosine similarity as $cos(\cdot)$. Equivariance across all transformations means the following term is large:

$$\mathcal{L}_{equi} = \sum_{i=1}^{k} cos(g_i^{-1} \circ F_\theta \circ g_i(\mathbf{x}), F_\theta(\mathbf{x})) \quad (2)$$

Algorithm 1 Equivariance Defense

```
Input: Potentially attacked image $\mathbf{x}$, step size $\eta$, number of iterations $T$, deep network $F$, reverse attack bound $\epsilon_v$, and equivariance loss function $\mathcal{L}_{equi}$.
Output: Prediction $\widehat{y}$
Inference: $\mathbf{x}' \leftarrow \mathbf{x}$
for $t = 1, ..., T$ do
    $\mathbf{x}' \leftarrow \mathbf{x}' + \eta \cdot \text{sign}(\text{Normalize}(\nabla_{\mathbf{x}'} \mathcal{L}_{equi}(\mathbf{x}')) + \mathcal{N}(0, \frac{T-1-t}{T}))$
    $\mathbf{x}' \leftarrow \Pi_{(\mathbf{x}, \epsilon_v)} \mathbf{x}'$, which projects the image back into the bounded region.
end for
Predict the final output by $\widehat{y} = F(\mathbf{x}')$
```

3.2. Equivariance for Adversarial Robustness

Let $\mathbf{y}$ be the ground-truth category labels for $\mathbf{x}$. Let the network that uses the feature $\mathbf{h}$ for final task prediction to be $C_{\theta'}$. For prediction, neural networks learn to predict the category $\widehat{\mathbf{y}} = C_{\theta'} \circ F_\theta(\mathbf{x})$ by minimizing the loss $L(\widehat{\mathbf{y}}, \mathbf{y})$ between the predictions and the ground truth. For example, for semantic segmentation, L is cross-entropy for each pixel output. We define the loss for the final task as follows:

$$\mathcal{L}_t(\mathbf{x}, \mathbf{y}) = L\left(C_{\theta'} \circ F_\theta(\mathbf{x}), \mathbf{y}\right), \tag{3}$$

Adversarial Attack. To fool the model's prediction, the adversarial attack finds additive perturbations $\boldsymbol{\delta}$ to the image such that the loss of the task (Equation 15) is maximized.

$$\mathbf{x}_a = \underset{\mathbf{x}_a}{\operatorname{argmax}} \mathcal{L}_t(\mathbf{x}_a, \mathbf{y}), \quad \text{s.t.} \quad ||\mathbf{x}_a - \mathbf{x}||_q \leq \epsilon, \tag{4}$$

where the perturbation vector $\boldsymbol{\delta} = \mathbf{x}_a - \mathbf{x}$ has a q norm bound that is smaller than ϵ, keeping the perturbation invisible to humans.

Equivariance Recalibration Defense. Given an input image, we can calculate the equivariant loss $\mathcal{L}_{equi}$. As shown in Figure 2, the representations are non-equivariant when the input $\mathbf{x}_a$ is adversarially perturbed, i.e., the term $\mathcal{L}_{equi}$ is low. We will find an intervention to recalibrate the input image $\mathbf{x}_a$ such that we can improve the feature equivariance of the image. To do this, we optimize a vector $\mathbf{r}$ by maximizing the equivariance objective:

$$\mathbf{r} = \underset{\mathbf{r}}{\operatorname{argmax}} \mathcal{L}_{equi}(\mathbf{x}_a + \mathbf{r}), \quad \text{s.t.} \quad ||\mathbf{r}||_q \leq \epsilon_v, \tag{5}$$

where ϵ_v defines the bound of our reverse attack $\mathbf{r}$. The additive intervention $\mathbf{r}$ will modify the adversarial image $\mathbf{x}_a$ such that it restores the equivariance in feature space.

We optimize the above objective via projected gradient descent to repair the input. To avoid the optimization converging to a local optimal, we first perform SGLD (Welling & Teh, 2011) to get a good Bayesian posterior distribution of the solution (Wang et al., 2019). To avoid sampling from the posterior distribution of SGLD and improve the inference speed, we then use maximum a posterior (MAP) estimation to find a single solution. Empirically, we add Gaussian noise to the gradient when optimizing and linearly anneal the noise level to zero. We show the optimization procedure in Algorithm 1. We use the same optimizer for the invariance objective and compare.

In contrast to Mao et al. (2021); Shi et al. (2020), we do not need to pre-train another network for the self-supervision task offline. In addition, equivariance in the feature space provides a dense constraint because, by projecting the transformation back to the original space, we can match each element in the feature space. Image-level self-supervision tasks, such as contrastive loss and rotation prediction, do not have this dense supervision advantage.

Adaptive Attack I. We now analyze our methods' robustness when the attacker knows our defense strategy and takes our defense into consideration. Following the defense-aware attack setup in (Mao et al., 2021), the adaptive attacker can maximize the following equation:

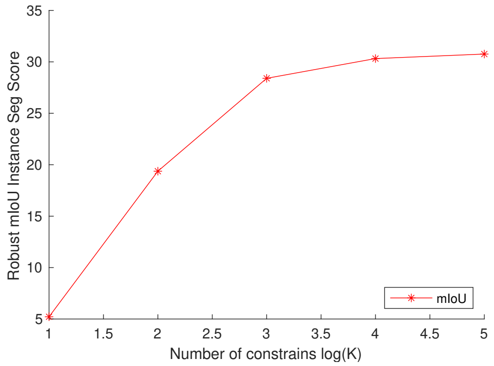

Figure 3. Adversarial robustness under an increased number of constraints through equivariance at inference time.

$$\mathcal{L}_l(\mathbf{x}_a, \mathbf{y}, \lambda_s) = \mathcal{L}_t(\mathbf{x}_a, \mathbf{y}) + \lambda_e \mathcal{L}_{equi}(\mathbf{x}_a). \tag{6}$$

where the first term fools the final task, and the second term optimizes for equivariance. A larger λ_e allows the adversarial budget to focus more on respecting the feature equivariance, which reduces the defense capability of our defense. However, with a fixed adversarial budget, increasing λ_e also reduces the attack efficiency for the final task. Our defense creates a lose-lose situation for the attacker. If they consider our defense, they hurt the attack efficiency for the final task. If they ignore our defense, our defense will fix the attack.

Adaptive Attack II. The above adaptive attack avoids the unstable gradient from the iterative optimization with a Lagrangian regularization term. Another way to bypass such defense is through BPDA (Athalye et al., 2018). Specifically, the equivariance recalibration process formulated in Eq. 5 can be treated as a preprocessor $h(\cdot)$ that is employed at test time, where $h(\mathbf{x}_a) = \mathbf{x}_a + \mathbf{r}$. Given a pre-trained classifier $f(\cdot)$, this method can be formulated as $f(h(x))$. The proposed process $h(\cdot)$ may cause exploding or vanishing gradients. According to (Athalye et al., 2018; Croce et al., 2022), we can use BPDA to approximate $h(\cdot)$, where an identity function is used for the backward pass of the restored images. While this method may make the backward gradient inaccurate, it avoids differentiation through the inner optimization procedure, which often leads to vanished or exploded gradients.

3.3. Theoretical Results for Adversarial Robustness with Multiple Constraints

One major advantage of equivariance is that it allows dense constraints through the inverse transformation. We show theoretical insights for why using a dense intrinsic constraint rather than a single intrinsic constraint. Existing methods restore the input image to respect a single self-supervision label $y^{(s_1)}$. With a dense intrinsic constraint, the defense model can predict with a set of fine-grained self-supervision signals. $y^{(s_i)}$, where $i = 1, 2, ..., K$. In our case, each $y_a^{(s_i)}$ is the predicted self-supervision value under adversarial attack, and each $y^{(s_i)}$ is the predicted self-supervision value in our feature map after equivariance transformation. Following (Mao et al., 2021), we propose the following lemma:

Lemma 3.1. *The standard classifier under adversarial attack is equivalent to predicting with* $P(\mathbf{Y}|\mathbf{X}_a, y_a^{(s_1)}, y_a^{(s_2)}, ..., y_a^{(s_k)})$*, and our approach is equivalent to predicting with* $P(\mathbf{Y}|\mathbf{X}_a, y^{(s_1)}, y^{(s_2)}, ..., y^{(s_k)})$*.*

By adjusting the input image such that it satisfies a set of denser constraints, the predicted task Y uses both the information from the image and the intrinsic equivariance structure. We now show that by restoring the dense constraints in our visual representation, from an information perspective, the upper bound can be strictly improved than just restoring the structure from a single self-supervision task (Mao et al., 2021; Shi et al., 2020).

