Reconstruct and Match: Out-of-Distribution Robustness via Topological Homogeneity

Chaoqi Chen[1] **Luyao Tang**[2] **Hui Huang**[1*]
[1]College of Computer Science and Software Engineering, Shenzhen University
[2]School of Informatics, Xiamen University
cqchen1994@gmail.com, lytang@stu.xmu.edu.cn, hhzhiyan@gmail.com

Abstract

Since deep learning models are usually deployed in non-stationary environments, it is imperative to improve their robustness to out-of-distribution (OOD) data. A common approach to mitigate distribution shift is to regularize internal representations or predictors learned from in-distribution (ID) data to be domain invariant. Past studies have primarily learned pairwise invariances, ignoring the intrinsic structure and high-order dependencies of the data. Unlike machines, humans recognize objects by first dividing them into major components and then identifying the topological relation of these components. Motivated by this, we propose Reconstruct and Match (REMA), a general learning framework for object recognition tasks to endow deep models with the capability of capturing the topological homogeneity of objects without human prior knowledge or fine-grained annotations. To identify major components from objects, REMA introduces a selective slot-based reconstruction module to dynamically map dense pixels into a sparse and discrete set of slot vectors in an unsupervised manner. Then, to model high-order dependencies among these components, we propose a hypergraph-based relational reasoning module that models the intricate relations of nodes (slots) with structural constraints. Experiments on standard benchmarks show that REMA outperforms state-of-the-art methods in OOD generalization and test-time adaptation settings.

1 Introduction

Distribution shift has emerged as a major challenge for the success of machine learning systems [34, 25]. Although deep learning models are believed to generalize well to in-distribution (ID) data, a well-trained model deployed in the open world often encounters out-of-distribution (OOD) data whose contexts may differ from the training distribution, resulting in dramatic performance degeneration and raising concerns about model safety in many high-staking applications, such as autonomous driving and medical diagnosis. This gives rise to the importance of OOD generalization [92, 76], which aims to build a robust learning machine that can perform well in unseen test environments.

Regarding OOD generalization, a central theme is how to learn general features from training data that can be extrapolated to test distributions. Following this idea, a plethora of OOD generalization methods have been proposed over the past few years, including domain alignment [42], latent feature disentanglement [61, 48, 85], meta-learning [40, 41], invariant risk minimization [2, 1, 96], and augmentation-guided invariant predictor [74, 83, 95], to name a few. On the other hand, given the natural adaptivity gap between training and test distributions [15], recent works [29, 30, 81, 11, 8] attempt to further enhance the source-trained model through test-time adaptation [46], which leverages unlabeled target samples to update the model in an online manner. These two series of studies address distribution shifts during training and inference and have been proven to work synergistically [8].

*Corresponding author

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

 https://doi.org/10.52202/079017-3990

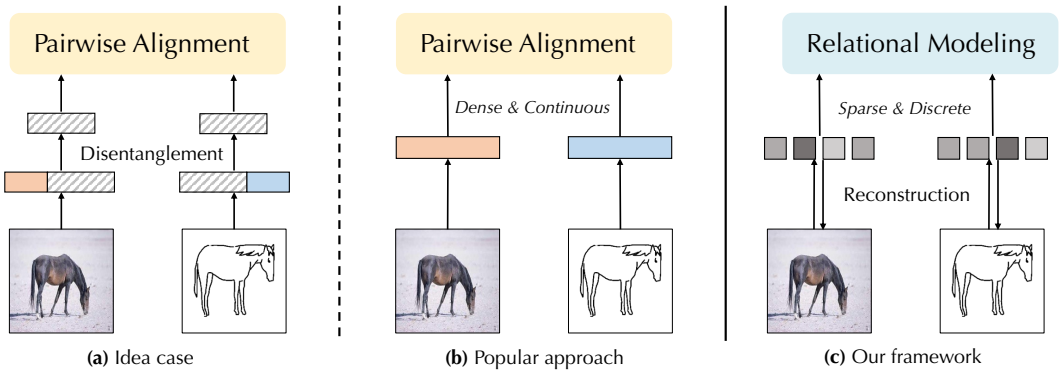


Figure 1: **Motivation of the proposed REMA. (a)** Ideally, by partitioning the latent factors into common and specific parts, aligning the common factors directly enables perfect matching of two different distributions. **(b)** Since the latent features learned by deep networks are typically dense and continuous, most well-performing methods seek direct alignment without exploring the inherent structures. **(c)** Our REMA introduces a sparse and discrete element – slot [51] – to serve as a bridge for both reconstruction and relational modeling, avoiding explicit disentanglement in the latent space.

Albeit their general efficacy for various tasks, these prior efforts largely overlook the topological structure of the image data. As a consequence, existing OOD generalization methods highly rely on specialized regularization objectives and are significantly less interpretable than human vision systems. Unlike machines, human object recognition [4] typically involves initially decomposing objects into several major components (*e.g.,* keypoints), followed by identifying the structural relationships among these components, and finally making predictions by comprehensively considering the main components and their inherent relations. To this end, a critical question remains open in the field:

How to devise a unified framework that imitates the human vision process for OOD generalization?

In this work, we believe that OOD generalization requires consideration of two key aspects: 1) how to represent the structure of data, and 2) how to model the relationships between different entities.

