DICE: Domain-attack Invariant Causal Learning for Improved Data Privacy Protection and Adversarial Robustness

Qibing Ren
Shanghai Jiao Tong University
Shanghai, China
renqibing@sjtu.edu.cn

Yiting Chen
Shanghai Jiao Tong University
Shanghai, China
sjtucyt@sjtu.edu.cn

Yichuan Mo
Shanghai Jiao Tong University
Shanghai, China
mo536226@sjtu.edu.cn

Qitian Wu
Shanghai Jiao Tong University
Shanghai, China
echo740@sjtu.edu.cn

Junchi Yan*
Shanghai Jiao Tong University
Shanghai, China
yanjunchi@sjtu.edu.cn

ABSTRACT

The adversarial attack reveals the vulnerability of deep models by incurring test domain shift, while delusive attack relieves the privacy concern about personal data by injecting malicious noise into the training domain to make data unexploitable. However, beyond their successful applications, the two attacks can be easily defended by adversarial training (AT). While AT is not the panacea, it suffers from poor generalization for robustness. For the limitations of attack and defense, we argue that to fit data well, DNNs can learn the spurious relations between inputs and outputs, which are consequently utilized by the attack and defense and degrade their effectiveness, and DNNs can not easily capture the causal relations like humans to make robust decisions under attacks. In this paper, to better understand and improve attack and defense, we first take a bottom-up perspective to describe the correlations between latent factors and observed data, then analyze the effect of domain shift on DNNs induced by attack and finally develop our causal graph, namely Domain-attack Invariant Causal Model (DICM). Based on DICM, we propose a coherent causal invariant principle, which guides our algorithm design to infer the human-like causal relations. We call our algorithm Domain-attack Invariant Causal Learning (**DICE**)[1] and the experimental results on two attacks and one defense task verify its effectiveness.

CCS CONCEPTS

• **Computing methodologies** → **Machine learning**; • **Security and privacy**;

KEYWORDS

Data Privacy, Robustness, Causal Inference, Attack Transferability

ACM Reference Format:
Qibing Ren, Yiting Chen, Yichuan Mo, Qitian Wu, and Junchi Yan. 2022. DICE: Domain-attack Invariant Causal Learning for Improved Data Privacy Protection and Adversarial Robustness. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '22), August 14–18, 2022, Washington, DC, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3534678.3539242

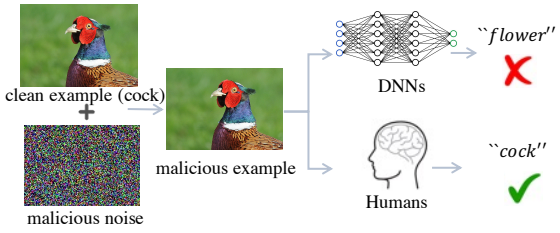

Figure 1: Vulnerability of DNNs under manual attacks.

1 INTRODUCTION

Existing DNNs often rely on the IID assumption that the training and test data follow the same distribution. When there exist domain shifts, the performance of DNNs on the new test domain would suffer dramatic degradation, which has been demonstrated by the pervasive existence of *adversarial examples* generated via injecting an imperceptible yet malicious noise to test domain [13, 43]. Meanwhile, poisoning train data with manual noise, also named *delusive attack*, recently shows its effectiveness on maximizing test error of DNNs [9–11, 18], and thus by poisoning personal data before releasing them online, we can protect data privacy against unauthorized or even illegal use because training DNNs on the poisoned data would greatly degrade their performance on clean test data. As shown in Fig. 1, although the shift away from the original domain incurred by the above two attacks brings a huge performance gap of DNNs, it does NOT affect human decisions since perturbations resulting from these attacks are imperceptible to humans [13, 43]. One popular hypothesis is that the human cognitive system is capable of capturing causal relations that are invariant to domain shifts [14], while DNNs tend to fit all types of correlations to fit

*Junchi Yan is the correspondence author, who is also with MoE Key Lab of Artificial Intelligence, Shanghai Jiao Tong University, and Shanghai AI Laboratory. This work was partly supported by National Key Research and Development Program of China (2020AAA0107600), Shanghai Municipal Science and Technology Major Project (2021SHZDZX0102), and NSFC (61972250, 72061127003).

KDD '22, August 14–18, 2022, Washington, DC, USA

ACM ISBN 978-1-4503-9385-0/22/08…$15.00
https://doi.org/10.1145/3534678.3539242

[1]Implementation available at https://github.com/Thinklab-SJTU/DICE.

data well and there can be spurious ones, i.e., not the cause of labels. Therefore, it is reasonable to assume that the attacker succeeds by exploiting such spurious factors, to shift data away from the natural distribution. However, the vulnerability of DNNs revealed by attacks can be greatly mitigated by adversarial training (AT) [25], a defense strategy that minimizes *adversarial risk* on malicious data, which shows that such spurious factors disturbed by the attacker can be easily recovered by the defender. Consequently, the attackers can improve their power if they can identify and perturb the casual relation instead of the spurious one. In terms of data privacy, the stronger delusive attack indicates better protection of personal data.

On the opposite side, AT itself suffers from poor generalization with relatively low robustness on test domain [36, 37]. Since we assume that attacks succeed by utilizing spurious factors, we argue that to defend against attacks, AT tends to fit such spurious correlation between outputs and latent factors, which is not necessarily invariant across domains. Therefore, when the spurious association changes, e.g., from training domain to test domain, the performance of robustness will change accordingly, resulting in a large generalization gap. Overall, our defense is aimed at better robustness generalization by focusing on causal factors, instead of the above spurious correlation. In this spirit, since causal reasoning can identify causal relation and remove the spurious bias by intervention [30], we initiate our study towards understanding and improving attack and defense using this powerful tool.

However, here comes two main challenges to our goals: (i) How to construct a causal graph to describe causal relationships between latent factors and observed variables in the context of attack and defense. (ii) Based on our causal graph, how to efficiently infer the unobserved causal relations from observed variables remains to be solved.

To address the first challenge, along with the perspective of latent data generating, we propose our simple yet principled bottom-up Structural Causal Model [12, 30]: namely the Domain-attack Invariant Causal Model (DICM), as illustrated in Fig. 2. In DICM, following causal assumptions of [24, 41], we split the latent factors into output-causative factors S (e.g., shape or contour of the object) and others V (e.g., the style of the object), both of which constitute the inputs X. We additionally assume that S and V are spuriously correlated, which forms a spurious and even harmful path from V to Y. Different from [24, 41], we introduce an extra domain variable D to explicitly model the effect of domain shifts induced by the attack. So far, we are capable of re-interpreting how the attack and defense work using a causal view: (i) change the mechanism from domain variable to latent factors, e.g., the attacker crafts malicious examples via $P(X|S,V)$ by perturbing $P(V|D)$ with interventions on D; (ii) manipulate the distribution of domain variable, e.g., some defense seek robust prediction through an ensemble of cross-domain models [28, 46]; or more unlabeled data [5, 27], both of which extend single domain to multiple domains and hence make models fit domain-invariant S. Finally, based on the above assumptions embedded on DICM, we formulate our *causal invariance* principle over attack-induced domain shift, which guides our algorithm design for causal inference.

For the second challenge, based on DICM, we propose a causal inference pipeline, called **D**omain-attack **I**nvariant **C**ausal **Le**arning

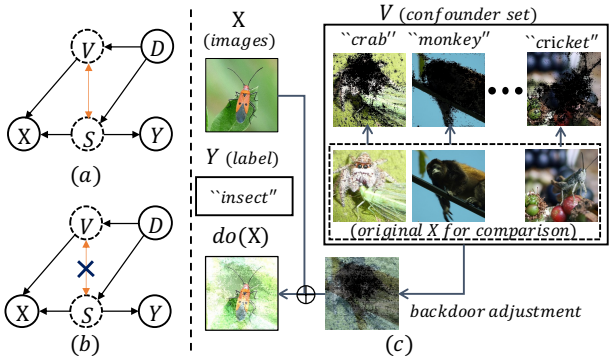

Figure 2: (a) Our domain-attack invariant causal model (DICM), (b) The intervened DICM under causal intervention, (c) Realization of causal intervention by backdoor adjustment.