Theorem 3.2. *Assume the classifier operates better than chance and instances in the dataset are uniformly distributed over n categories. Let the prediction accuracy bounds be* $P(\mathbf{Y}|y_a^{(s_1)}, y_a^{(s_2)}, ..., y_a^{(s_k)}, \mathbf{X}_a) \in [b_0, c_0]$, $P(\mathbf{Y}|y^{(s_1)}, \mathbf{X}_a) \in [b_1, c_1]$, $P(\mathbf{Y}|y^{(s_1)}, y^{(s_2)}, \mathbf{X}_a) \in [b_2, c_2]$, ..., *and* $P(\mathbf{Y}|y^{(s_1)}, y^{(s_2)}, ..., y^{(s_k)}, \mathbf{X}_a) \in [b_k, c_k]$*. If the conditional mutual information* $I(\mathbf{Y}; \mathbf{Y}^{(s_i)}|\mathbf{X}_a) > 0$ *and* $I(\mathbf{Y}; \mathbf{Y}^{(s_i)}|\mathbf{X}_a, \mathbf{Y}^{(s_j)}) > 0$ *where $i \neq j$, we have* $b_0 \leq b_1 \leq ... \leq b_k$ *and* $c_0 < c_1 < c_2 < ... < c_k$*, which means our approach strictly improves the upper bound for classification accuracy.*

In words, the adversarial perturbation $\mathbf{X}_a$ corrupts the shared information between the label $\mathbf{Y}$ (our target task) and the equivariance structure $\mathbf{Y}^{s_i}$ (self-supervised task). Theorem 3.2 shows that by recovering information from more $\mathbf{Y}^{s_i}$, the task performance can be improved.

Directly increasing the number of invariance objectives is a straightforward baseline to increase the number of constraints. However, this can be limited because 1) each invariance objective only adds one constraint, which is less efficient, and 2) invariance cannot be directly applied to many transformations, such as resizing and rotation, due to the mismatch in fine-grained representation, where equivariance can. In contrast to invariance, dense equivariance allows us add one constraint on each element in the feature map[1], which can increase constraints orders of magnitude faster with a more diverse set of transformations, providing an efficient way to apply multiple constraints. By subsampling different number of constraints from equivariance, Figure 3 validates the trend of improving robustness as the number of constraints increases.

The adaptive attack needs to respect the information in Y^{s_i}, which itself limits the ability of the attacker, as the attacker performs a multitask optimization which is harder (Mao et al., 2020). The adaptive attacker predicts the task conditioned on the right set of self-supervision label $\mathbf{Y}^{s_i}$, which fulfills our Theorem 3.2 and improves robustness.

4. Experiments

Our experiments evaluate the adversarial robustness on four datasets: ImageNet (Deng et al., 2009), Cityscapes (Cordts et al., 2016), PASCAL-VOC (Everingham et al., 2010), and MS-COCO (Lin et al., 2014). We use up to 6 different strong attacks, including Houdini, adaptive attack, and BPDA to evaluate the robustness. We first show that our equivariance-based defense improves the robustness of the state-of-the-art adversarially trained robust models. We then show that even on the standard models without defense training, adding test-time equivariance can improve their robustness.

4.1. Dataset and Tasks

ImageNet (Deng et al., 2009) contains 1000 categories. Due to its large size, we randomly sample 2% of data for evaluation. *Cityscapes* (Cordts et al., 2016) is a urban driving scene dataset. We study the semantic segmentation task. Following (Mao et al., 2020), we resize the image to 680×340 for fast inference. We use pretrained dilated residual network (DRN) for segmentation. *PASCAL-VOC* (Everingham et al., 2010) is a dataset for semantic segmentation task. We resize images to 480×480. We use the pre-trained DeepLabV3+ model. *MS-COCO* (Lin et al., 2014) is a large-scale image dataset of common objects that supports semantic segmentation and instance segmentation task. For semantic segmentation, we resize the images to 400×400. We use pretrained DeeplabV3 and MaskRCNN for semantic segmentation and instance segmentation, respectively.

4.2. Attack Methods

IFGSM (seg) (Arnab et al., 2018) was used to evaluate the robustness of segmentation models with multiple steps of the fast gradient sign method. *PGD* (Madry et al., 2017) is a standard iterative-based adversarial attack, which performs gradient ascent and projects the attack vector inside the defined p norm ball. *MIM* (Dong et al., 2018) adds a momentum term to the gradient ascent of PGD attack, which is a stronger attack that can get out of local optima. *Houdini* (Cisse et al., 2017a) is the state-of-the-art adversarial attack for decreasing the mIoU score of semantic segmentation. It proposes a surrogate objective function that can be optimized on the mIoU score directly. *Adaptive Attack (AA)* (Mao et al., 2021) is the standard defense-aware attack for inference time defense method, where the adaptive attack knows our defense algorithm, and optimizes the attack

[1] A 100 by 100 feature map would provide 10000 constraints

Table 1. Classification accuracy on ImageNet and segmentation mIoU on Cityscapes dataset on adversarially trained models with $\epsilon = 4/255$. Using equivariance improves robustness more than other methods.

	ImageNet; Adversarially Pretrained Model (Wong et al., 2020); Classification Accuracy					
Evaluation Method	Vanilla	Random	Rotation	Contrastive	Invariance	Equivariance (Ours)
Clean	**51.5**	49.4	49.5	49.2	48.8	49.3
PGD	26.5	28.0	28.2	29.3	28.6	**32.2**
CW	26.6	28.3	28.6	29.8	32.2	**32.2**
AA	26.5	28.0	28.2	29.3	28.6	**32.2**
BPDA	26.5	28.0	27.9	28.8	28.9	**30.4**

	Cityscape; Adversarially Trained DRN-22-d; Segmentation MIoU					
Evaluation Method	Vanilla	Random	Rotation	Contrastive	Invariance	Equivariance (Ours)
Clean	**53.23**	52.96	51.72	53.00	49.04	48.74
IFGSM (seg)	33.06	33.21	33.47	33.59	32.36	**34.04**
PGD	26.61	27.04	27.68	28.14	27.74	**29.65**
MIM	26.59	27.06	27.72	28.24	27.76	**29.56**
Houdini	23.47	24.07	25.56	26.61	26.97	**29.80**
AA	26.61	27.04	27.68	28.14	27.74	**29.63**
BPDA	26.61	27.00	26.37	28.50	23.31	**29.83**

vector to respect equivariance while fooling the final task. Since the attack already respects and adapts to equivariance, our defense has less space to improve by further optimizing for equivariance. *BPDA* (Athalye et al., 2018; Croce et al., 2022) is an adaptive attack for input purification. In our case, we forward the adapted images in the forward pass and straight-through the gradient from our adapted image to the input image.

4.3. Baselines

We compare our method with the vanilla feed-forward inference and four existing inference-time defense methods. *Random* defense (Kumar et al., 2020) defends adversarial attack by adding random noise to the input, which is used as a baseline in (Mao et al., 2021). *Rotation* defense (Shi et al., 2020) purifies the adversarial examples by restoring the performance of the rotation task at inference time, which can recover the image information that relates to rotation. However, the information related to rotation may be misleading due to the illusion issue (ill), which limits its power for complex tasks. *Contrastive* defense (Mao et al., 2021) restores the intrinsic structure of the image using SimCLR (Chen et al., 2020) objective at inference time, which achieves state-of-the-art adversarial robustness on image recognition tasks. Contrastive learning requires images to be object-centric, which may not be true on the segmentation and detection dataset where multiple objects appear in the same image. *Invariance* defense follows the same setup as our equivariance experiment but replaces the equivariance loss with the invariance loss. To obtain several constrants from invariance, we use the same diversified set of transformations as the equivariance setup. We propose this baseline to study the importance of using equivariance to apply multiple constraints.

4.4. Implementation details

We choose the number of transformations to be $K = 8$, which empirically can be fit into a 2080Ti GPU with batch size 1. To increase the constraints obtained from equivariance, we empirically use a diversified set of transformations, which includes four resizing transformations ranging from 0.3 to 2 times of size change; one color jittering transformation; one horizontal flip transformation; and two rotation transformations between -15 to 15 degrees. For transformations that cause part of the original image not in the view, we only consider the overlapped region when calculating the loss. Ablation study for the effect of each transformation is shown in Section 4.7. We use steps $T = 20$ for all our defense tasks. Since after the spatial transformations, the invariance objective cannot be performed in the dense feature space due to the position mismatch, we apply an average pooling for all the features and then compute the invariance loss.

4.5. Results on Adversarial Trained Models

Adversarial training is the standard way to defend against adversarial examples. We first validate whether our proposed approach can further improve the robustness of adversarially trained models. For ImageNet, we use the adversarial pretrained model with $\epsilon = 4/255$ from (Wong et al., 2020). We set the defense vector bound to be $\epsilon_v = 2\epsilon$. With the state-of-the-art contrastive learning method (Mao et al., 2021), we improve robustness accuracy by 3 points to the Vanilla

Table 2. Semantic segmentation mIoU on Cityscapes, PASCAL VOC, and MSCOCO dataset. All models are not adversarially trained. Under different types of attack bounded by $L_\infty = 4/255$, our method consistently outperforms other defense methods.