Grounded on these insights, we propose Reconstruct and Match (REMA), a general framework to endow deep models with the capability of capturing the *topological homogeneity* of objects without using human prior knowledge or fine-grained attribute annotations. Figure 1 illustrates the motivation. To identify major components from objects, REMA introduces a slot-based reconstruction module to map dense pixels into a sparse set of slot vectors in an unsupervised manner. This module encourages the deep model to reduce unwanted redundancy and preserve those predominant parts (the words "component" and "part" are used interchangeably). Then, to model and reason the high-order dependencies among these components, we propose a hypergraph-based matching module that discovers the relations of slots with structural constraints, *i.e.,* topological homogeneity of the object. The main contributions of this paper are summarized as follows:

- We propose REMA, a novel OOD generalization framework to mitigate distribution shifts in deep learning models by imitating the human visual recognition process.
- We introduce a self-supervised reconstruction process to identify informative parts from objects without additional supervision, and a hypergraph matching module to reason the high-order part-based object relationships and interactions across domains.
- Experiments on six widely used benchmarks demonstrate that REMA outperforms state-of-the-art methods in OOD generalization and test-time adaptation settings.

2 Preliminaries

Problem Setup. Assume that $\mathcal{X}$ is the input space, $\mathcal{Z}$ is the latent space, and $\mathcal{Y}$ is the output space. The predictor $f = h \circ g$ is comprised of a featurizer $g : \mathcal{X} \mapsto \mathcal{Z}$ that learns to extract embedding features, and a classifier $h : \mathcal{Z} \mapsto \mathcal{Y}$ that makes predictions based on the extracted features. The goal

https://doi.org/10.52202/079017-3990

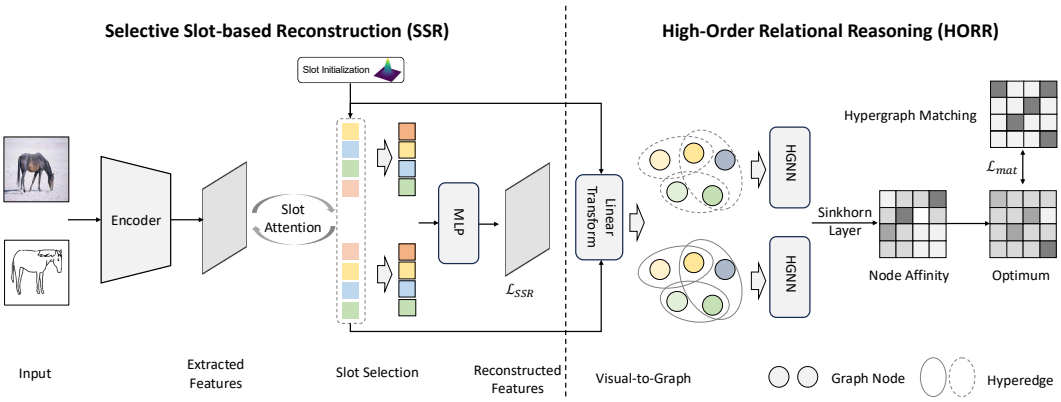


Figure 2: Overview of the proposed REMA, which consists of two key modules, *i.e.,* SSR and HORR. (1) **Abstraction:** Slot-based reconstruction to discover the main components from the data by binding objects with a set of discrete vectors; (2) **Reasoning:** Introduce high-order relational inductive bias (*i.e.,* topological homogeneity) to the network via the process of hypergraph construction, learning, and matching. HGNN means hypergraph neural networks.

of OOD generalization is to find a predictor $f : \mathcal{X} \mapsto \mathcal{Y}$ that generalizes well to all unseen target domains. In deep neural networks, empirical risk minimization (ERM) [72] is capable of learning highly predictive features, making it the simplest baseline for this problem.

ERM Objective. We train the model by minimizing the empirical cross-entropy loss function:

$$\arg\min_{\theta\in\Theta} \mathbb{E}_{(x,y)\sim P^{tr}}[\ell(f_\theta(x), y)]. \tag{1}$$

where P^{tr} is the training distribution, Θ denotes the parameter space, and ℓ is the loss function. The trained model will be evaluated on a test set from a test distribution P^{ts}, where $P^{tr} \neq P^{ts}$.

3 Proposed Method

In this section, we provide a detailed description of our Reconstruct and Match (REMA) framework. As shown in Figure 2, REMA consists of two key modules tackling the main component discovery (Section 3.1) and high-order relational modeling and reasoning (Section 3.2).

3.1 On Discovering Main Components

For OOD generalization/adaptation, a long-standing issue is how to distinguish between transferable (domain-agnostic) and non-transferable (domain-related) information in the data. An ideal solution is to disentangle latent representations in an unsupervised manner (see Fig. 1(a)), which has been proven to be prohibitively difficult and even infeasible [50]. Therefore, most prior works (see Fig. 1(b)) aim to learn domain-invariant representations through elaborate feature alignment modules, such as adversarial training [21] and pseudo-labeling [37], without exploring the intrinsic composition of the features themselves. Despite their general efficacy, these methods may introduce a significant amount of redundant and noisy information during the alignment process. For humans, when identifying an object, we tend to first look for its main components, which can be regarded as a process of information compression from a learning perspective. Inspired by this, we aim to mimic the very first step of the human recognition process by identifying the main components from object images without human prior knowledge or fine-grained annotations.