(DICE), to remove the spurious bias by causal intervention. Specifically, we propose to use the backdoor adjustment method [31] for intervention, i.e., blocking the spurious path from V to Y by intervening on V as Fig. 2 (b) shows. Instead of performing the expensive "physical" intervention, DICE performs a practical "virtual" one from only the observation data. Motivated by a line of interpretability works that adversarial training produces human-perception aligned gradients [21, 47], we utilize such a prior provided by the robust model to approximate the confounder V. We also propose to improve the diversity and confounding effect of V in an adversarial way and we provide a random-sampled confounder set for an effective approximation of backdoor adjustment, as shown in Fig. 2 (c). Finally, DICE learns to infer S against V through minimizing the devised causal invariant risk.

Experimentally, we verify the advantages of DICE through three downstream tasks on real-world dataset CIFAR-10 and CIFAR-100 [22]: (i) for ***delusive attack***, we propose to facilitate this attack by attacking the learned domain-invariant factors S. Our attack outperforms the current state-of-the-art methods even under strong defense; (ii) for ***adversarial attack***, adversarial examples crafted on the learned S exhibit strong transferability across different unknown target models, even being comparable to white-box attacks; (iii) for **defense**, we verify the generality and effectiveness of DICE by integrating DICE with two popular defense baselines, PGD-AT [25] and TRADES [59], as two defense variants. Both of variants can significantly improve adversarial robustness over baselines.

Our main contributions are: (i) **Methodologically**, from a bottom-up data generating process, we build our causal graph, namely Domain-attack Invariant Causal Model (DICM), to provide a unified view of the attack and defense. With our specific and moderate assumptions embedded in DICM, we further derive *causal invariance* principle, pointing out the necessity of identifying the output-causative factors. (ii) **Algorithmically**, we propose a causal inference pipeline, namely Domain-attack Invariant Causal Learning (DICE) to infer domain-invariant features via an effective approximation of backdoor adjustment. (iii) **Experimentally**, we demonstrate our DICE outperforms baselines under two attack and one defense scenarios, with better transferability of *adversarial attack*, better data privacy protection by *delusive attack* and better robustness by defense.

2 PRELIMINARIES

Consider a classification task with data $(x, y) \in \mathcal{X} \times \mathcal{Y}$ from an observed distribution D. D_{train} denotes the training domain with D_{test} as the test domain, while $\hat{D}$ is the manually perturbed domain based on D. We first discuss the vanilla structural casual model, followed by the basic ideas on attack and defense as well as privacy and robustness, which serve the preliminaries for our main approach.

2.1 The Vanilla Structural Causal Model

2.1.1 Structural Causal Model (SCM).. The structural causal model (SCM) [12, 30] is used to describe the causal relationships. As shown in Fig. 2, each directed arrow indicates the causal relationship between two variables, e.g., $S \rightarrow Y$ denotes that S is the cause of Y. In a SCM, if a variable is the common cause of two variables, it is called the confounder, which induces spurious correlation between them, e.g., $X \leftarrow V \leftrightarrow S \rightarrow Y$, where V is the confounder of X and Y. Furthermore, the above path from X to Y is called back-door path, defined as a path that ends with an arrow pointing to X. At last, causes of variables should meet some requirement, as previous works [33, 38] express in Principle 2.1:

Principle 2.1. ([33, 38]). **Independent Causal Mechanisms (ICM) Principle**: The conditional distribution of each variable given its causes, i.e., its mechanism, does NOT inform or influence the other mechanisms.

2.1.2 Label prediction with SCM.. For causal learning for output prediction, one line of work ascribes causes of observed variables to the latent factors: those unobserved abstractions constitute inputs and their outputs [20, 24, 41]. [24, 41] further assume that the latent factors can be disentangled into output-causative factors and others while there exist domain-specific correlation between them. Our work adopts the above assumptions as our **base SCM** while under the scenario of attack and defense, we build our SCM with **two main contributions**: (i) while [24, 41] focus on Out-of-Distribution (OOD) problem and only consider the *natural sampling bias* across multiple environments/domains as the source, we focus on the single domain shift incurred by the attack, and analyze the effect of *manually injected bias* such as adversarial noise, which is necessary and more realistic since it is very likely for practitioners to collect artificial malicious examples from untrusted sources; (ii) we introduce a domain variable to model the effect of domain shift on the latent factors, resulted by the attack and defense. Though [41] also adds a domain variable, it acts as the domain index, which is unobserved and simple. However, we conceptually re-interpret attack and defense in terms of domain effect (cf. Sec. 3.1) and explicitly model domain effect in algorithm design (cf. Sec. 3.2 3.3).

2.2 Preliminaries for Attack and Defense

Without loss of generality, we use the image recognition task and its conventional learning paradigm is to train a classifier $f : \mathcal{X} \rightarrow \mathcal{Y}$, to minimize the loss $L(\cdot, \cdot)$. In this regard, standard training (ST) works by empirical risk minimization (ERM) over the clean data, which is defined as:

$$R_{nat}(f, D) = \mathbb{E}_{(x,y)\sim D}[L(f(x), y)] \tag{1}$$

In terms of the attack and defense, a widely used constraint especially in vision is that the added perturbation is within a ball, $\mathcal{B}(x, \epsilon) = \{x' : d_p(x, x') \leq \epsilon\}$, and $d_p(x, x')= \| x' - x \|_p$ is the similarity metric, and a common choice is ℓ-∞ norm.

2.3 Preserving Privacy via Train Data Poisoning

Delusive attack seeks to manipulate the *train domain* without disturbing the original label, such that DNNs trained on it perform worse on the original *test domain*, which is defined as:

$$\begin{aligned} &\max_{\hat{D}_{train} \in \mathcal{B}(D_{train}, \epsilon)} R_{nat}(f_{\hat{D}_{train}}, D_{test}) \\ &\text{s.t. } f_{\hat{D}_{train}} = \arg\min_{f} R_{nat}(f, \hat{D}_{train}) \end{aligned} \tag{2}$$

2.4 Evaluating Robustness via Test Data Attack

Adversarial attack targets at *test domain*, producing adversarial examples to fool DNNs while being imperceptible to human [13]. Its goal can also be defined in a bi-level optimization form as follows:

$$\begin{aligned} &\max_{\hat{D}_{test} \in \mathcal{B}(D_{test}, \epsilon)} R_{nat}(f_{D_{train}}, \hat{D}_{test}) \\ &\text{s.t. } f_{D_{train}} = \arg\min_{f} R_{nat}(f, D_{train}) \end{aligned} \tag{3}$$

2.5 Lifting Robustness via Adversarial Training

For resistance of adversarial (delusive) attacks, adversarial training (AT) [25] remains the most effective approach, by minimizing the adversarial risk R_{adv} as follows:

$$R_{adv}(f, D_{train}) = \mathbb{E}_{(x,y)\sim D_{train}}[\max_{x' \in \mathcal{B}(x,\epsilon)} L(f(x'), y)] \tag{4}$$

3 METHODOLOGY

We build our causal graph from the causal data-generating process to form a unified view of the attack and defense, depicted in our Domain-attack Invariant Causal Model (DICM) in Sec. 3.1. Based on our causal assumptions embedded in DICM, we further formalize our causal invariance principle against domain shifts, which guides us to infer invariant factors by causal intervention in Sec. 3.2 and Sec. 3.3. Finally, we propose our causality guided criterion on performing two attacks and one defense task in Sec. 3.4. Throughout the paper, upper-cased letters like X denote random variables, while lower-case letters like x denote deterministic value of variables.