	Cityscape; Pretrained DRN-22-d Model; Segmentation MIoU					
Evaluation Method	Vanilla	Random	Rotation	Contrastive	Invariance	Equivariance (Ours)
Clean	58.29	52.38	34.30	37.22	33.84	37.95
PGD	1.31	1.47	13.20	8.44	14.49	**30.76**
MIM	1.40	1.49	13.80	8.13	14.57	**30.10**
Houdini	0.00	0.21	16.31	10.12	14.16	**30.52**
AA	1.31	1.47	13.20	8.44	14.49	**30.28**
BPDA	1.31	1.47	6.20	4.78	6.82	**11.64**

	PASCAL VOC dataset; Pretrained DeepLabV3; Segmentation MIoU					
Evaluation Method	Vanilla	Random	Rotation	Contrastive	Invariance	Equivariance (Ours)
Clean	69.52	68.96	28.63	66.92	63.64	56.58
PGD	6.46	6.52	6.91	18.72	39.07	**43.51**
MIM	5.63	5.74	6.35	18.25	37.43	**41.56**
Houdini	0.02	0.08	6.14	19.11	31.30	**52.26**
BPDA	6.46	6.46	8.23	5.45	15.15	**25.68**

	MSCOCO dataset; Pretrained DeeplabV3-resnet50; Segmentation MIoU					
Evaluation Method	Vanilla	Random	Rotation	Contrastive	Invariance	Equivariance (Ours)
Clean	63.02	62.97	60.92	57.28	43.07	44.71
PGD	2.62	2.65	5.79	14.75	23.92	**24.51**
MIM	2.71	2.52	5.66	13.61	20.53	**21.30**
Houdini	0.05	0.10	4.78	22.69	36.94	**37.33**
BPDA	2.62	2.63	1.15	2.35	17.13	**18.69**

	MaskRCNN; Instance Segmentation maskAP					
Evaluation Method	Vanilla	Random	Rotation	Contrastive	Invariance	Equivariance (Ours)
Clean	34.5	33.6	31.2	29.7	14.3	23.4
PGD	0.0	1.6	2.6	8.9	12.9	**21.3**
MIM	0.0	1.6	2.7	9.1	13.2	**21.2**
BPDA	0.0	0.0	0.3	1.7	8.7	**9.9**

defended model. Adaptive attack (AA) poses a lose-lose situation and does not further decrease the robustness accuracy, which is consistent with the observation of (Mao et al., 2021). With the strongest adaptive attack BPDA (Croce et al., 2022), it drops 0.5 points. [2] Using the equivariance objective, under both standard attack and the adaptive attack BPDA, it improves robustness more than the other methods. Even though BPDA decreases equivariance defense by 1.8 points, equivariance still improves robustness by 3.9 points than not using it.

On Cityscapes, we downsample the image from 2048 × 1024 to 680 × 340 to reduce computation, which follows the setup of (Mao et al., 2020). We adversarially train a segmentation model and evaluate it in Table 1, which is measured with mean Intersection over Union (mIoU) for semantic segmentation. We set the defense vector bound to be $\epsilon_v = 2.5\epsilon$. For the standard attack, Houdini reduces the robustness accuracy the most, where using equivariance constraints at test time can recover 6 points of performance. Using the adaptive attack (Mao et al., 2021), the robust accuracy of equivariance only drops by 0.2 points. Using the BPDA adaptive attack, the robustness of the invariance-based method drops 4 points, which suggests that invariance relies mostly on obfuscated gradients and it is not an effective constraint to maintain at inference time for segmentation. In contrast, BPDA *cannot* undermine the equivariance-based model's robustness. On the adversarially trained model, equivariance consistently outperforms all other test-time defenses, which demonstrates that equivariance is a better intrinsic structure to respect during inference time.

[2] Recent work (Croce et al., 2022) uses a batch size of 50 for contrastive loss, which is a weaker defense due to the small batch size. Here, we use the original batch size of 400 setup as (Mao et al., 2021), which provides a stronger defense due to the large batch size, where we see robust accuracy improved than Vanilla.

Table 3. Segmentation mIoU with targeted attack on Cityscapes. We use the DRN-22-d backbone. Restoring the equivariance moves the predicted segmentation map to the groundtruth.

	mIoU to Attack Target ↓			mIoU to Groundtruth ↑		
Evaluation Method	Vanilla	Invariance	Equivariance (Ours)	Vanilla	Invariance	Equivariance (Ours)
PGD	68.03	12.92	14.96	10.08	25.10	**30.01**
MIM	71.49	13.78	12.63	9.78	23.11	**28.51**
Houdini	54.49	12.83	16.80	17.17	24.56	**30.26**
BPDA	68.03	25.64	25.82	10.08	17.73	**20.14**

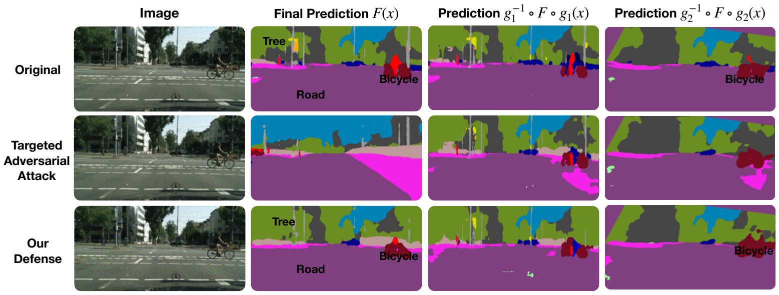
Image
Final Prediction $F(x)$
Prediction $g_1^{-1} \circ F \circ g_1(x)$
Prediction $g_2^{-1} \circ F \circ g_2(x)$
Original
Targeted Adversarial Attack
Our Defense
Tree
Road
Bicycle
Bicycle
Tree
Road
Bicycle
Bicycle

Figure 4. Our method improves robustness under targeted adversarial attacks (Random Sample). By adding targeted adversarial attacks, the model fails to predict the bicycle on the road and instead predicts a sidewalk. In the middle row, the attacked model's representation produces different segmentation maps under different transformations, suggesting that the model is no longer equivariant. By restoring the equivariance, we correct the model prediction.

4.6. Results on Non-Adversarial Trained Models

We have shown that equivariance improves the robustness of adversarially trained models. However, most pretrained models are not adversarially defended. We thus study whether our method can also improve standard models' robustness.

CityScapes Semantic Segmentation. In Table 2, we first conduct five types of attacks for the DRN-22-d segmentation model. We use 20 steps of defense, i.e., $K = 20$, and use a step size of $\eta = 2\epsilon_v$, and set the defense vector bound to be $\epsilon_v = 1.5\epsilon$. While the strongest Houdini attack can reduce the mIoU score to 0, our defense can restore the mIoU score by over **30** points. For the adaptive attack, we search the optimal λ_e that reduces the robust performance the most and find $\lambda_e = 1000$ produces the most effective attack, which still cannot bypass our defense. For baselines, we find $\lambda_e = 0$ produces the most effective attack. We find for standard backbones that are not adversarially trained, BPDA is the most effective attack, we thus only evaluate on BPDA on the following datasets. We run 10 steps of BPDA with 20 steps of reversal in the inner loop, which is a totally of 200 backward steps. Under the BPDA attack, equivariance-based defense is still more effective than other methods, including the invariance-based method.

PASCAL VOC Semantic Segmentation. We show results in Table 2. We use the pretrained DeepLabV3 (Chen et al., 2018; 2017) model. We use $K = 20$ and step size $\eta = 2\epsilon_v$, and $\epsilon_v = 1.5\epsilon$. Our approach can significantly improve the robustness compared with other methods.

MSCOCO Semantic Segmentation. We show results in Table 2. We use the pretrained DeepLabV3 (Chen et al., 2018; 2017) model. On COCO, we use $K = 2$ and step size $\eta = 2\epsilon_v$, $\epsilon_v = 1.25\epsilon$. Using equivariance outperforms other test-time defense methods.

Instance Segmentation. Our defense can also secure the more challenging instance segmentation model. In Table 2, our method improves instance segmentation maskAP by up to 21 points, which demonstrates that our method can be applied to a large number of vision applications.

Targeted Attack. The above attacks are untargeted. We also analyze whether our conclusion holds under targeted attacks, where the attacker needs to fool the model to predict a specific target. In Table 3, the targeted attack successfully misleads the model to predict the target, and our equivariance defense corrects the prediction to be the ground truth. Equivariance improves up to 10 points on the mIoU metric. We show visualizations in Figure 4.

4.7. Analysis

Equivariance Measurement. We calculate the equivariance value measured by Equation 2 for clean images, adversarial attacked images, and our defended images. We show the numerical results in Table 5. While adversarial attacks corrupt the equivariance of the image, as shown by the lowered value in the table, our method is able to restore it. Visualizations in Figure 2 also show our method clearly restores the equivariance under attack.

Ablation Study for Equivariance Transformations. In Table 6, we study the impact of using different transformations in our equivariance defense. We find transformations, which the model should be equivariant to but, in fact, does not due to attacks, are the most effective ones in improving

	Vanilla	Random	Rotation	Contrastive	Invariance	Equivariance (Ours)
Running Time (sec/sample)	0.016	0.016	0.322	0.152	1.632	1.653
Memory Usage (GB)	0.391	0.391	3.102	0.731	10.049	10.357

Table 4. Running time and GPU memory usage for MS COCO semantic segmentation task. We evaluate on a single A6000 GPU.