To be specific, we introduce a self-supervised reconstruction module based on Slot Attention [51], termed *Selective Slot-based Reconstruction (SSR)*, to assist deep models in learning a set of part-based representations for sparsely characterizing a target object. Formally, given an input x, g extracts a set of feature embeddings $\mathbf{z} \in \mathbb{R}^{N\times d_z}$, where N is the number of embedding and d_z denotes embedding dimension. The Slot Attention module will first take a set of slots $\mathbf{s} \in \mathbb{R}^{K\times d_s}$ (K is the number of slots) and the feature map $\mathbf{z}$, then project them to dimension d_s by a linear transformation f_k for slots

https://doi.org/10.52202/079017-3990

and f_q, f_v for $\mathbf{z}$. The Slot Attention will be trained as follows,

$$\texttt{update}(\boldsymbol{A}, \mathbf{v}) = \boldsymbol{A}^T \mathbf{v}, A_{ij} = \frac{\texttt{attn}(\mathbf{q}, \mathbf{k})_{ij}}{\sum_{l=1}^{K} \texttt{attn}(\mathbf{q}, \mathbf{k})_{lj}}, \texttt{attn}(\mathbf{q}, \mathbf{k}) = \frac{e^{M_{ij}}}{\sum_{l=1}^{N} e^{M_{il}}}, \boldsymbol{M} = \frac{\mathbf{k}\mathbf{q}^T}{\sqrt{d_s}}, \quad (2)$$

where $\mathbf{q} = f_q(\mathbf{z}) \in \mathbb{R}^{K \times d_s}$, $\mathbf{k} = f_k(\mathbf{z}) \in \mathbb{R}^{N \times d_s}$, and $\mathbf{v} = f_v(\mathbf{z}) \in \mathbb{R}^{N \times d_s}$ denote the query, key and value vectors respectively. $\boldsymbol{A} \in \mathbb{R}^{N \times K}$ stands for the attention matrix. Unlike self-attention, the queries in slot attention are a function of the slots $\mathbf{s} \sim \mathcal{N}(\mathbf{s}; \boldsymbol{\mu}, \boldsymbol{\sigma}) \in \mathbb{R}^{K \times d_s}$, and will be updated iteratively over T iterations. The slots are initialized by random sampling from a standard Gaussian distribution. Queries at iteration t are represented by $\hat{\mathbf{q}}^t = f_q(\mathbf{s}^t)$, and the slot updating process is: $\mathbf{s}^{t+1} := \texttt{update}(\texttt{attn}(\hat{\mathbf{q}}^t, \mathbf{k}), \mathbf{v})$. After each iteration, a Gated Recurrent Unit (GRU) is employed on the slot representations $\mathbf{s}^{t+1}$ to update their states, integrating new information while retaining relevant context from previous iterations.

Since the number of slots K is manually predefined and fixed during training, attention networks may learn redundant or even incorrect associations, thus affecting the understanding of objects and scenes. Therefore, our SSR introduces a selective slot attention mechanism that dynamically adjusts the importance of slots for reconstruction. To quantitatively represent which slots are more important for reconstruction, we first introduce an importance score ρ for each slot. For each slot $\mathbf{s}_i$, we use a lightweight neural network h_θ to predict an initial importance score: $\rho_i^{\text{init}} = \sigma(h_\theta(\mathbf{s}_i))$, where $\sigma(\cdot)$ is the sigmoid function, ensuring $\rho_i^{\text{init}} \in [0, 1]$. To capture interactions among slots, we further introduce an interaction matrix $\boldsymbol{W} \in \mathbb{R}^{K \times K}$, where each element $\boldsymbol{W}_{ij}$ is defined as $\boldsymbol{W}_{ij} = \text{softmax}_j\left((\boldsymbol{U}\mathbf{s}_i)^\top (\boldsymbol{V}\mathbf{s}_j)/\sqrt{d}\right)$. Here, $\boldsymbol{U}$ and $\boldsymbol{V}$ are learnable projection matrices, and d is a scaling factor. The final importance score ρ_i for each slot $\mathbf{s}_i$ is then computed as $\rho_i = \sum_{j=1}^{K} \boldsymbol{W}_{ij} \rho_j^{\text{init}}$. We then scale each slot representation by its respective importance score to create a weighted slot representation $\tilde{\mathbf{s}}_i = \rho_i \cdot \mathbf{s}_i$. To this end, the overall training objective of SSR can be formulated as:

$$\mathcal{L}_{\text{SSR}} = \|\hat{x} - x\|_2^2 + \lambda \sum_{i=1}^{K} \left\|\rho^{\text{init}}\right\|_2, \quad (3)$$

where λ is the balancing parameter. The regularization term encourages sparsity in the initial importance scores, guiding the model to utilize only the most relevant slots for reconstruction. On the other hand, the less relevant slots are down-weighted rather than discarded, which may help retain subtle information. Compared to continuous semantic representations, this training process is more manageable due to the discretization of the slots. Since we lack truly fine-grained annotations, the learned slots or "concepts" may not be as readily interpretable as individual words. It is worth noting that incorporating a strong visual feature extractor could significantly boost baseline performance. However, for the sake of fairness, we have opted not to use approaches, such as DINOSAUR [64].