3.1 Domain-attack Invariant Causal Model

In this section, in line with the majority of literature, we study the attack and defense for classification and present a causal view of the data-generating process behind it. Specifically, as illustrated in Fig. 2, we inspect the causalities among five variables: input data instance X, instance-level label Y, causal factors S, non-causal factors V, and the domain variable D, with a Structural Causal Model [12]. Following the discussion in Sec. 2.1.2, we next introduce the three main causal assumptions in DICM.

- **(S, V)$\rightarrow X$, $S \rightarrow Y$ (latent generating mechanism)[1]:** We introduce two latent factors S, V as the abstractions that determine the observed variables (X, Y), which have been similarly assumed

[1]Note that we consider the scenario where X and Y are generated concurrently, i.e., there is neither directed path from X to Y nor Y to X.

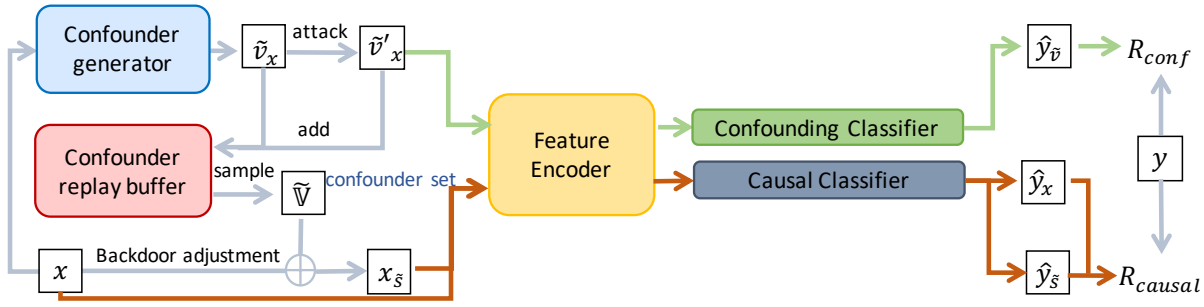

Figure 3: Pipeline of Domain-attack Invariant Causal Learning. The *green line* corresponds to the data flow through the confounding classifier, where $\tilde{v}'_x$ is the adversarial confounder being attacked. On the causal data flow via the *orange line*, both the original input x and intervened $x_{\tilde{s}}$ get inferred to the shared encoder to obtain output $\hat{y}_x$ and $\hat{y}_{\tilde{s}}$ respectively.

in existing works [24, 41]. To be specific, S as the Y-causative factors has a direct causal link to Y, which *for example* refers to the shape or contour of an object, while the non-causal factors V together with S, generate X, and in OOD setting, the generation process usually correlates with *natural bias* during sampling process, e.g., light or view, across multiple domains, while in the context of attack and defense, we assume the domain shift is incurred by *manually injected bias*, e.g., malicious noise by attacks. Different from [41], we focus on V related to *manual bias*. Recall that the key constraint on both adversarial and delusive attack is human-imperceptibility [13], which means in the manner of human perception, X (the appearance of an object) and Y (the annotated label) remain unchanged. Thus we argue that the attacks do not disturb the label-causative factor S, and $S \to Y$ is invariant to domain shifts. However, the counter-intuitive behavior of DNNs against attacked examples X reveals that the attacker may exploit high-frequency components to fool DNNs [48], which are not perceivable to humans and spuriously correlated with labels, i.e., V. Therefore, the generative mechanism $(S, V) \to X$ on DNNs is different from humans. Beyond such difference, there is a common property for both DNNs and humans based on the ICM Principle (Principle 2.1): the generative mechanism $P(X|V, S)$ is unaffected under the intervened domain shift, i.e., $P(V, S|do(D))$, which constitutes our main Assump. 3.1:

Assumption 3.1. (Causal Invariance over Attack-induced Domain Shifts) By ICM Principle [33, 38], for causal models in DICM, the $P(X|S, V)$ and $P(Y|S)$ are invariant to attack-induced domain shifts.

We give further discussion based on the above assumption.

• $S \leftrightarrow V$ **(latent spurious correlation):** We further assume that there exists a *spurious correlation* between S and V, marked as a bidirected arrow, which results from natural (manual) bias in dataset, e.g., the background (V) variation in data instances by sampling (attacks). Such a correlation $P(S, V)$ opens a backdoor path, i.e., $V \leftrightarrow S \to Y$, between V and Y, and we ascribe the vulnerability of DNNs under delusive or adversarial attacks to the learned spurious bias during fitting the dataset.

• $D \to (S, V)$**:** For the correlation $P(S, V)$, we introduce an auxiliary variable D. The magnitude of correlation can be either due to the changing mechanism $D \to (S, V)$ or the mutable distribution of the confounder D. Next we take a causal view to understand the attack and defense with help of D.

Causal view on criterion of attack and defense: So far, we can utilize our causal tool to redefine the criterion of attack and defense: given a data instance x, the label y and the target label y_t, the attacker fools the DNN-based classifier, outputting y_t instead of y, by injecting v_t, the target spurious features, into v, or in other words, disturbing $P(S, V|D)$ close to $P(S, V_t|D)$. The only difference between delusive and adversarial attack is that the former intervenes on training domain D_{train} while the latter does on test domain D_{test}. For defense, AT improves robustness via minimizing *adversarial risk* on malicious data. Since we assume that malicious data is generated from $P(S, V_t|D)$, we further state that AT improves model robustness by proactively fitting malicious data from $P(X|S, V_t)$ to defend against future possible attacks. Moreover, for the recent defense methods that require more data [5, 27] or an ensemble of cross-domain models [28, 46], we argue that they improve robustness by manipulating $P(D)$, extending it from single domain to multiple domains.

3.2 Causal Intervention by Backdoor Adjustment

Assump. 3.1 motivates us to identify S against V to fundamentally endow models with robustness or protect data against abuse. However, it is difficult to directly discover the causal factors S from the observed variables, since in our analysis DNNs also fit spurious bias V for prediction. We propose to use causal intervention: $P(Y|do(X))$ to learn the invariant mechanism $P(Y|S)$. Since "physical" intervention on X, i.e., collecting instances with all possible views or backgrounds e.g. for images, is impossible, we apply backdoor adjustment [31] to do "virtual" intervention on X in Eq. 5.

$$P(Y|do(X), D) = \sum_{v} P(Y|X, V = v)P(v|D) \tag{5}$$

Based on Assump. 3.1, $P(Y|X, V = v)$ is domain-agnostic as D is separated from X and Y with given v. Thus $P(Y|X, V)$ generalizes to test-domain $D = t$ even if trained within domain $D = s$. Moreover, by stratifying different values of confounder set $\mathbb{V} = \{v\}$ in Eq. 5 and intervening V as v, we remove the causal link $V \to X$. Since V is also not observable, to this end, we approximate confounder set $\mathbb{V} = \{v_1, v_2, \ldots, v_n\}$ using the class-wise instance-specific mask, where n is class size in dataset and $v_i \in \mathbb{R}^{h \times w}$ where $h \times w$ is the instance size. We design an additional module to generate each mask for class i to approximate the non-causal part of this class. For

$P(v|D)$, to avoid dependence of causal intervention on the training domain, $P(v|D)$ is set as the uniform $1/n$.

3.3 Domain-attack Invariant Causal Learning

Our framework consists of four components, as shown in Fig. 3 .