	Dataset			
	ImageNet	Cityscapes	PASAL VOC	COCO
Clean Images	0.539	0.694	0.900	0.901
Attacked Images	0.538	0.448	0.642	0.774
Restored Images	**0.581**	**0.713**	**0.921**	**0.914**

Table 5. Measurement for equivariance on clean images, attacked images, and our restored images. A high score indicates better equivariance. Adversarial attack corrupts the equivariance. Our method restores the equivariance back to the same level as the clean images.

	Transformations of Equivariance			
Loss	Flip	Resize	Rotation$\leq 15°$	Rotation$\geq 90°$
Invariance	9.56	9.90	9.75	**9.60**
Equivariance	**20.50**	**26.00**	**17.03**	8.61

Table 6. The impact of using different transformations on the performance of our method. We show results from a standard segmentation model on Cityscape.

robustness. For example, flipping and resizing are most effective for our studied semantic segmentation. Rotation below 15 degrees helps robustness more than rotation larger than 90 degrees. Large rotation performs worse because segmentation models are not equivariant to large rotation, even on clean data, which reduces the effectiveness of our approach. In Section 4.4, we empirically choose the combination of transformations that produces good empirical results for our approach.

The Trade-off between Robustness and Clean Accuracy. In Table 7, we show that increasing the bound ϵ_v for the defense vector creates a trade-off between clean accuracy and robust accuracy. Specifically, bound $\epsilon_v = 1/255$ is a sweet spot, where one can increase robustness by 0.4 without any loss of clean accuracy. Our method allows dynamically conducting trade-off between robustness and clean accuracy by controlling the additive vector's bound.

Runtime Analysis and GPU Memory Usage. In Table 4, we show the running time and GPU memory usage on our studied methods. While our method leads to longer running time and larger GPU memory usage, we believe this is a necessary trade-off to achieve the best robustness. In many important applications, sacrificing accuracy or robustness for the sake of reducing running time/memory usage would be counterproductive. To mitigate this, we also propose to first detect adversarial examples, then only perform our test-time adaptation for the detected adversarial ones.

Detecting Adversarial Samples. A straightforward way to speed up our inference and improve the accuracy on clean samples, is to first detect adversarial samples, and only run our algorithm on the adversarial samples. Table 8 reports the time running on COCO images with a single A6000 GPU, which shows that detection is less expensive compared to our defense and can be used to reduce our computational cost. Since test-time optimization on clean examples decrease clean performance, we can also increase the clean accuracy by first detecting the adversarial examples. In Appendix A.3.1, we show that we can increase clean performance by only performing test-time optimization on the detected adversaries.

	Equivariance defense vector bound $\epsilon_v = i/255$						
Accuracy	i=0	i=1	i=2	i=4	i=6	i=8	i=10
Clean	53.23	**53.24**	53.09	52.38	50.62	48.84	48.74
Robustness	26.61	27.03	27.57	28.53	29.22	29.57	**29.83**

Table 7. Trading-off Robustness vs. Clean Accuracy on Cityscape using our equivariance method under BPDA attack. If clean performance is important, we can simply decrease the defense vector bound to increase the clean accuracy.

	Method			
	Rotation	Contrastive	Invariance	Equivariance (Ours)
Inference (sec)	0.016	0.016	0.016	0.016
Detection (sec)	0.048	0.049	0.147	0.169
Defense (sec)	0.306	0.136	1.616	1.637

Table 8. Running time for different methods on vanilla feedforward inference (Inference), detecting adversarial samples (Detection), and our test-time defense (defense).

5. Conclusion

Robust perception under adversarial attacks has been an open challenge. We find that equivariance can be a desired structure to maintain at inference time because it can provide dense structural constraints on a fine-grained level. By dynamically restoring equivariance at inference, we show significant improvement in adversarial robustness across three datasets. Our work hints toward a new direction that uses the right structural information at inference time to improve robustness.

Acknowledgement

This research is based on work partially supported by the DARPA SAIL-ON program, the NSF NRI award #1925157, a GE/DARPA grant, a CAIT grant, and gifts from JP Morgan, DiDi, and Accenture. We thank the anonymous reviewers for their valuable feedback in improving the paper.

References

Rabbit and duck illusion. https://en.wikipedia.org/wiki/Rabbit-duck_illusion.

Arnab, A., Miksik, O., and Torr, P. H. On the robustness of semantic segmentation models to adversarial attacks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 888–897, 2018.

Athalye, A., Carlini, N., and Wagner, D. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International conference on machine learning*, pp. 274–283. PMLR, 2018.

Barnard, E. and Casasent, D. Invariance and neural nets. *IEEE Transactions on neural networks*, 2(5):498–508, 1991.

Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., and Raffel, C. A. Mixmatch: A holistic approach to semi-supervised learning. *Advances in neural information processing systems*, 32, 2019.

Carlini, N. and Wagner, D. A. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy*, pp. 39–57, 2017.

Carmon, Y., Raghunathan, A., Schmidt, L., Duchi, J. C., and Liang, P. S. Unlabeled data improves adversarial robustness. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Chaman, A. and Dokmanic, I. Truly shift-invariant convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3773–3783, 2021.

Chaman, A. and Dokmanić, I. Truly shift-equivariant convolutional neural networks with adaptive polyphase upsampling. In *2021 55th Asilomar Conference on Signals, Systems, and Computers*, pp. 1113–1120. IEEE, 2021.

Chen, L.-C., Papandreou, G., Schroff, F., and Adam, H. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 801–818, 2018.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Cisse, M., Adi, Y., Neverova, N., and Keshet, J. Houdini: Fooling deep structured prediction models. *arXiv preprint arXiv:1707.05373*, 2017a.

Cisse, M., Bojanowski, P., Grave, E., Dauphin, Y., and Usunier, N. Parseval networks: Improving robustness to adversarial examples. In *International Conference on Machine Learning*, pp. 854–863. PMLR, 2017b.

Cohen, T. and Welling, M. Group equivariant convolutional networks. In *International conference on machine learning*, pp. 2990–2999. PMLR, 2016a.

Cohen, T. S. and Welling, M. Steerable cnns. *arXiv preprint arXiv:1612.08498*, 2016b.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3213–3223, 2016.

Croce, F. and Hein, M. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *ICML*, 2020.

Croce, F., Gowal, S., Brunner, T., Shelhamer, E., Hein, M., and Cemgil, T. Evaluating the adversarial robustness of adaptive test-time defenses. *arXiv preprint arXiv:2202.13711*, 2022.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

Dieleman, S., De Fauw, J., and Kavukcuoglu, K. Exploiting cyclic symmetry in convolutional neural networks. In *International conference on machine learning*, pp. 1889–1898. PMLR, 2016.

Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., and Li, J. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9185–9193, 2018.

Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88 (2):303–338, 2010.

Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *ICLR*, 2019.

Gidaris, S., Singh, P., and Komodakis, N. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018.

Gupta, D. K., Arya, D., and Gavves, E. Rotation equivariant siamese networks for tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12362–12371, 2021.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. In *ICLR*, 2019.

Hendrycks, D., Mazeika, M., Kadavath, S., and Song, D. Using self-supervised learning can improve model robustness and uncertainty. *Advances in Neural Information Processing Systems*, 32, 2019.

Hendrycks, D., Basart, S., Mu, N., Kadavath, S., Wang, F., Dorundo, E., Desai, R., Zhu, T., Parajuli, S., Guo, M., Song, D., Steinhardt, J., and Gilmer, J. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *ICCV*, 2021.

Kamath, S., Deshpande, A., Kambhampati Venkata, S., and N Balasubramanian, V. Can we have it all? on the tradeoff between spatial and adversarial robustness of neural networks. *Advances in Neural Information Processing Systems*, 34, 2021.

Kumar, A., Levine, A., Feizi, S., and Goldstein, T. Certifying confidence via randomized smoothing. *Advances in Neural Information Processing Systems*, 33:5165–5177, 2020.

Laptev, D., Savinov, N., Buhmann, J. M., and Pollefeys, M. Ti-pooling: transformation-invariant pooling for feature learning in convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 289–297, 2016.

Lawhon, M., Mao, C., and Yang, J. Using multiple self-supervised tasks improves model robustness. *arXiv preprint arXiv:2204.03714*, 2022.

Lee, H., Lee, K., Lee, K., Lee, H., and Shin, J. Improving transferability of representations via augmentation-aware self-supervision. *Advances in Neural Information Processing Systems*, 34, 2021.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Mahajan, D., Tople, S., and Sharma, A. Domain generalization using causal matching. In *International Conference on Machine Learning*, pp. 7313–7324. PMLR, 2021.

Mao, C., Zhong, Z., Yang, J., Vondrick, C., and Ray, B. Metric learning for adversarial robustness. *Advances in Neural Information Processing Systems*, 32, 2019.

Mao, C., Gupta, A., Nitin, V., Ray, B., Song, S., Yang, J., and Vondrick, C. Multitask learning strengthens adversarial robustness. In *European Conference on Computer Vision*, pp. 158–174. Springer, 2020.

Mao, C., Chiquier, M., Wang, H., Yang, J., and Vondrick, C. Adversarial attacks are reversible with natural supervision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 661–671, 2021.

Mao, C., Geng, S., Yang, J., Wang, X., and Vondrick, C. Understanding zero-shot adversarial robustness for large-scale models. *arXiv preprint arXiv:2212.07016*, 2022.