3.2 High-Order Relational Reasoning (HORR)

Albeit we have obtained part-based representations in the form of slots, their inner- and inter-object relations are still under-explored. Prior graph-based [9, 7] and non-graph [33] works only accommodate pairwise relationships (*e.g.,* connection between two nodes in a simple graph), but are inadequate to model the high-order relations inherent in images. Addressing this issue, we introduce a simple yet effective technique—HORR—to model high-order topological relations with a collection of graph nodes and hyperedges, divided into three stages: construction, learning, and matching.

Definition 1 (Topological Homogeneity) *We conceptualize topological homogeneity between the same object across images as a (hyper)graph matching problem, which is mathematically formulated as a relaxed quadratic assignment problem [52],*

$$\begin{aligned} \min_{\mathbf{X}} \|\mathbf{A} - \mathbf{X}\mathbf{B}\mathbf{X}^T\|_F^2 - tr(\mathbf{X}_u^T \mathbf{X}), \\ \mathbf{X} \in [0, 1]^{n \times m}, \mathbf{X}\mathbf{1}_n \leq \mathbf{1}_m, \mathbf{X}^T \mathbf{1}_m \leq \mathbf{1}_n, \end{aligned} \quad (4)$$

where $\mathbf{A} \in \mathbb{R}^{n \times n}$ and $\mathbf{B} \in \mathbb{R}^{m \times m}$ are the adjacent matrix encoding structure information of the graph $\mathcal{G}_A$ and $\mathcal{G}_B$ respectively, n and m are the number of graph nodes, $||\cdot||_F$ is the Frobenius norm, $\mathbf{X}_u \in \mathbb{R}^{m \times n}$ is the unary affinity matrix and generally specified as the node affinity $\mathbf{M}_{\text{aff}}$, and $\mathbf{X}$ is the relaxed permutation matrix encoding node-to-node assignment.

Hypergraph Construction. Given visual slot representations from the previous step, we carry out a linear transformation to obtain graph nodes $\mathcal{V}_A$ and $\mathcal{V}_B$. This transformation is intended to map the visual features into the graph domain, ensuring effective graph matching. Then, we build hypergraph graphs $\mathcal{G}_A$ and $\mathcal{G}_B$ in both domains, modeling the topological structures inherent in the images, x_A and x_B, respectively. The hypergraph [17] is defined as: $\mathcal{G}_{A/B} = (\mathcal{V}_{A/B}, \mathcal{E}_{A/B}, \mathbf{W})$, where $\mathcal{V}_{A/B}$ is the node sets, $\mathcal{E}_{A/B}$ is the hyperedge sets, and $\mathbf{W} \in \mathbb{R}^{|\mathcal{E}|\times|\mathcal{E}|}$ is a diagonal matrix of edge weights. The hypergraph is denoted by an incidence matrix $\mathbf{H} \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{E}|}$, where $\mathbf{H}(v, e) = 1$ indicates the node $v \in e$ and $\mathbf{H}(v, e) = 0$ indicates $v \notin e$. We adopt non-parametric density estimation, *i.e.,* K-Nearest Neighbors (KNN), to establish hyperedge connections. The Euclidean distance is used to calculate the distance between node embeddings.

Hypergraph Learning. After constructing the hypergraphs, we update the graph node features according to the connectivity defined by the hyperedges. Technically, we introduce hypergraph convolution [17] to perform message-passing and feature aggregation,

$$\tilde{\mathcal{V}} = \mathbf{D}_v^{-1/2}\mathbf{H}\mathbf{W}\mathbf{D}_e^{-1}\mathbf{H}^T\mathbf{D}_v^{-1/2}\mathbf{\Theta}, \tag{5}$$

where $\mathbf{\Theta}$ represents the parameter to be learned during training. By doing so, we can explicitly model intricate correlations among different parts of an image (*i.e.,* slots). On the other hand, in contrast to grids, sequences, and even graphs, this high-order modeling can more effectively capture relations among nodes, avoiding excessive or erroneous pairwise connections.

Hypergraph Matching. Since the graph node features have been updated by Eq. (5), we introduce an affinity matrix $\mathbf{W}_{\text{aff}}$ to measure the node correspondence between $\mathcal{G}_A$ and $\mathcal{G}_B$. Following [19, 44], we use the differentiable Sinkhorn layer [67] to calculate the affinity matrix $\hat{\mathbf{W}}_{\text{aff}}$, where each element indicates the degree of matching between pairs of nodes across graphs. To leverage the topological structures, we need a training objective to minimize the pairwise structural discrepancy between the hypergraphs [78]. In particular, we follow the Definition 1 to specify unary affinity matrix $\mathbf{X}_u$ as the obtained node affinity matrix $\hat{\mathbf{W}}_{\text{aff}}$. Since the constructed hypergraphs naturally encode rich high-order relationships, they enhance the cross-domain topological matching process. Conversely, the matching process introduces additional structural knowledge to the current graph through message propagation. Formally, the hypergraph-based matching objective is formulated as follows:

$$\mathcal{L}_{\text{mat}} = \underbrace{\sum_i \frac{1}{n}\left[\max_j(\hat{\mathbf{W}}_{\text{aff}} \odot \mathbf{Y}_{\mathbf{\Pi}})_{i,j} - \mathbf{1}\right]^2}_{\text{enhance true positive matches}} + \underbrace{\sum_{i,j} \frac{1}{\|\mathbf{1} - \mathbf{Y}_{\mathbf{\Pi}})\|_1}\left[\hat{\mathbf{W}}_{\text{aff}} \odot (\mathbf{1} - \mathbf{Y}_{\mathbf{\Pi}})\right]_{i,j}^2}_{\text{penalize false positive matches}}, \tag{6}$$

where the (i, j) element in $\mathbf{Y}_{\mathbf{\Pi}} \in \mathbb{R}^{n\times m}$ is $\mathbf{1}$ if $v_i^A \in \mathcal{G}_A$ and $v_j^B \in G_B$ belong to the same object class, otherwise it will be $\mathbf{0}$. Here, $\mathbf{Y}_{\mathbf{\Pi}}$ can be regarded as pseudo labels for matching due to the absence of ground-truth correspondence. Moreover, as the matching primarily targets high-level semantic relationships, we avoid imposing additional structural constraints [23, 43, 44] on this training objective, thereby significantly simplifying the training process.