Confounder Generator. Motivated by the observation that loss gradients from robust models align better with salient data features and human perception, which well outlines the contour of an object in images [21, 47] (see visualization in Fig. 2), in the context of image classification, we specify to utilize magnitude of gradients as the importance of pixels and generate the mask. Specifically, given an instance x, the label y, the intervention generator first adopt a robust model with f_{prior} to obtain mask prior M_{prior}, to output the instance-specific confounder $\tilde{v}_x$:

$$\delta = \nabla_x L(f_{prior}(x), y), \quad M_{ij} = \begin{cases} 0, \max\limits_k \delta_{ijk} > d \\ 1, otherwise \end{cases}, \quad \tilde{v}_x = M_{prior} \odot x \tag{6}$$

where $\delta \in \mathbb{R}^{h \times w \times 3}$ is the loss gradient of robust models, d is the threshold that determines the number of pixels as the confounder, and $\odot$ is the element-wise multiplication. For simplicity, we select the value that excludes 50% pixels per image as d in all experiments. Moreover, we propose to adaptively update masks based on prior from robust model and knowledge learned by our model during training in a running average way, i.e., $M = \alpha * M_{ours} + (1 - \alpha) * M_{prior}$, where α increases monotonically with more epochs.

Confounder Replay Buffer. Unlike existing invariant learning works [1, 6, 23] for OOD, which construct different domains by partitioning the training set to infer the invariant features, we try to identify invariant features on a single domain by causal intervention, as shown in Eq. 5. To block all spurious paths from X to Y, we need intervene on all possible confounders V, which is impractical. In this regard, we collect $\tilde{v}_x$ of all the inputs computed previously into a replay buffer and next randomly sample a confounder set $\tilde{\mathbb{V}} = \{\tilde{v}_1, \tilde{v}_2, \ldots, \tilde{v}_n\}$ to conduct the intervention on the inputs of the modules: encoder and classifiers.

Encoder & Classifiers. To measure the prediction discrimination of S against V, we employ a shared encoder h with two classifiers, g_s and g_v, where h maps the intervened input $x_{\tilde{s}}$ to representation $\mathbf{z}_{\tilde{s}} \in \mathbb{R}^d$, and g_s projects $\mathbf{z}_{\tilde{s}}$ into a probability distribution $\hat{y}_{\tilde{s}}$ over class labels, shown in Eq. 7. Analogously, the predictive power of confounder $\tilde{v}_x$, i.e., $\hat{y}_{\tilde{v}}$, can be measured via g_v.

$$\mathbf{z}_{\tilde{s}} = h(x_{\tilde{s}}), \quad \hat{y}_{\tilde{s}} = g_s(\mathbf{z}_{\tilde{s}}) \tag{7}$$

To calculate Eq. 5, with the confounder set $\tilde{\mathbb{V}}$, it takes n times forward pass, which is expensive. To address this challenge, we adopt the Normalized Weighted Geometric Mean (NWGM) approximation [52] to move the outer sampling over v into the feature level, i.e., $\sum_v P(Y|X, V = v)P(v) \approx P(Y|X, \sum_v P(v)v)$, such that the forward cost is reduced to only once. Furthermore, since we obtain the pixel-level confounder $\tilde{v}_x$, we constitute $x_{\tilde{s}} = x + \sum_{\tilde{v} \in \tilde{\mathbb{V}}} P(\tilde{v})\tilde{v}$ by feature addition.

Optimization. Having obtained the prediction $\hat{y}_{\tilde{s}}$ under causal intervention, we can build our causal invariant risk R_s as follows:

$$R_s(f_s, D) = \mathbb{E}_{(x,y)\sim\mathcal{D}, S=x_{\tilde{s}}} [L(f_s(x_{\tilde{s}}), y)] \tag{8}$$

where $f_s(\cdot)$ denotes $g_s \circ h(\cdot)$ for simplicity, $\mathcal{D}$ is the training domain. Based on Eq. 8, we define the final risk R_{causal} by combining R_{nat} and R_s with a tunable hyper-parameters β as:

$$R_{causal}(f_s, D) = \mathbb{E}_{(x,y)\sim\mathcal{D}, S=x_{\tilde{s}}} [L(f_s(x), y) + \beta * L(f_s(x_{\tilde{s}}), y)] \tag{9}$$

For optimizing g_v, aiming at generating various and powerful confounders that may not exist in original domain, we propose to perform adversarial attack on the confounders $\tilde{v}_x$ and add them into the replay buffer. The confounding adversarial risk R_{conf} is as follows:

$$R_{conf}(f_v, D) = \mathbb{E}_{(x,y)\sim\mathcal{D}, V=\tilde{v}_x} \left[\max_{\tilde{v}'_x \in \mathcal{B}(\tilde{v}_x, \epsilon)} L(f_v(\tilde{v}'_x), y) \right] \tag{10}$$

where $f_v(\cdot)$ denotes $g_v \circ h(\cdot)$. Overall, we can jointly optimize the above two components:

$$\min_{h, g_s} R_{causal}(f_s, D) + \min_{g_v} R_{conf}(f_v, D) \tag{11}$$

The training procedure and detailed implementations are summarized in Appendix 7.1.

3.4 Causality guided Attack and Defense

In this subsection, we propose our causal invariant criterion to guide the three mainstream tasks of trustworthy AI, aiming at boosting their performance by the learned causal invariant features.

Delusive Attack: For this attack, we adopt the setting of [10], which crafts attack examples based on standard training (ST) via maximizing classification loss. Based on our causal classifier f_s trained by Eq. 11, we propose to regularize the attack generating process to focus more on causal features:

$$\max_{\hat{D}_{train} \sim \mathcal{B}(D_{train}, \epsilon)} [R_{nat}(f_{base}, \hat{D}_{train}) + \gamma * R_{nat}(f_s, \hat{D}_{train})] \tag{12}$$

where f_{base} is the standard trained classifier with f_s as ours, and γ is a tunable hyper-parameter.

Adversarial Attack: For adversarial attack, we directly attack our model via the causal classifier g_s:

$$\max_{\hat{D}_{test} \sim \mathcal{B}(D_{test}, \epsilon)} R_{nat}(f_s, \hat{D}_{test}) \tag{13}$$

Adversarial Defense: For adversarial defense, our causal invariant criterion can be seamlessly integrated with popular defense mechanisms such as PGD-AT [25], and TRADES [59]. Formally, we introduce the adversarial risk in Eq. 4 into our causal invariant risk in Eq. 9 as:

$$\min_{f_s} \max_{\hat{D}_{train} \sim \mathcal{B}(D_{train}, \epsilon)} R_{causal}(f_s, \hat{D}) \tag{14}$$

Since we focus on defending against adversarial attacks on x via minimizing R_{causal}, we propose to stop the gradient flow of R_{conf} before the feature encoder to avoid the spurious bias resulted by $\tilde{v}'_x$ interferes with robust representation learning on x.

4 RELATED WORKS

OOD Generalization with Causality Previous studies mostly seek to construct multiple domains/environments to infer causal invariant features by either partitioning the training data by prior knowledge [1, 6, 23, 24, 41], or adversarial environment inference [49,

51], while our work focuses on single domain shift incurred by the attack and defense. [34] proposes an adversarial domain augmentation to achieve domain generalization based on a single domain while it does not use causal assumptions. For causal assumptions, the old school methods [3, 32] assumes causal relations directly reside between observed inputs and outputs, which may not well suit with visual data which is the majority of benchmarks, since causality lies in conceptually latent space [24]. For latent factors, we follow the assumption of [24, 41], i.e., the mechanism that maps latent factors to observational distribution is invariant to domain shift. However, we further propose our assumptions by (i) extending the cause of spurious relation from *natural sampling bias* to *manually injected bias* such as adversarial noise; (ii) introduce a domain variable to model the effect of attack and defense on latent factors; (iii) propose our domain-attack invariance principle. Moreover, [24, 41] infer latent factors with latent generative models while we do by causal intervention [31]. Another similar work is [26] which augments images by intervening non-causal factors in self-supervised learning, while the causal factors actually depend on non-causal ones for given inputs based on its assumption.