Marcos, D., Volpi, M., and Tuia, D. Learning rotation invariant convolutional filters for texture classification. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 2012–2017. IEEE, 2016.

McDermott, N. T., Yang, J., and Mao, C. Robustifying language models with test-time adaptation. In *ICLR 2023 Workshop on Pitfalls of limited data and computation for Trustworthy ML.*

Mi, L., Wang, H., Tian, Y., and Shavit, N. Training-free uncertainty estimation for neural networks. In *AAAI*, 2022.

Pastore, G., Cermelli, F., Xian, Y., Mancini, M., Akata, Z., and Caputo, B. A closer look at self-training for zero-label semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2693–2702, 2021.

Rice, L., Wong, E., and Kolter, Z. Overfitting in adversarially robust deep learning. In *International Conference on Machine Learning*, pp. 8093–8104. PMLR, 2020.

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.

Shi, C., Holtz, C., and Mishne, G. Online adversarial purification based on self-supervised learning. In *International Conference on Learning Representations*, 2020.

Sosnovik, I., Szmaja, M., and Smeulders, A. Scale-equivariant steerable networks. *arXiv preprint arXiv:1910.11093*, 2019.

Sun, Y., Wang, X., Liu, Z., Miller, J., Efros, A., and Hardt, M. Test-time training with self-supervision for generalization under distribution shifts. In *International conference on machine learning*, pp. 9229–9248. PMLR, 2020.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

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.

Tsai, Y.-Y., Mao, C., Lin, Y.-K., and Yang, J. Self-supervised convolutional visual prompts. *arXiv preprint arXiv:2303.00198*, 2023.

Wang, D., Shelhamer, E., Liu, S., Olshausen, B., and Darrell, T. Tent: Fully test-time adaptation by entropy minimization. *ICLR*, 2021.

Wang, H., Mao, C., He, H., Zhao, M., Jaakkola, T. S., and Katabi, D. Bidirectional inference networks: A class of deep bayesian networks for health profiling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 766–773, 2019.

Weiler, M. and Cesa, G. General e (2)-equivariant steerable cnns. *Advances in Neural Information Processing Systems*, 32, 2019.

Weiler, M., Hamprecht, F. A., and Storath, M. Learning steerable filters for rotation equivariant cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 849–858, 2018.

Welling, M. and Teh, Y. W. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pp. 681–688. Citeseer, 2011.

Wong, E., Rice, L., and Kolter, J. Z. Fast is better than free: Revisiting adversarial training, 2020.

Yu, F., Koltun, V., and Funkhouser, T. Dilated residual networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 472–480, 2017.

Yun, S., Han, D., Oh, S. J., Chun, S., Choe, J., and Yoo, Y. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6023–6032, 2019.

Zamir, A. R., Sax, A., Cheerla, N., Suri, R., Cao, Z., Malik, J., and Guibas, L. J. Robust learning through cross-task consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11197–11206, 2020.

Zhang, H., Yu, Y., Jiao, J., Xing, E., El Ghaoui, L., and Jordan, M. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pp. 7472–7482. PMLR, 2019.

Zhang, R. Making convolutional networks shift-invariant again. In *International conference on machine learning*, pp. 7324–7334. PMLR, 2019.

A. Appendix.

A.1. Theoretical Results for Adversarial Robustness

We now show detailed proof for Lemma 1 and theorem 2.

Lemma A.1. *The standard classifier under adversarial attack is equivalent to predicting with $P(\mathbf{Y}|\mathbf{X}_a, y_a^{(s_1)}, y_a^{(s_2)}, ..., y_a^{(s_k)})$, and our approach is equivalent to predicting with $P(\mathbf{Y}|\mathbf{X}_a, y^{(s_1)}, y^{(s_2)}, ..., y^{(s_k)})$.*

Proof. For the standard classifier under attack, we know that $P(y_a^{(s_1)}, y_a^{(s_2)}, ..., y_a^{(s_k)}|\mathbf{X} = \mathbf{x}_a) = 1$. Thus we know the standard classifier under adversarial attack is equivalent to

$$\begin{aligned} P(\mathbf{Y}|\mathbf{X} = \mathbf{x}_a) &= \sum_{y_a^{(s_1)}, y_a^{(s_2)}, ..., y_a^{(s_k)}} P(y_a^{(s_1)}, y_a^{(s_2)}, ..., y_a^{(s_k)}|\mathbf{X} = \mathbf{x}_a) P(\mathbf{Y}|y_a^{(s_1)}, y_a^{(s_2)}, ..., y_a^{(s_k)}, \mathbf{X} = x_a) \\ &= P(\mathbf{Y}|y_a^{(s_1)}, y_a^{(s_2)}, ..., y_a^{(s_k)}, \mathbf{X} = \mathbf{x}_a). \end{aligned}$$

Our algorithm finds a new input image $\mathbf{x}_{\text{max}}^{(n)}$ that

$$\begin{aligned} &\underset{\mathbf{x}^{(n)}}{\text{argmax}}\, P(\mathbf{X}^{(n)} = \mathbf{x}^{(n)}|\mathbf{X} = \mathbf{x}_a) P(y^{(s_1)}, y^{(s_2)}, ..., y^{(s_k)}|\mathbf{X}^{(n)} = \mathbf{x}^{(n)}) \\ &= \underset{\mathbf{x}^{(n)}}{\text{argmax}}\, P(\mathbf{X}^{(x)} = \mathbf{x}^{(n)}|\mathbf{X} = \mathbf{x}_a, y^{(s_1)}, y^{(s_2)}, ..., y^{(s_k)}). \end{aligned}$$

Our algorithm first estimate $\mathbf{x}_{\text{max}}^{(n)}$ with adversarial image $\mathbf{x}_a$ and self-supervised label $\mathbf{y}^{(s)}$. We then predict the label $\mathbf{Y}$ using our new image $\mathbf{x}_{\text{max}}^{(n)}$. Thus, our approach in fact estimates $P(\mathbf{Y}|\mathbf{X}^{(n)} = \mathbf{x}_{\text{max}}^{(n)}) P(\mathbf{X}^{(n)} = \mathbf{x}_{\text{max}}^{(n)}|\mathbf{X} = \mathbf{x}_a, y^{(s_1)}, y^{(s_2)}, ..., y^{(s_k)})$. Note the following holds:

$$\begin{aligned} &P(\mathbf{Y}|\mathbf{X} = \mathbf{x}_a, y^{(s_1)}, y^{(s_2)}, ..., y^{(s_k)}) \\ &= \sum_{\mathbf{x}^{(n)}} P(\mathbf{Y}|\mathbf{x}^{(n)}) P(\mathbf{x}^{(n)}|\mathbf{X} = \mathbf{x}_a, y^{(s_1)}, y^{(s_2)}, ..., y^{(s_k)}) \\ &\approx P(\mathbf{Y}|\mathbf{X}^{(n)} = \mathbf{x}_{max}^{(n)}) P(\mathbf{X}^{(n)} = \mathbf{x}_{max}^{(n)}|\mathbf{X} = \mathbf{x}_a, y^{(s_1)}, y^{(s_2)}, ..., y^{(s_k)}). \end{aligned}$$

Thus our approach is equivalent to estimating $P(\mathbf{Y}|\mathbf{X} = \mathbf{x}_a, y^{(s_1)}, y^{(s_2)}, ..., y^{(s_k)})$. □

We use the maximum a posteriori (MAP) estimation $\mathbf{x}_{max}^{(n)}$ to approximate the sum over $\mathbf{X}^{(n)}$ because: (1) sampling a large number of $\mathbf{X}^{(n)}$ is computationally expensive; (2) our results show that random sampling is ineffective; (3) our MAP estimate naturally produces a denoised image that can be useful for other downstream tasks.