3.3 Training and Inference

Putting everything together, the full training objective is formulated as follows:

$$\mathcal{L}_{\text{REMA}} = \mathcal{L}_{\text{CE}} + \alpha\mathcal{L}_{\text{SSR}} + \beta\mathcal{L}_{\text{mat}}, \tag{7}$$

where α and β are hyper-parameters for balancing different loss terms. Our algorithm first trains deep models using the reconstruction objective to obtain discrete part representations with sufficient sparsity. Then it turns to model the cross-image components correlations as the graph matching problem where we further introduce the structural regularization term into the matching loss.

Note that our approach can be directly applied to test-time adaptation, similar to those self-supervised methods [69, 49, 3, 20]. The difference lies in the fact that their proxy tasks are somewhat heuristic, such as predicting rotations, jigsaw puzzles, and random masking. In contrast, our method directly exploits the intrinsic structured properties of the data, making it more robust and versatile.

https://doi.org/10.52202/079017-3990

Table 1: Comparison with OOD generalization methods on the PACS, Office-Home, and VLCS. *Note that the results reported in this table do not involve any test-time adaptation strategies.* Results are averaged over 3 random seeds. $\pm x$ denotes the rounded standard error.

Algorithm	PACS	Office-Home	VLCS	Average Acc. (%)
ERM [72]	85.5	67.6	77.5	76.7
CORAL [68]	86.2	68.7	78.8	77.9
DANN [21]	83.7	65.9	78.6	76.1
MLDG [40]	84.9	66.8	77.2	76.3
CDANN [45]	82.6	65.7	77.5	75.3
MMD [42]	84.7	66.4	77.5	76.2
IRM [2]	83.5	64.3	78.6	75.5
GroupDRO [62]	84.4	66.0	76.7	75.7
I-Mixup [80, 82, 84]	84.6	68.1	77.4	76.7
RSC [28]	85.2	65.5	77.1	75.9
ARM [88]	85.1	64.8	77.6	75.8
MTL [5]	84.6	66.4	77.2	76.1
VREx [36]	84.9	66.4	78.3	76.5
Mixstyle [95]	85.2	60.4	77.9	74.5
SelfReg [32]	85.6	67.9	77.8	77.1
SagNet [55]	86.3	68.1	77.8	77.4
GVRT [53]	85.1	70.1	79.0	78.1
VNE [33]	86.9	65.9	78.1	77.0
REMA (Ours)	$88.7_{\pm 0.3}$	$72.0_{\pm 0.4}$	$79.4_{\pm 0.3}$	**80.0**

4 Experiments

In this section, we empirically evaluate the proposed REMA in two types of OOD scenarios, *i.e.,* OOD generalization and test-time adaptation. In the following, we first describe the experimental setup (Section 4.1) and then provide the main results (Section 4.2) and ablation studies (Section 4.3).

4.1 Experimental Setup

Datasets. For OOD generalization, we leverage the three most widely used benchmark datasets. **PACS** [39] comprises 9,991 images and exhibits significant variations in image styles. It consists of 4 domains each with 7 classes, *i.e.,* `Photo, Art Painting, Cartoon, Sketch`. **VLCS** contains 10,729 images of 5 classes from 4 photographic domains: `PASCAL VOC 2007, LabelMe, Caltech, Sun`. **Office-Home** [73] is collected from both office and home environments, and its domain shifts stem from variations in viewpoint and image style. It has 15,500 images of 65 classes from 4 domains, *i.e.,* `Artistic, Clipart, Product, Real World`. Regarding test-time adaptation, we follow the common benchmarks [75, 30, 79] that utilize **CIFAR-10/100** [35] and **ImageNet** [14] as the ID (training) data. **CIFAR-10/100C** [26] and **Imagenet-C** [26] are used as OOD (test) data, comprising different corruptions applied to their original datasets.

Implementation Details. For OOD generalization, we use ResNet-50 for PACS, Office-Home, and VLCS and ResNet-18 for CIFAR-10. The model is trained using stochastic gradient descent with momentum 0.9, and weight decay 10^{-4}. The training batch size is set to 128. The learning rate is 10^{-4}. Following common practice, the model selection is based on a training domain validation set. For test-time adaptation, we use ResNet-50 for all datasets. We utilize the Adam optimizer to update the network parameters. To facilitate a fair comparison, the test batch size of 64 in all methods. λ in Eq. (3) is set to 0.01 in all experiments. α and β in Eq. (7) is set to 10 and 0.1, respectively.