Adversarial robustness with causality The most relevant work is ADA [61], which analyzes adversarial attack through a causal lens. However, ADA assumes the independence between causal factors and non-causal factors, which is unrealistic. Methodologically, ADA uses a "soft" intervention by penalizing the adversarial distribution and the natural distribution while we approximate "hard" intervention by backdoor adjustment. More importantly, our work provides a unified causal view of adversarial(delusive) attack and defense together while ADA only considers the adversarial data generating process. Another relevant work is CAMA [56], which directly models the latent space with a generative model, being different from our causal inference from observed data. Moreover, CAMA assumes the output is the cause of inputs, which is impractical, e.g., intervening the label with noise by distracting the sampler does not change the image.

Causal Inference It is about causal-effect reasoning or counterfactual learning by "do-calculus" intervention [30]. Recent works mostly follow the causal assumption that the observed inputs have a portion of causal relations with outputs [49, 53, 56], while [51] generalizes similar hypothesis to non-Euclidean graph data for capturing causal information among ego-graph features. However, these studies focus on guiding the representation or predictor model to leverage causality behind data, while our work designs an effective approximation of backdoor adjustment to directly infer causal factors for robust learning purpose. Similarly, CONTA [57] and CaaM [49] also utilize backdoor adjustment for causal inference, besides difference in basic causal assumptions, our work differs from them in two aspects: (i) we generate confounders based on human-perception aligned gradients from robust models while CONTA utilizes CAM and CaaM based on the attention module, which is technically differently; (ii) we design a confounder replay buffer with adversarial confounders added, bringing a more efficient approximation of backdoor adjustment while such designs are missing in CONTA and CaaM. Moreover, another related work CATT [53] proposes to remove spurious bias on attention-based vision-language models via a frontdoor adjustment, instead of our backdoor adjustment.

Adversarial Attack Since the seminal works [13, 43] that reveal the vulnerability of DNNs, studying stronger adversarial attacks has become a trending direction, among which FGSM [13], PGD attack [25], C&W attack [4], and AutoAttack [8] become the popular attack baselines to evaluate robustness of DNNs.

Delusive Attack Besides adversarial attack on test data, poisoning train data, i.e., delusive attack, recently has become a heated topic for preserving data privacy. DeepConfuse [9] first applies delusive attack to DNNs with auto-encoder. More delusive attacks have been devised by adopting new techniques and approximations, e.g., gradient alignment [11], computation graph unrolling [19] and loss minimization objective [18], etc. Our work is based on loss maximization objective [10], and further mitigates its limitation against adversarial training by our learned causal features.

Adversarial Defense The discovery of adversarial examples promote the development of defense methods, among which adversarial training (AT) [25] is considered a *principled* defense against adversarial attacks. Recent works also show the effectiveness of AT against delusive attacks [10, 18, 44]. Moreover, many works analyze the limitations of AT in two aspects: (i) the trade-off between accuracy and robustness [40, 47, 54, 59], (ii) *robust overfitting* [7, 36], for which we take a causal view to derive our insight and solution in this work. Many variants have also been proposed to improve AT by appearance-similar regularization [35], rethinking the misclassified examples [50], sample-wise importance reweighting [60], and adding more unlabeled data [5, 27]. Based on our causal graph, all of them can be encapsulated into two types: (i) manipulating the distribution of domain variable; (ii) changing the mechanism from domain variable to latent factors.

5 EXPERIMENTS

We verify the efficacy of DICE on three downstream tasks. We first show that our delusive attack is stronger than the current state-of-the-art methods for better privacy protection (c.f. Sec. 5.1). Secondly, DICE also improves transferability of adversarial attack under the challenging black-box setting, even achieving comparable performance to white-box attacks (c.f. Sec. 5.2). Finally, DICE integrated with two defense paradigms exhibits better robustness over their baselines (c.f. Sec. 5.3).

5.1 Improving Data Privacy Protection

Evaluation metrics and training details. In Sec. 3.4, we introduce our delusive attack based on TAP [10]. In line with TAP, we perform delusive attack on the base model, then train victim models on the poisoned data and compute its accuracy on clean test data to measure the attack's performance.

Attack Models. To fairly compare our method with EM and TAP, we use ResNet-18 (RN18) [15] as the backbone of all methods to generate poisoned data. In line with TAP, we perform PGD-200 [25] attack to craft poison data examples under l-∞ norm with $\epsilon = 8/255$, and we adopt the differentiable data augmentation from TAP to further improve the potency of the generated poisons.

Defense Models To fully evaluate the attack performance, similar to TAP, we test two data augmentations *not* known to the attacker during generating poisons, Gaussian Smoothing (GS) and

Table 1: Test accuracy (%) trained on poisoned data protected by defensive noises including MIXUP, DGSGD, and adversarial training with different perturbation radii ϵ. EM: error-minimizing noise. TAP: targeted adversarial poisoning noise.

Defender \ Attacker	Clean	EM	TAP	DICE (ours)
Standard Training	94.66	34.74	12.20	**6.70**
Gaussian Smoothing	95.09	40.49	31.13	**9.88**
MIXUP	95.77	46.87	19.88	**14.35**
AT (ϵ=1/255)	93.74	79.04	12.98	**8.15**
AT (ϵ=2/255)	92.37	91.84	23.25	**15.79**

(a) Evaluation results on CIFAR10.

Defender \ Attacker	Clean	EM	TAP	DICE (ours)
Standard Training	78.25	**15.99**	32.75	24.57
Gaussian Smoothing	77.05	**18.05**	30.90	24.30
MIXUP	78.99	34.43	35.28	**34.03**
AT (ϵ=1/255)	70.12	66.95	53.71	**52.11**
AT (ϵ=2/255)	67.54	65.64	67.38	**65.24**

(b) Evaluation results on CIFAR100.

Table 2: Test accuracy (%) of DICE vs. TAP with different data protection proportions on CIFAR10.

Model	Data Protection Proportion							
	20%		40%		60%		80%	
	mixed	clean	mixed	clean	mixed	clean	mixed	clean
TAP	90.47	94.81	86.52	93.90	78.80	92.12	56.37	86.60
DICE	**85.95**		**74.16**		**65.90**		**56.25**	

Mixup [58], which mixes inputs and outputs during training. Moreover, we test adversarial training, a powerful defense against recent delusive attacks [10, 18, 44], see training details in Appendix 7.2.2.

Baseline CIFAR10 & CIFAR100 Results. Table 1 shows the result of delusive attack. Our DICE generates more powerful poisoned data over the state-of-the-art EM and TAP attacks against all defenses. Even under AT, our poisoned data still remains its effectiveness. We report more results under AT with larger perturbation radii ϵ in Appendix 7.2.3, which our attack consistently shows its advantage. It is worth noting that AT might not be an ideal solution for delusive attack, since it is computationally expensive and degrades clean accuracy especially for large scale dataset, e.g., in Table 1, CIFAR100 under AT (ϵ=2/255) drops from 78.25% to 67.54% without being attacked.

Less Data. We then study a more challenging and realistic scenario, where only part of data is protected, i.e., being poisoned. Specifically, we randomly select a certain proportion of data from CIFAR10, poison them, and train models on the mixed data and the remaining clean data. Table 2 shows that (i) models trained on mixed data perform worse than only clean data in all cases, verifying the effectiveness of poisons; (ii) as data protection proportion increases, i.e., more poisoned data is added, both of attacks become stronger while DICE is consistently stronger than TAP. Since EM [18] are observed to be slightly helpful when combined with clean data, we do not report results of EM. All poisoned data are crafted by PGD-50.