Theorem A.2. *Assume the classifier operates better than chance and instances in the dataset are uniformly distributed over n categories. Let the prediction accuracy bounds be $P(\mathbf{Y}|y_a^{(s_1)}, y_a^{(s_2)}, ..., y_a^{(s_k)}, \mathbf{X}_a) \in [b_0, c_0]$, $P(\mathbf{Y}|y^{(s_1)}, \mathbf{X}_a) \in [b_1, c_1]$, $P(\mathbf{Y}|y^{(s_1)}, y^{(s_2)}, \mathbf{X}_a) \in [b_2, c_2]$, ..., and $P(\mathbf{Y}|y^{(s_1)}, y^{(s_2)}, ..., y^{(s_k)}, \mathbf{X}_a) \in [b_k, c_k]$. If the conditional mutual information $I(\mathbf{Y}; y^{(s_i)}|\mathbf{X}_a) > 0$ and $I(\mathbf{Y}; y^{(s_i)}|\mathbf{X}_a, y^{(s_j)}) > 0$ where $i \neq j$, we have $b_0 \leq b_1 \leq ... \leq b_k$ and $c_0 < c_1 < c_2 < ... < c_k$, which means our approach strictly improves the bound for classification accuracy.*

Proof. If $I(\mathbf{Y}; y^{(s_i)}|\mathbf{X} = \mathbf{x}_a) > 0$, and $I(\mathbf{Y}; y^{(s_i)}|\mathbf{X}_a, y^{(s_j)}) > 0$ where $i \neq j$, then it is straight-forward that:

$$I(\mathbf{Y}; y^{(s_1)}, y^{(s_2)}, ..., y^{(s_k)}, \mathbf{X}_a) > I(\mathbf{Y}; y^{(s_i)}, \mathbf{X}_a) > I(\mathbf{Y}; y_a^{(s_i)}, \mathbf{X}_a) = I(\mathbf{Y}; \mathbf{X}_a).$$

$$I(\mathbf{Y}; y^{(s_1)}, y^{(s_2)}, ..., y^{(s_k)}, \mathbf{X}_a) > I(\mathbf{Y}; y_a^{(s_1)}, y_a^{(s_2)}, ..., y_a^{(s_k)}, \mathbf{X}_a) = I(\mathbf{Y}; \mathbf{X}_a).$$

We let the predicted label be $\widehat{\mathbf{Y}}$, we assume there are n categories, and let the lower bound for prediction accuracy to be $Pr(\widehat{\mathbf{Y}} = \mathbf{Y}) \geq 1 - \epsilon_p$. We define $H(\epsilon_p) = -\epsilon_p log \epsilon_p - (1 - \epsilon_p) log(1 - \epsilon_p)$. Using the *Fano's Inequality*, we have

$$H(\mathbf{Y}|\mathbf{X}_a) \leq H(\epsilon_p) + \epsilon_p \cdot \log(n - 1) \tag{7}$$

$$-\epsilon_p \cdot \log(n-1) \le H(\epsilon_p) - H(\mathbf{Y}|\mathbf{X}_a) \tag{8}$$

We add $H(\mathbf{Y})$ to both side

$$H(\mathbf{Y}) - \epsilon_p \cdot \log(n-1) \le H(\epsilon_p) + I(\mathbf{Y};\mathbf{X}_a) \tag{9}$$

because $I(\mathbf{Y};\mathbf{X}_a) = H(\mathbf{Y}) - H(\mathbf{Y}|\mathbf{X}_a)$.

Then we get

$$H(\epsilon_p) + \epsilon_p \log(n-1) \ge -I(\mathbf{Y};\mathbf{X}_a) + H(\mathbf{Y}) \tag{10}$$

Now we define a new function $G(\epsilon_p) = H(\epsilon_p) + \epsilon_p log(n-1)$. Given that in the classification task, the number of categories $n \ge 2$. We know $log(n-1) \ge 0$. Given that the entropy function $H(\epsilon_p)$ first increase and then decrease, the function $G(\epsilon_p)$ should also first increase, peak at some point, and then decrease.

We calculate the ϵ_p for the peak value via calculating the first order derivative $G'(\epsilon_p) = 0$. By solving this, we have:

$$\epsilon_p = 1 - \tfrac{1}{n} \tag{11}$$

which shows that the function $G(\epsilon_p)$ is monotonically increasing when $\epsilon_p \in [0, 1 - \frac{1}{n}]$.

Given that we know, the base classifier already achieves accuracy better than random guessing, thus the given classifier satisfies $\epsilon_p \in [0, 1 - \frac{1}{n}]$.

Now, the function $G(\epsilon_p) = H(\epsilon_p) + \epsilon_p log(n-1)$ is a monotonically increasing function in our studied region, which has the inverse function G^{-1}.

By rewritting the equation 10 We then have

$$G(\epsilon_p) \ge -I(\mathbf{Y};\mathbf{X}_a) + H(\mathbf{Y}) \tag{12}$$

We apply the inverse function G^{-1} to both side:

$$\epsilon_p \ge G^{-1}(-I(\mathbf{Y};\mathbf{X}_a) + H(\mathbf{Y})) \tag{13}$$

$$1 - \epsilon_p \le 1 - G^{-1}(-I(\mathbf{Y};\mathbf{X}_a) + H(\mathbf{Y})) \tag{14}$$

Note that $(1 - \epsilon_p)$ is our defined accuracy. Similarly, we have:

$$\begin{aligned}
&1 - \epsilon_p \le c_1 = 1 - Q^{-1}(-I(\mathbf{Y};\mathbf{X}_a) + H(\mathbf{Y})),\\
&1 - \epsilon_p \le c_2 = 1 - Q^{-1}(-I(\mathbf{Y};y^{(s)},\mathbf{X}_a) + H(\mathbf{Y})),\\
&\quad\ldots\\
&1 - \epsilon_p \le c_k = 1 - Q^{-1}(-I(\mathbf{Y};y^{(s_1)},y^{(s_2)},...,y^{(s_k)},\mathbf{X}_a) + H(\mathbf{Y})),
\end{aligned}$$

where the upper bound is a function of the mutual information. Since $H(\mathbf{Y})$ is a constant, a larger mutual information will strictly increase the bound. Thus, $c_0 < c_1 < c_2 < ... < c_k$.

In addition, the lower bound will not get worse given the additional information. Thus $b_0 \le b_1 \le ... \le b_k$ and.

□

A.2. Detection

Anomaly detection, also referred to as novelty detection or outlier detection, predicts the data when the model is uncertain about a deviated model. (Ruff et al., 2018) conducts anomaly detection by training a binary classifier on in-distribution and collected out-of-distribution data, however, it is hard to foresee the out-of-distribution data. (Hendrycks et al., 2019; Gidaris et al., 2018; Tack et al., 2020) need to train the model with self-supervision first and then perform OOD detection using the performance of the self-supervision task. In this paper, we will focus on the training-free method that uses sensitivity to estimate the uncertainty of the model (Mi et al., 2022).

A.2.1. Equivariance for Anomaly Detection

Let $\mathbf{y}$ be the ground-truth category labels for $\mathbf{x}$. Let the network that uses the feature $\mathbf{h}$ for final task prediction to be $C_{\theta'}$. For prediction, neural networks learn to predict the category $\widehat{\mathbf{y}} = C_{\theta'} \circ F_\theta(\mathbf{x})$ by minimizing the loss $L(\widehat{\mathbf{y}}, \mathbf{y})$ between the predictions and the ground truth. For example, for semantic segmentation, L is cross-entropy for each pixel output; for depth prediction, L is an L1 loss for each depth pixel prediction. We define the loss for the final task as follows:

$$\mathcal{L}_t(\mathbf{x}, \mathbf{y}) = L\left(C_{\theta'} \circ F_\theta(\mathbf{x}), \mathbf{y}\right), \tag{15}$$

As shown in Figure 6, **??**, and 7, when the model is uncertain about the input and makes the wrong prediction, it is often *less equivariant*. Work (Tack et al., 2020; Hendrycks et al., 2019) has shown that self-supervision tasks perform worse when the model is uncertain. Thus, we propose to use the equivariance of the output $\widehat{\mathbf{y}} = C_{\theta'} \circ F_\theta(\mathbf{x})$ for anomaly detection. We calculate the variance of the output after transformations g_i:

$$\mathcal{L}_{equi}^{output} = \sum_i ||g_i^{-1} \circ C_{\theta'} \circ F_\theta \circ g_i(\mathbf{x}) - C_{\theta'} \circ F_\theta(\mathbf{x})||^2, \tag{16}$$

where we use $C_{\theta'} \circ F_\theta(\mathbf{x})$ as the surrogate mean prediction. Here larger variance indicates less equivariance and therefore higher probability that $\mathbf{x}$ is an out-of-sample data point (see details in sec:theory$_{a}no$).

A.2.2. Theoretical Results for Anomaly detection with Multiple Constraints

Below we provide theoretical analysis on why the equivariant loss can be used for anomaly detection. For each pixel in an image, we denote as X and Y the input pixel and target label. We use $Z_0 = \widetilde{F}_\theta(X)$ and $Z = \widetilde{g}^{-1} \circ \widetilde{F}_\theta \circ \widetilde{g}(X)$ to denote the model predictions of the original and transformed input, respectively, where $\widetilde{g}$ and $\widetilde{F}_\theta$ are the associated pixel operations for g and F_θ. Correspondingly $e = |Z_0 - Y|$ is the error of the model's prediction for the pixel. Note that $\widetilde{g}(\cdot)$ and $\widetilde{g}^{-1}(\cdot)$ are equivariant transformations. There can be multiple equivariant transformations $\widetilde{g}_i(\cdot)$ for the same input pixel X, leading to different model predictions. For the same input pixel X, we then denote as $\mu(X) = \mathbb{E}_{\widetilde{g}}[Z|X]$ and $\sigma(X)^2 = \mathbb{V}_{\widetilde{g}}[Z|X]$ the mean and variance of the model predictions over different equivariant transformations. Here, $\sigma(X)$ measures the *sensitivity* of the model for the input pixel X; below we use a shorthand σ for $\sigma(X)$ when the context is clear. Following (Mi et al., 2022), we now introduce our model-agnostic assumptions below.

Assumption A.3 (Heterogeneous Perturbation). $\epsilon_1 = \frac{Z_0 - \mu(X)}{\sigma(X)} \sim \mathcal{N}(0, 1)$. That is, the model prediction given the original input X behaves like a random Gaussian draw from the model predictions produced by different equivariant transformations.