Baselines. We compare REMA against two types of baseline methods. **(1) OOD Generalization:** We adopt the leave-one-domain-out evaluation protocol and follow the model selection strategy used in [25]. The main baselines have optimization-based CORAL [68], MLDG [40], CDANN [45] and MMD [42], augmentation-based Mixstyle [95], SagNet [55] and I-Mixup [80, 82, 84], and so on. **(2) Test-time adaptation:** All methods are based on online batch-level test data adaptation setting and the following baselines are included: entropy-minimization: Tent [75] and SHOT [47], pseudo-labeling: T3A [29] and TAST [31], consistency-alignment: TSD [79] and TIPI [56].

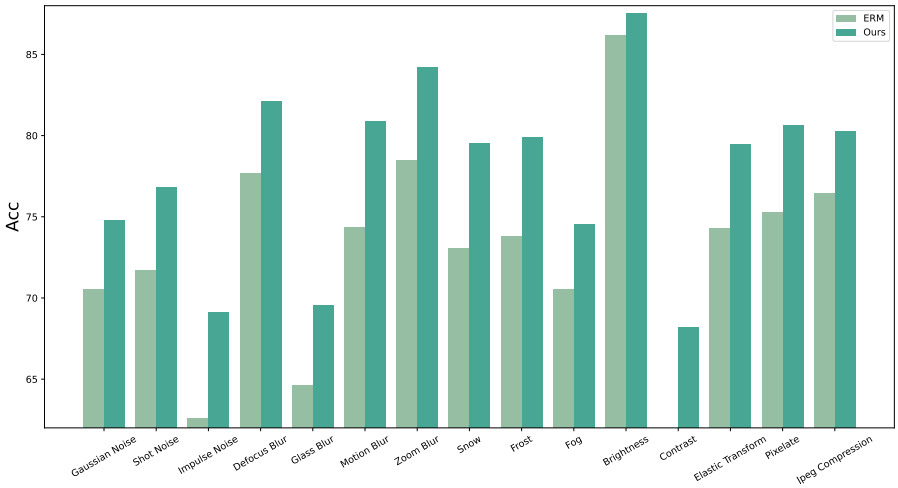


Figure 3: Generalizing from CIFAR-10 to CIFAR-10C using ERM and our REMA.

Table 2: Comparisons with the state-of-the-art methods with average error rate (%) on image corruption benchmarks. Testing is conducted on the highest level of image corruption. All methods use ResNet-50 backbone. ↓ means lower is better.

Method	**CIFAR-10C** ↓	**CIFAR-100C** ↓	**ImageNet-C** ↓	**Avg.** ↓
No Adaptation	29.1	60.4	82.0	57.2
+SHOT [47]	15.3	41.5	58.3	38.4
+Tent [75]	14.0	39.0	58.1	37.0
+PL [37]	22.3	40.1	63.0	41.8
+T3A [29]	26.7	58.3	75.8	53.6
+TAST [31]	26.6	60.7	-	-
+TAST-BN [31]	13.1	37.8	67.1	39.3
+TIPI [56]	13.5	38.3	55.9	35.9
+TSD [79]	13.1	37.7	53.2	34.6
+REMA	12.0	35.8	52.2	**33.3**

4.2 Main Results

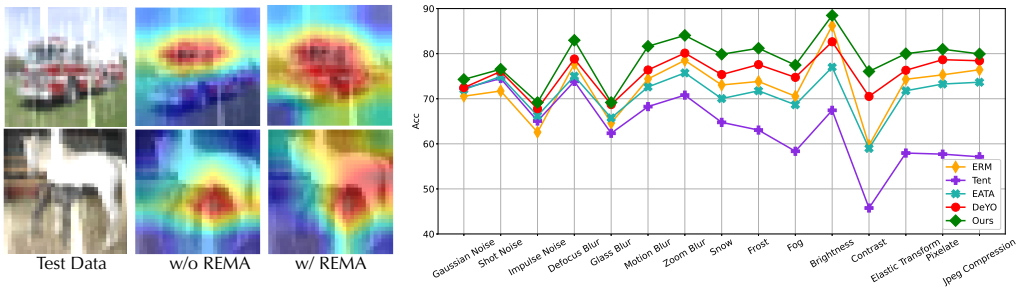
OOD Generalization. The main results are presented in Table 1 and Figure 3. Notably, REMA consistently outperforms all baseline methods by a large margin in each dataset. For example, compared to the recent method VNE, REMA improves classification accuracy by 1.8% for PACS, 6.1% for Office-Home, and 1.3% for VLCS, revealing the importance of our reconstruction (semantic abstraction) and matching (topological homogeneity) modules. Moreover, there are two notable observations: (1) Compared to statistical matching methods (*e.g.,* CORAL, DANN, and MMD) that directly optimize moment matching objectives in the latent space, REMA demonstrates superior performance by introducing topological information with more sparse input (slots). (2) Style or data augmentation based methods (*e.g.,* MixStyle) is simple and easy-to-implement. However, their intuitive nature of interpolating around the training set may not accurately cover the target distribution region. By contrast, REMA directly learns major components from training data. (3) Compared to PACS and VLCS datasets, the Office-Home dataset has more categories and total samples. Many baseline approaches even perform worse than ERM, indicating their limited scalability. Instead, REMA demonstrates a larger performance gain in this challenging task.