Transferability. We further test the transferability of our delusive attack. Specifically, we select RN18 as the surrogate model on which poisoned data gets crafted by PGD-200, then train various target models on the poisoned data. For a thorough evaluation, we select VGG19 [39], GoogleNet [42], DenseNet-121 (DN121) [17] and MobileNet-V2 (MOB-V2) [16] as target models. Results of Table 3 show that our DICE is stronger than TAP across all targets models.

Table 3: Test accuracy (%) on CIFAR10 w/ different backbones as attack target in DICE and TAP-based transfer attacks.

SURROGATE \ TARGET	VGG19	GoogleNet	DN121	MOB-V2
TAP	11.78	10.59	9.89	7.27
DICE (ours)	**9.17**	**6.42**	**8.72**	**7.19**

Table 4: Test accuracy (%) on CIFAR10 of different target models in transfer-based FGSM and PGD-100 black box attacks, which are generated based on WRN (baseline) and DICE. We also report the results of white-box attacks directly on target models.

SURROGATE \ TARGET	VGG16	RN18	RN50	DN121
WRN	67.18	61.66	64.35	74.54
DICE (ours)	**43.91**	**34.97**	39.73	51.42
White-box	52.69	36.84	**33.68**	**38.67**

(a) Evaluation results under FGSM attack.

SURROGATE \ TARGET	VGG16	RN18	RN50	DN121
WRN	69.23	48.94	56.62	80.49
DICE (ours)	9.95	2.8	5.68	24.27
White-box	**0.01**	**2.40**	**1.78**	**0.22**

(b) Evaluation results under PGD-100 attacks.

Ablation Study. We perform a sensitivity analysis about γ of Eq. 12 in Appendix 7.2.4. We also compare the effects of different attack steps during crafting poisoned data on model performance. See Appendix 7.2.5 for details. Finally, we present time complexity analysis in Appendix 7.2.6.

5.2 Improving Transferability of Adversarial Attack

Evaluation metrics and training details. To evaluate the transferability of attacks, we generate adversarial examples based on *surrogate model*, then transfer those examples to *target models* to compute the test accuracy as a measurement. For a thorough evaluation, we perform FGSM [13] and PGD-100 attacks on CIFAR10 respectively, which get crafted under the l_∞ norm with ϵ = 8/255.

Baseline Models. To cover as many different deep architectures as possible, we select VGG16 [39], ResNet-18 (RN18), ResNet-50 (RN50) [15], and DenseNet-121 (DN121) [17] as our target models. We reproduce these models based on a popular Github repository[2]. For a fair comparison, we select WRN-34-10 (WRN) [55], the backbone of our model, as the surrogate model baseline.

Transfer attack results. Table 4 shows that our model consistently achieves stronger transferability over the target models than WRN by a huge margin, which is comparable to white-box attacks that have the full knowledge of target models.

[2] https://github.com/kuangliu/pytorch-cifar

Table 5: Test accuracy (%) of DICE-M with different regularization hyperparameter β values on CIFAR10 Under WRN.

β	Clean	PGD-20	C&W-20
0	82.88	46.59	49.19
1	82.81	47.04	50.97
2	81.98	47.73	50.36
3	82.26	48.51	51.24
4	**83.18**	**49.28**	51.96
5	82.67	48.54	**52.52**

Table 6: Test Accuracy (%) of our DICE-M and DICE-T vs. Baselines on CIFAR100 Under WRN.

Defender \ Attacker	Clean	FGSM	PGD-100	C&W-100	AA.
PGD-AT	66.26	35.31	22.95	24.39	22.01
TRADES	58.64	37.12	26.63	24.48	23.2
DICE-M	**67.15**	37.4	24.35	**26.99**	23.61
DICE-T	59.84	**39.88**	**28.42**	26.37	**25.73**

5.3 Improving Robustness over Adversarial Training

Evaluation metrics and training details. For robustness evaluation, we craft adversarial examples under l-∞ norm with $\epsilon = 8/255$ by FGSM, PGD, C&W [4] attacks, and AutoAttack (AA.) [8]. We choose WRN-34-10 (WRN) as the backbone for both CIFAR-10 and CIFAR-100. We adopt the hyper-parameters recipe for training suggested by [29] and robustness is evaluated on the last checkpoint of all models. More training details is given in Appendix 7.3.

Baseline Models. We propose our causal guided adversarial defense objective in Eq. 14. We apply our DICE to two popular defense paradigms, PGD-AT [25] and TRADES [59] respectively, to build our robust model, named DICE-M and DICE-T.

Sensitivity of regularization hyperparameter β on CIFAR10 As Eq. 9 shows, β is an important hyper-parameter. In Table 5, we select WRN as the backbone and compare PGD-AT baseline to DICE-M with different β values, and find that as $\beta \in [0, 4]$, larger β brings better clean accuracy vs. robustness trade-off than baseline (β=0). However, as λ further increases, e.g., $\beta = 5$, our model starts to overfit the adversarial noise induced by the C&W attack since its robustness begins to decrease under PGD attack. Therefore, we set β as 4 for both CIFAR-10 and CIFAR-100.

Performance on CIFAR-100 Besides CIFAR-10, we further apply our method on CIFAR-100 and Table 6 shows that our two variants DICE-M and DICE-T consistently exhibit better robustness than their baseline PGD-AT and TRADES respectively under various attacks including AutoAttack. Moreover, DICE helps robust baselines with a better trade-off between accuracy and robustness: all variants enjoy better test accuracy than their baselines.

Consideration of gradient obfuscation To exclude the possible effect of obfuscated gradients that give a false sense of security, we perform a series of sanity checks suggested by [2]: (i) on CIFAR-10, our model shows stronger robustness under FGSM attack (64.18%) than multi-step attack: PGD-20 (49.28%) and C&W-20 attack (50.95%); (ii) our models show stronger robustness under black-box PGD-20 attacks (78.50%) than white-box PGD-20 attacks (49.28%); (iii) PGD attacks with more steps become stronger, i.e., our models exhibit stronger robustness against PGD-20 (49.28%) than PGD-100 (48.57%). Therefore, according to the criterion given by [2], we believe that the robustness of our model does not result from obfuscated gradients.

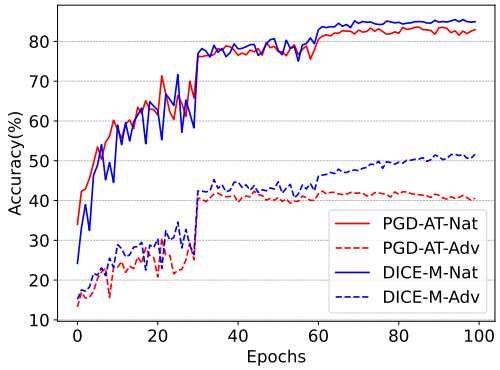

Figure 4: Natural (Robust) test accuracy of PGD-AT (red line) and DICE-M (blue line) trained using WRN-34-10 on CIFAR-10 under PGD-20 attack during training. PGD-AT-Nat and PGD-AT-Adv mean the accuracy of PGD-AT on natural and adversarial data respectively, which also applies to our model. To better verify the effectiveness of our model on mitigating *robust overfitting*, the learning rate gets decayed at 30 and 60 epochs during the whole 100 epochs.

Consideration of adaptive attack Based on the adaptive attack criterion [45], we generate attacks by maximizing our causal invariant risk R_{causal}, i.e., Eq. 9 and DICE-M on CIFAR-10 achieves 49.04% robust accuracy under the adaptive PGD-20 attack compared to 49.28% under vanilla PGD-20 attack, showing our model consistently achieves strong robustness even under a specifically designed attack.

Mitigating robust overfitting *Robust overfitting* is a common limitation existing in current defense methods [7, 36], which means that model robustness on test data begins decreasing after the first learning rate decay while robustness on train data keeps increasing. The test accuracy plot in Fig. 4 shows that our model indeed mitigates this limitation with both better clean accuracy and robustness, which is aligned with our analysis.