Assumption A.4 (Random Bias). $\epsilon_2 = Y - \mu(X) \sim \mathcal{N}(0, B^2)$. That is, the bias of the model prediction behaves like Gaussian noise with bounded variance B^2.

Assuming each image contains n input pixels, $\{X_i\}_{i=1}^n$, we have the corresponding target labels $\{Y_i\}_{i=1}^n$, errors $\{e_i\}_{i=1}^n$, and sensitivity $\{\sigma_i\}_{i=1}^n$. We denote as $\bar{e} = \frac{1}{n}\sum_i e_i$ the average error of an image. As is usually the case, we further assume the errors are bounded, i.e., $a \le e_i \le b$. Our goal is to bound the average pixel error $\bar{e}$ for an image using the image's computed uncertainty (sensitivity) score $\{\sigma_i\}_{i=1}^n$. With the assumptions above, we have:

Theorem A.5 (Estimator for $\bar{e}$)**.** *With probability at least δ, one can estimate the average error $\bar{e}$ for an image using $\sqrt{\frac{2}{\pi}}\mathbb{E}[\sigma_B]$ with the following guarantee:*

$$|\bar{e} - \sqrt{\tfrac{2}{\pi}}\mathbb{E}[\sigma_B]| < \tfrac{b-a}{\sqrt{n}}\sqrt{\ln\tfrac{1}{\delta}},$$

where $\sigma_B \triangleq \sqrt{\sigma^2 + B^2}$ is the smoothed version of the uncertainty (sensitivity) σ and B is a constant from Assumption A.4.

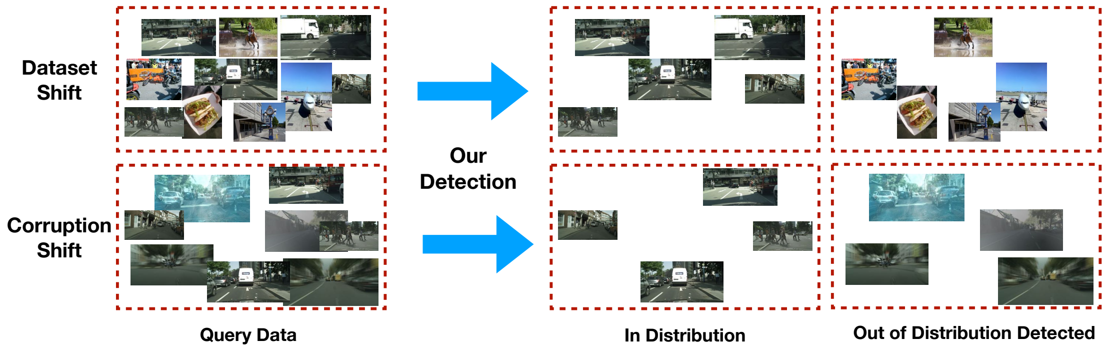

Figure 5. Using equivariance can detect both dataset shifting and corruption shifting.

Table 9. AUROC (multiplied by 100) of anomaly detection on corrupted images. Our equivariance method achieves better detection efficiency over 15 types of corruptions.

	Cityscape														
Model	Gauss	Shot	Impul	Defoc	Glass	Motion	Zoom	Snow	Frost	Fog	Bright	Cont	Elast	Pixel	JPEG
Rot (Hendrycks et al., 2019)	57	54	49	43	55	35	44	39	64	51	46	52	42	44	59
CSI (Tack et al., 2020)	67	67	62	65	62	57	64	45	55	63	63	65	55	53	54
Inv	99	99	99	100	94	87	80	86	95	93	86	98	52	98	100
Ours	**100**	**100**	**100**	**100**	**99**	**98**	**99**	**100**	**100**	**98**	**94**	**100**	**77**	**99**	**100**

	PASCAL VOC														
Model	Gauss	Shot	Impul	Defoc	Glass	Motion	Zoom	Snow	Frost	Fog	Bright	Cont	Elast	Pixel	JPEG
Rot (Hendrycks et al., 2019)	37	39	39	55	53	54	54	43	49	55	51	55	50	52	50
CSI (Tack et al., 2020)	49	51	50	55	55	58	54	61	58	58	52	72	53	54	54
Inv	**99**	**99**	**99**	66	21	36	37	70	74	68	54	66	45	35	40
Ours	98	98	98	**96**	**95**	**91**	**93**	**85**	**86**	**81**	**60**	**92**	**70**	**85**	**75**

	MSCOCO														
Model	Gauss	Shot	Impul	Defoc	Glass	Motion	Zoom	Snow	Frost	Fog	Bright	Cont	Elast	Pixel	JPEG
Rot (Hendrycks et al., 2019)	95	95	95	93	93	93	93	93	94	93	93	93	93	93	92
CSI (Tack et al., 2020)	88	89	86	76	79	77	75	42	47	44	77	22	84	82	85
Inv	98	98	98	96	97	96	96	98	98	97	98	98	97	97	97
Ours	**98**	**98**	**98**	**98**	**98**	**98**	**98**	**98**	**98**	**98**	**98**	**98**	**98**	**98**	**98**

Estimated Average Error for Anomaly Detection. We can see that $\sqrt{\frac{2}{\pi}}\mathbb{E}[\sigma_B]$ can be a good estimate for the average error for an image, and that this estimate gets more accurate as n gets larger. Therefore, it can be used directly for anomaly detection; larger $\sqrt{\frac{2}{\pi}}\mathbb{E}[\sigma_B]$ indicates a potentially larger error, meaning that the image is more likely to be an anomaly. Note that the expectation $\mathbb{E}[\sigma_B]$ is over the space of pixels in all images governed by the assumptions above, not over the pixels in a specific image. In practice, we estimate $\mathbb{E}[\sigma_B]$ by averaging over the pixel-level sensitivity in an image, i.e., $\frac{1}{n}\sum_i \sqrt{\sigma_i^2 + B^2}$, leading to Eqn. 16.

Extension to the Multivariate Case. Theorem A.6 assumes one scalar output for *each pixel* in an image; this is directly applicable for dense regression tasks, e.g., depth estimation. For dense classification tasks, e.g., segmentation, the label for each pixel is represented by a one-hot vector. Fortunately, Theorem A.6 can be naturally extended to the multivariate case, and therefore works for both regression and classification tasks. Note that in classification tasks, $\mu(X)$ and $\sigma(X)$ are both real-valued vectors where all entries in a vector sums up to 1, while Y and Z_0 are both one-hot vectors (vectors with one entry equal to 1 and others equal to 0); therefore the two mild assumptions above are still reasonable.

	Rotation			Contrast			Invariance			Equivariance		
Model	CI	VO	CO	CI	VO	CO	CI	VO	CO	CI	VO	CO
CI	-	56	51	-	72	52	-	93	91	-	**98**	**91**
VO	67	-	61	54	-	85	**95**	-	**91**	71	-	88
CO	**72**	93	-	69	83	-	51	96	-	55	**99**	-

Table 10. AUROC for out of dstribution detection. The rows are the source data that the models have been trained on. The columns are the data where the OOD are sampled from.

A.2.3. Robustness on Anomaly Detection

Dataset and Tasks. We conduct anomaly detection experiments with 15 common corruptions (Hendrycks & Dietterich, 2019) on Cityscapes, PASCAL VOC, and MSCOCO. We also study whether using equivariance can detect examples from a different dataset.

Baselines. CSI (Tack et al., 2020) uses contrastive loss as indicator for novelty detection. Hendrycks et al. (Hendrycks et al., 2019) (Rot) uses rotation task's performance for detection. Invariance (Inv) uses the consistency between different views of the same image for anomaly detection, which uses the same setup as our equivariance method except for the reversed transformation g^{-1} in feature space.

Results. We show visualization of the task in Figure 5. Table 9 shows the detection performance on *corrupted images*. Our approach in general improves the AUROC score over the baselines, achieving up to 15 AUROC points improvement, which demonstrates the corruption detection of our approach. We show results on detecting *dataset shifting* in Table 10. We denote Cityscapes, PASCAL VOC, and COCO as CI, VO, and CO, respectively. Each row indicates the source model that we trained on, and each column is the OOD sampled to detect. Our method in general achieves better out-of-distribution detection efficiency over the existing approaches.

A.2.4. Theoretical Results for Anomaly Detection

In the main paper, we provide theoretical analysis on why the equivariant loss can be used for anomaly detection. We now show detailed proof for our Theorem 1.

Theorem A.6 (Estimator for $\bar{e}$)**.** *With probability at least δ, one can estimate the average error $\bar{e}$ for an image using $\sqrt{\frac{2}{\pi}}\mathbb{E}[\sigma_B]$ with the following guarantee:*

$$|\bar{e} - \sqrt{\tfrac{2}{\pi}}\mathbb{E}[\sigma_B]| < \tfrac{b-a}{\sqrt{n}}\sqrt{\ln\tfrac{1}{\delta}},$$

where $\sigma_B \triangleq \sqrt{\sigma^2 + B^2}$.