Test-Time Adaptation. We compared REMA with existing advanced TTA methods, and the results are summarized in Table 2 and Table 3. REMA consistently provides improvements on multiple datasets. Compared to TSD [79], it achieves a 1.3% improvement on three pixel-level corruption datasets and a 3.1% improvement on three challenging cross-domain datasets. Compared to the SOTA, REMA has three major advantages: (1) Tent [75] or SHOT [47] based on entropy minimization has limitations, as they can overcome distribution shifts caused by pixel corruption but fail to handle cross-domain test data, even leading to performance degradation. However, REMA can handle both types of OOD data. (2) T3A [29] and TAST [31] based on pseudo-labels have performance limitations, and the pros and cons of pseudo-labels significantly impact the adaptation, which can be

https://doi.org/10.52202/079017-3990

Table 3: Comparisons with the state-of-the-art methods on three image classification benchmarks.

Method	VLCS	PACS	Office-Home	Average Acc. (%)
ERM	$76.7_{\pm 0.5}$	$83.2_{\pm 1.1}$	$67.1_{\pm 1.0}$	75.3
+Tent [75]	$73.0_{\pm 1.3}$	$85.2_{\pm 0.6}$	$66.3_{\pm 0.8}$	74.9
+TentClf [75]	$75.8_{\pm 0.7}$	$82.7_{\pm 1.6}$	$66.8_{\pm 1.0}$	75.1
+SHOT [47]	$67.1_{\pm 0.9}$	$84.1_{\pm 1.2}$	$67.6_{\pm 0.7}$	72.9
+T3A [29]	$77.3_{\pm 0.4}$	$83.9_{\pm 1.1}$	$68.3_{\pm 0.8}$	76.5
+TAST [31]	$77.7_{\pm 0.5}$	$84.1_{\pm 1.2}$	$68.6_{\pm 0.7}$	76.8
+TAST-BN [31]	$73.5_{\pm 1.4}$	$89.2_{\pm 0.5}$	$68.9_{\pm 0.5}$	77.2
+TSD [79]	$74.5_{\pm 0.9}$	$89.3_{\pm 0.6}$	$68.4_{\pm 0.7}$	77.3
+REMA	$79.4_{\pm 0.4}$	$90.3_{\pm 0.3}$	$71.6_{\pm 0.6}$	**80.4**



(a) Grad-CAM Visualization (b) Adaptation Order (→)

Figure 4: **(a)** Visualization. **(b)** Analysis on continuous test-time adaptation.

observed in their ceilings on two types of datasets. REMA enhances resistance to noisy labels by capturing higher-order semantic dependencies. (3) REMA converts dense pixels into sparse slots, accurately capturing domain-invariant features, which will not be affected by pixel corruption and domain shift. Implementing consistent optimization during test time makes REMA more robust, resulting in a 5.1% accuracy improvement across all benchmarks, *i.e.,* ERM *vs.* REMA.

4.3 Ablation Studies

Table 4: Ablation of REMA (%).

ASR	HORR	VLCS	PACS	Office-Home
×	×	76.7	83.2	67.1
✓	×	78.6	88.1	70.3
×	✓	78.3	89.0	70.2
✓	✓	79.4	90.3	71.6

Ablations of key modules in REMA. In this part, we provide the ablation results in Table 4, investigating the independent and combined effects of SSR and HORR proposed in REMA. As can be seen, incorporating SSR and HORR separately leads to improved generalization performance, demonstrating their contributions to abstraction and relational modeling. In addition, integrating SSR and HORR in our method yields the best performance, highlighting that the two modules systematically work together and reciprocate each other. Moreover, as shown in Figure 4(a), the full REMA can provide more complete and accurate representations of objects.

Analysis on continuous test-time adaptation. We empirically evaluate the continuous learning ability of REMA by comparing it with state-of-the-art test-time adaptation methods, namely Tent [75], EATA [57], and DeYO [38]. The results are presented in Fig. 4(b), where the adaptation order is from left to right. We can see that the proposed REMA substantially and consistently outperforms all baseline methods as adaptation proceeds, revealing the robustness and the capability to handle open-world changes.

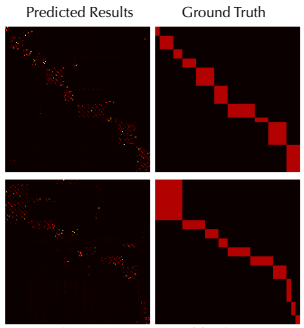
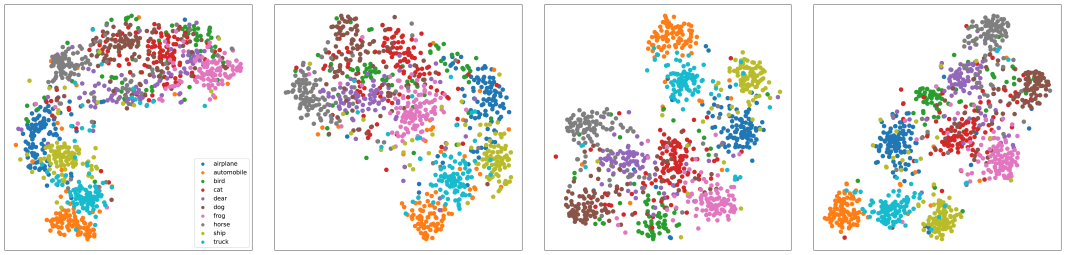


Figure 5: Learned affinity *vs.* GT

Analysis on hypergraph matching. In Figure 5, we visualize the learned doubly stochastic node affinity matrix and the ground-truth (GT) matrix. We observe that the proposed graph matching method effectively identifies the correct node affinity, revealing its effectiveness in handling cross-domain higher-order relationships.