6 CONCLUSIONS

In this paper, we take a bottom-up view to model the latent data generating process and understand the limitations of the attack and defense. In specific, we propose our causal graph and show that the spurious correlation between latent factors and outputs is exploited by attacks while the defense also learns to fit such spurious relations to defend against attacks, and such reliance may result in poor generalization defense. Therefore, we argue that inferring causal relations like humans is important for improving both attack and defense. Inspired by our causal graph, we propose a causal inference pipeline to learn domain-invariant features via an effective approximation of causal intervention. Experimental results verify the utility of our method, bringing better privacy protection on *delusive attack*, improved transferability on *adversarial attack*, and higher robustness on defense.

REFERENCES

[1] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. 2019. Invariant risk minimization. arXiv preprint arXiv:1907.02893 (2019).

[2] Anish Athalye, Nicholas Carlini, and David Wagner. 2018. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In ICML.

[3] Peter Bühlmann. 2020. Invariance, causality and robustness. Statist. Sci. (2020).

[4] Nicholas Carlini and David Wagner. 2017. Towards evaluating the robustness of neural networks. In IEEE Symposium on Security and Privacy.

[5] Yair Carmon, Aditi Raghunathan, Ludwig Schmidt, John C Duchi, and Percy S Liang. 2019. Unlabeled data improves adversarial robustness. In NeurIPS, Vol. 32.

[6] Shiyu Chang, Yang Zhang, Mo Yu, and Tommi Jaakkola. 2020. Invariant rationalization. In ICML.

[7] Tianlong Chen, Zhenyu Zhang, Sijia Liu, Shiyu Chang, and Zhangyang Wang. 2020. Robust overfitting may be mitigated by properly learned smoothening. In ICLR.

[8] Francesco Croce and Matthias Hein. 2020. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In ICML.

[9] Ji Feng, Qi-Zhi Cai, and Zhi-Hua Zhou. 2019. Learning to confuse: generating training time adversarial data with auto-encoder. In NeurIPS.

[10] Liam Fowl, Micah Goldblum, Ping-yeh Chiang, Jonas Geiping, Wojciech Czaja, and Tom Goldstein. 2021. Adversarial examples make strong poisons. In NeurIPS.

[11] Jonas Geiping, Liam Fowl, W Ronny Huang, Wojciech Czaja, Gavin Taylor, Michael Moeller, and Tom Goldstein. 2021. Witches' brew: Industrial scale data poisoning via gradient matching. In ICLR.

[12] Madelyn Glymour, Judea Pearl, and Nicholas P Jewell. 2016. Causal inference in statistics: A primer. John Wiley & Sons.

[13] I. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and Harnessing Adversarial Examples. arXiv preprint arxiv:1412.6572 (2015).

[14] Alison Gopnik, Clark Glymour, David M Sobel, Laura E Schulz, Tamar Kushnir, and David Danks. 2004. A theory of causal learning in children: causal maps and Bayes nets. Psychological review 111, 1 (2004), 3.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In CVPR.

[16] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861 (2017).

[17] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. 2017. Densely connected convolutional networks. In CVPR.

[18] Hanxun Huang, Xingjun Ma, Sarah Monazam Erfani, James Bailey, and Yisen Wang. 2021. Unlearnable examples: Making personal data unexploitable. In ICLR.

[19] W Ronny Huang, Jonas Geiping, Liam Fowl, Gavin Taylor, and Tom Goldstein. 2020. Metapoison: Practical general-purpose clean-label data poisoning. In NeurIPS.

[20] Ilyes Khemakhem, Diederik Kingma, Ricardo Monti, and Aapo Hyvarinen. 2020. Variational autoencoders and nonlinear ica: A unifying framework. In AISTATS.

[21] Beomsu Kim, Junghoon Seo, and Taegyun Jeon. 2019. Bridging adversarial robustness and gradient interpretability. arXiv preprint arXiv:1903.11626 (2019).

[22] Alex Krizhevsky, Geoffrey Hinton, et al. 2009. Learning multiple layers of features from tiny images. (2009).

[23] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. 2021. Out-of-distribution generalization via risk extrapolation (rex). In ICML.

[24] Chang Liu, Xinwei Sun, Jindong Wang, Haoyue Tang, Tao Li, Tao Qin, Wei Chen, and Tie-Yan Liu. 2021. Learning causal semantic representation for out-of-distribution prediction. In NeurIPS.

[25] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2017. Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083 (2017).

[26] Jovana Mitrovic, Brian McWilliams, Jacob Walker, Lars Buesing, and Charles Blundell. 2020. Representation learning via invariant causal mechanisms. arXiv preprint arXiv:2010.07922 (2020).

[27] Amir Najafi, Shin-ichi Maeda, Masanori Koyama, and Takeru Miyato. 2019. Robustness to adversarial perturbations in learning from incomplete data. In NeurIPS, Vol. 32.

[28] Tianyu Pang, Kun Xu, Chao Du, Ning Chen, and Jun Zhu. 2019. Improving adversarial robustness via promoting ensemble diversity. In ICML.

[29] Tianyu Pang, Xiao Yang, Yinpeng Dong, Hang Su, and Jun Zhu. 2020. Bag of tricks for adversarial training. arXiv preprint arXiv:2010.00467 (2020).

[30] Judea Pearl. 2009. Causality. Cambridge university press.

[31] Judea Pearl. 2014. Interpretation and identification of causal mediation. Psychological methods 19, 4 (2014), 459.

[32] Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. 2016. Causal inference by using invariant prediction: identification and confidence intervals. Journal of the Royal Statistical Society: Series B (Statistical Methodology) (2016).

[33] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. 2017. Elements of causal inference: foundations and learning algorithms. The MIT Press.

[34] Fengchun Qiao, Long Zhao, and Xi Peng. 2020. Learning to learn single domain generalization. In CVPR.

[35] Qibing Ren, Qingquan Bao, Runzhong Wang, and Junchi Yan. 2022. Appearance and Structure Aware Robust Deep Visual Graph Matching: Attack, Defense and Beyond. In CVPR.

[36] Leslie Rice, Eric Wong, and Zico Kolter. 2020. Overfitting in adversarially robust deep learning. In ICML.

[37] Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. 2018. Adversarially robust generalization requires more data. In NeurIPS.

[38] Bernhard Schölkopf, Dominik Janzing, Jonas Peters, Eleni Sgouritsa, Kun Zhang, and Joris Mooij. 2012. On causal and anticausal learning. In ICML.

[39] Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In ICLR.

[40] Dong Su, Huan Zhang, Hongge Chen, Jinfeng Yi, Pin-Yu Chen, and Yupeng Gao. 2018. Is Robustness the Cost of Accuracy?–A Comprehensive Study on the Robustness of 18 Deep Image Classification Models. In ECCV.

[41] Xinwei Sun, Botong Wu, Xiangyu Zheng, Chang Liu, Wei Chen, Tao Qin, and Tie-yan Liu. 2021. Recovering Latent Causal Factor for Generalization to Distributional Shifts. In NeurIPS.

[42] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In CVPR.

[43] Christian Szegedy, W. Zaremba, Ilya Sutskever, Joan Bruna, D. Erhan, I. Goodfellow, and R. Fergus. 2014. Intriguing properties of neural networks. arXiv preprint arxiv:1312.6199 (2014).

[44] Lue Tao, Lei Feng, Jinfeng Yi, Sheng-Jun Huang, and Songcan Chen. 2021. Better safe than sorry: Preventing delusive adversaries with adversarial training. In NeurIPS.

[45] Florian Tramer, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. 2020. On adaptive attacks to adversarial example defenses. In NeurIPS.

[46] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. 2017. Ensemble adversarial training: Attacks and defenses. arXiv preprint arXiv:1705.07204 (2017).