Proof. By the law of total expectation, we have

$$\begin{aligned}\mathbb{E}[e] &= \mathbb{E}_\sigma\mathbb{E}[e|\sigma] = \mathbb{E}_\sigma\mathbb{E}\left[|\sigma\epsilon_1 - \epsilon_2||\sigma\right] \\ &= \mathbb{E}_\sigma\mathbb{E}\left[|\mathcal{N}(0, \sigma^2 + B^2)||\sigma\right] \\ &= \sqrt{\tfrac{2}{\pi}}\mathbb{E}_\sigma[\sqrt{\sigma^2 + B^2}] \\ &\triangleq \sqrt{\tfrac{2}{\pi}}\mathbb{E}[\sigma_B],\end{aligned}$$

where $\sigma_B \triangleq \sqrt{\sigma^2 + B^2}$. Defining the total error for an image of n pixels $S_n = \sum_{i=1}^n e_i$ and by Hoeffding's inequality, we then have

$$P(|S_n - \mathbb{E}[S_n]| \geq t) \leq \exp(-\tfrac{t^2}{n(b-a)^2}), \tag{17}$$

where

$$\mathbb{E}[S_n] = \sum_{i=1}^n \mathbb{E}[e_i] = n\sqrt{\tfrac{2}{\pi}}\mathbb{E}[\sigma_B]. \tag{18}$$

Combining Eqn. 18 and Eqn. 17, we have

$$P(|\bar{e} - \sqrt{\tfrac{2}{\pi}}\mathbb{E}[\sigma_B]| \geq \tfrac{t}{n}) \leq \exp(-\tfrac{t^2}{n(b-a)^2}),$$

where $\bar{e}$ is the average error of an image. Setting $\delta = \exp(-\frac{t^2}{n(b-a)^2})$, we then have that with probability at least δ,

$$|\bar{e} - \sqrt{\tfrac{2}{\pi}}\mathbb{E}[\sigma_B]| < \tfrac{b-a}{\sqrt{n}}\sqrt{\ln\tfrac{1}{\delta}}.$$

□

	Method					
	Vanilla	Random	Rotation	Contrastive	Invariance	Equivariance (Ours)
Cityscape	58.29	55.33	49.82	50.85	36.94	51.13
Pascal	69.52	69.24	49.07	68.22	66.58	63.05
COCO	63.02	63.03	61.69	56.76	56.06	58.71
COCO Instance	34.5	33.6	33.3	29.8	26.7	34.5

Table 11. Clean accuracy by first detecting adversarial example. By avoiding running test-time optimization on clean examples, we can largely improve clean accuracy.

A.3. Additional Analysis

A.3.1. Preserving clean accuracy by detecting first

In deploying our defense, one can further preserve the accuracy on clean images by a detect-then-defend algorithm described below. Based on our findings that clean images and attacked images have a large difference in the average equivariance score (Table 5), we can set a threshold value to determine whether or not deploy our defense for a potentially attacked image based on its equivariance score. Experimental results are reported below in Table 11. As shown in the table, clean accuracy can be preserved to a large degree without significant reduction in defense performance.

A.3.2. Ablation studies on optimizer

In the paper, we use the SGLD optimizer which add noise during optimization. We compare the performance of using SGLD and SGD optimizer without noise for our defense in the Table 12 below. While our method is both effective with both optimization algorithm, SGLD achieves higher robustness.

	MS COCO	
	Equivariance SGLD	Equivariance SGD
PGD	**24.51**	16.68

Table 12. Effect of optimizer

A.3.3. Runtime analysis

We report inference speed of our defense in Table 13. It is worth noting that if we deploy detection first, as mentioned in section A.3.1, the defense will skip the majority of clean images and will not have a large effect on runtime. For attacked images, our algorithm is 40 times slower due to the test-time optimization. Given the rare cases of adversarial examples, this delay is reasonable. We can spend more time on the hard adversarial examples, as there is no point of making the wrong predictions only to make it fast.

	MS COCO	
	Vanilla	Equivariance
runtime (s)	0.046	1.699

Table 13. Runtime analysis. Results are measured on a single A6000 GPU, averaged across 100 examples.

B. Visualization

We show additional visualization on the equivarance of representation when the input suffers from natural corruptions. In Figure 6 and 7,we show random visualizations on Cityscapes and PASCAL VOC.

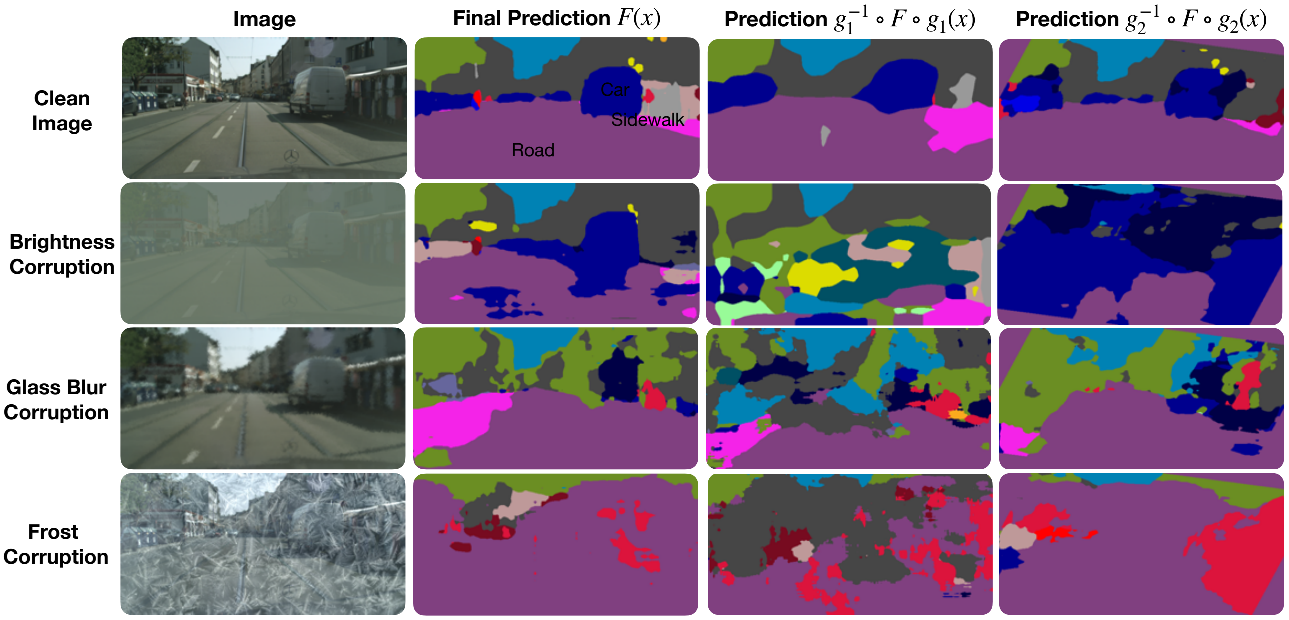

Figure 6. Examples showing equivariance on clean image and corrupted images on Cityscapes dataset. Clean image is equivariant. Images under corruption are not equivariant, allowing us to detect corruption using equivariance measurement.

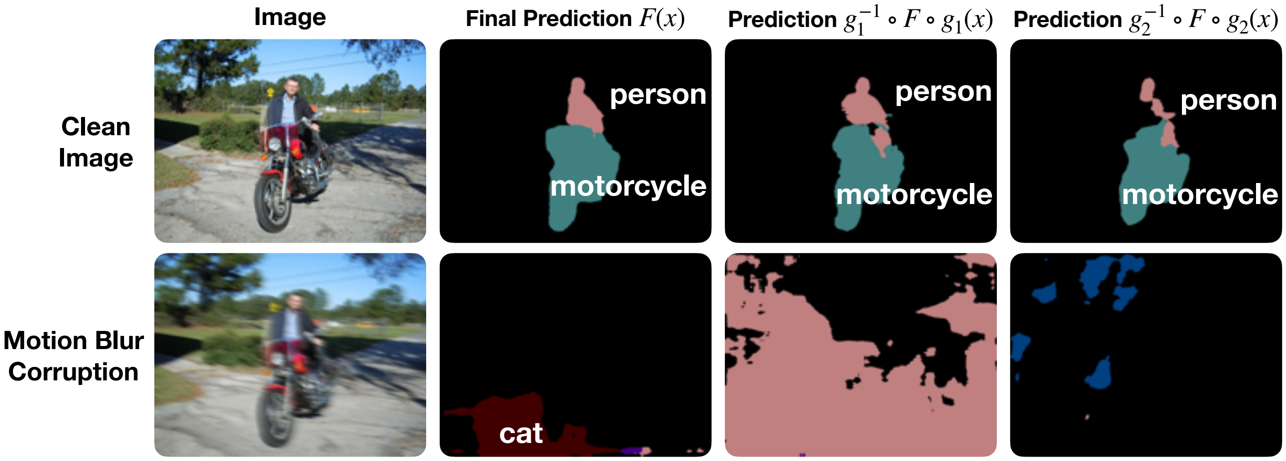

Figure 7. Examples showing equivariance on clean image and a random corrupted image on COCO dataset. Clean image is equivariant. Images under corruption are not equivariant, allowing us to detect corruption using equivariance measurement.