[47] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. 2018. Robustness may be at odds with accuracy. arXiv preprint arXiv:1805.12152 (2018).

[48] Haohan Wang, Xindi Wu, Zeyi Huang, and Eric P Xing. 2020. High-frequency component helps explain the generalization of convolutional neural networks. In CVPR.

[49] Tan Wang, Chang Zhou, Qianru Sun, and Hanwang Zhang. 2021. Causal attention for unbiased visual recognition. In ICCV.

[50] Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. 2019. Improving adversarial robustness requires revisiting misclassified examples. In ICLR.

[51] Qitian Wu, Hengrui Zhang, Junchi Yan, and David Wipf. 2022. Handling Distribution Shifts on Graphs: An Invariance Perspective. In ICLR.

[52] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In ICML.

[53] Xu Yang, Hanwang Zhang, Guojun Qi, and Jianfei Cai. 2021. Causal attention for vision-language tasks. In CVPR.

[54] Yao-Yuan Yang, Cyrus Rashtchian, Hongyang Zhang, Russ R Salakhutdinov, and Kamalika Chaudhuri. 2020. A closer look at accuracy vs. robustness. In NeurIPS.

[55] Sergey Zagoruyko and Nikos Komodakis. 2016. Wide residual networks. arXiv preprint arXiv:1605.07146 (2016).

[56] Cheng Zhang, Kun Zhang, and Yingzhen Li. 2020. A causal view on robustness of neural networks. In NeurIPS.

[57] Dong Zhang, Hanwang Zhang, Jinhui Tang, Xian-Sheng Hua, and Qianru Sun. 2020. Causal intervention for weakly-supervised semantic segmentation. In NeurIPS.

[58] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. 2018. mixup: Beyond empirical risk minimization. In ICLR.

[59] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. 2019. Theoretically principled trade-off between robustness and accuracy. In ICML.

[60] Jingfeng Zhang, Jianing Zhu, Gang Niu, Bo Han, Masashi Sugiyama, and Mohan Kankanhalli. 2021. Geometry-aware instance-reweighted adversarial training. In ICLR.

[61] Yonggang Zhang, Mingming Gong, Tongliang Liu, Gang Niu, Xinmei Tian, Bo Han, Bernhard Schölkopf, and Kun Zhang. 2021. Adversarial robustness through the lens of causality. arXiv preprint arXiv:2106.06196 (2021).

7 APPENDIXES

7.1 Domain-attack Invariant Causal Learning

7.1.1 training details. Models are trained on the SGM optimizer with momentum 0.9, weight decay 0.0005, and batch size 128. For both CIFAR10 and CIFAR100, we set the learning rate as 0.1, which gets decays at 100 and 105 epochs with total 110 epochs. β in Eq. 9 is kept as 1.0 for both delusive attack and adversarial attack, and 4.0 for adversarial defense.

For generating confounders, we find that directly relying on the current model without prior model achieves comparable or even better performance than ensembling the two models together. Therefore, we only utilize our current model to generate confounders.

For confounder replay buffer, since our confounder gets updated during training, the newly added confounders are relatively more important than old ones. So when sampling confounder set, we assign each confounder with a timestamp, which gets decayed since being first added, thus those newly added confounders are more likely to be sampled during training. We set the maximum size of reply buffer as 10000 and the confounder set size is set as 20.

7.2 Causality guided Delusive Attack

7.2.1 Hyper-parameters and training details for delusive attack. We follow TAP [10] to perform delusive attacks. For the standard trained classifier, it is trained for 40 epochs with a batch size of 128, a momentum factor of 0.9, a weight decay factor of 0.0005, an initial learning rate of 0.1, and a learning rate scheduler that decays learning rate by 0.1 after 15, 25 and 35 epochs.

The regularization hyper-parameter γ is set to 0.5 for every experiment. For CIFAR10, attack iteration is set to 200. For CIFAR100, attack iteration is set to 50.

7.2.2 Hyper-parameters and training details for evaluatioin. We follow TAP [10] to craft poisoned data on a fixed pretrained model. For pretraining, the classifier is trained for 200 epochs with a batch size of 128, a momentum factor of 0.9, a weight decay factor of 0.0005, an initial learning rate of 0.1, and a CosineAnnealing learning rate scheduler.

For the adversarial training paradigm used in defending delusive attacks, we follow the official setting of PGD-AT [25] and build the PGD-7 attack adversary with a relative step size 0.25. Classifiers are trained for 100 epochs with a batch size of 128, a momentum factor of 0.9, a weight decay factor of 0.0005, an initial learning rate of 0.1, and a learning rate scheduler that decays learning rate by 0.1 after 40 and 80 epochs.

7.2.3 More results under stronger AT. To fully compare the effect of AT against delusive attack, we perform stronger AT with larger perturbation budget ϵ=3/255 and 4/255 on DICE and TAP. Table 7 shows that when ϵ=3/255, DICE is stronger then TAP while with ϵ=4/255, the protection of both DICE and TAP becomes worthless. However, as we argue in our paper, a large AT perturbation would induce lower natural accuracy, hindering its applicability, thus it may not be the ideal solution for delusive attacks.

7.2.4 Sensitivity analysis of the regularization hyper-parameter γ. Eq. 12 tells that γ is an important regularization hyper-parameter for our delusive attack. For this experiment, we employ PGD-50 attack to perform delusive attack. In Table 8, we compare the effect of different γ values and find that larger *gamma* leads to more effective delusive attack.

Table 7: Test accuracy (%) of DICE vs. TAP under AT with different perturbation radii ϵ.

Defender \ Attacker	Clean	TAP	DICE(ours)
AT($\epsilon = 3/255$)	89.23	85.34	**78.56**
AT($\epsilon = 4/255$)	88.26	87.61	**87.50**

(a) Evaluation results on CIFAR10.

Defender \ Attacker	Clean	TAP	DICE(ours)
AT($\epsilon = 3/255$)	66.47	64.04	**62.52**
AT($\epsilon = 4/255$)	64.33	62.20	**61.27**

(b) Evaluation results on CIFAR100.

Table 8: Test accuracy (%) of DICE with different regularization hyper-parameter γ on CIFAR10. FGSM denotes adversarial training under FGSM attack.

Defender \ γ	0.1	0.5	1.0
ST	10.67	9.87	**7.58**
FGSM	21.00	19.35	**19.07**

Table 9: Test accuracy (%) of DICE with different PGD attack steps on CIFAR10.

Defender \ Attack steps	50	100	200
ST	13.21	10.21	**6.70**
AT(ϵ=1/255)	34.55	17.66	**8.15**
AT(ϵ=2/255)	79.19	43.36	**15.79**

7.2.5 Sensitivity analysis of different attack steps. Another important hyper-parameter of delusive attacks is the attack steps. Table 9 shows that more attack steps induce stronger poison attacks. For a better trade-off between computation cost and performance, in our paper, we finally choose poison data by PGD-200 step attack.

7.2.6 Time complexity analysis. In line with TAP, our method enjoys the same advantage of the ease of crafting perturbations: we craft the poisoned data on a fixed pretrained model, without extra training cost. In contrast, DeepConfuse [9] needs training an adversarial auto-encoder ($5\tilde{7}$ GPU days on simple datasets) and EM [18] needs iteratively updating model and poisons until the evaluation metric meets their threshold. Moreover, EM also needs grasp the whole training dataset at once, which is less realistic.

7.3 Causality guided Adversarial Training

7.3.1 Training details. We craft PGD attacks under the l-∞ norm with ϵ=8/255 in all defense experiments. In training, we use the PGD-10 attack adversary with a step size of 0.25. Models are trained on the SGM optimizer with momentum 0.9, weight decay 0.0005, and batch size 128. For CIFAR10/100, we set the learning rate as 0.1, which gets decays at 100 and 105 epochs with total 110 epochs.