(a) ERM **(b)** REMA w/o SSR **(c)** REMA w/o HORR **(d)** REMA (Full)

Figure 6: Feature visualization of different methods using t-SNE.

Feature visualization. Figure 6 demonstrates the t-SNE [71] visualization of feature embeddings for ERM, REMA w/o SSR, REMA w/o HORR, and REMA (Full). We extract feature embeddings using CIFAR-10C data, where the corruption type is snow with a severity level of 5. The different color stands for different classes. We can observe that our REMA exhibits better clustering patterns with both SSR and HORR modules. When SSR is removed, some closely related categories (*e.g.,* animals) become highly mixed, indicating the advantage of SSR in extracting robust semantic representations. On the other hand, without HORR, intra-class variations increase, indicating some degree of ambiguity between different categories. Our HORR addresses semantic ambiguity by modeling topological homogeneity.

5 Related Work

OOD Generalization. The aim of OOD Generalization is to train a model using data from the source distribution so that it can perform well on an unknown target distribution. A series of works can be divided into three categories: (1) Minimizing distribution differences [42, 45, 68, 90]: using meta-learning to simulate distribution shift [40, 18], or learning an invariant transformation based on adversarial learning [42, 45, 89, 94, 93]. (2) Domain-invariant representation learning: by analyzing the feature learning process of deep neural networks [16, 66, 27, 63, 12, 13], it is inferred how ID data can be generalized to OOD data [6, 65]. By understanding the interactions between ERM and generalization, domain-invariant features are learned. (3) More generic OOD generalization: many studies focus on generalization under a relaxed assumption, such as open environments with unknown classes [8], semi-supervised settings in the testing environment [86], adapting foundation models [91], and OOD data synthesis [60, 59, 22], aiming to identify generalizable features from sophisticated test distributions. However, previous studies have only focused on the instances themselves and have not explored the inherent high-order semantic dependencies in ID and OOD data, overlooking the topological homogeneity between distributions.

Test-time Adaptation. The goal of TTA [46] is to adjust the source trained model using test data without ground truth during the testing phase [47, 58, 10, 87, 79, 29]. Most of the existing methods directly perform gradient optimization on the test samples, with optimization objectives including prediction entropy [75], self-training and stochastic restoring [77], or based on normalization [54], etc., making the model adapt to the dynamic target environment at each step. Recently works further consider the scenario of noisy outlier samples [24, 70] appearing in the test stream. However, previous studies have simply optimized the test instances based on self-training, ignoring the topological relationship between samples and the high-order semantic dependencies within the test data steam, leading to suboptimal adaptation.

6 Conclusion

In this paper, we propose to achieve out-of-distribution robustness from the perspective of ensuring topological homogeneity between the same object class regardless of the surrounding environments. To achieve this goal, we propose a new framework REMA to first obtain a set of sparse and discrete representations from dense pixels and then model the high-order topological relations and dependencies via hypergraph (a generalized form of graph). Our experimental results reveal that REMA achieves superior performance on standard OOD generalization and test-time adaptation benchmarks. In the future, we aim to extend this framework to more complex and dynamically changing scenes.

https://doi.org/10.52202/079017-3990

Acknowledgement

We thank the reviewers for their valuable comments. This work was supported in parts by NSFC (U21B2023, U2001206), ICFCRT(W2441020), Guangdong Basic and Applied Basic Research Foundation (2023B1515120026), DEGP Innovation Team (2022KCXTD025), Shenzhen Science and Technology Program (KQTD20210811090044003, RCJC20200714114435012), and Scientific Development Funds from Shenzhen University.

References

[1] Kartik Ahuja, Karthikeyan Shanmugam, Kush Varshney, and Amit Dhurandhar. Invariant risk minimization games. In *ICML*, pages 145–155, 2020.

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

[3] Alexander Bartler, Andre Bühler, Felix Wiewel, Mario Döbler, and Bin Yang. Mt3: Meta test-time training for self-supervised test-time adaption. In *International Conference on Artificial Intelligence and Statistics*, pages 3080–3090. PMLR, 2022.

[4] Irving Biederman. Recognition-by-components: a theory of human image understanding. *Psychological review*, 94(2):115, 1987.

[5] Gilles Blanchard, Aniket Anand Deshmukh, Ürun Dogan, Gyemin Lee, and Clayton Scott. Domain generalization by marginal transfer learning. *The Journal of Machine Learning Research*, 22(1):46–100, 2021.

[6] Yuan Cao, Zixiang Chen, Misha Belkin, and Quanquan Gu. Benign overfitting in two-layer convolutional neural networks. *Advances in neural information processing systems*, 35:25237–25250, 2022.

[7] Chaoqi Chen, Jiongcheng Li, Hong-Yu Zhou, Xiaoguang Han, Yue Huang, Xinghao Ding, and Yizhou Yu. Relation matters: Foreground-aware graph-based relational reasoning for domain adaptive object detection. *TPAMI*, 2022.

[8] Chaoqi Chen, Luyao Tang, Yue Huang, Xiaoguang Han, and Yizhou Yu. Coda: Generalizing to open and unseen domains with compaction and disambiguation. *NeurIPS*, 36, 2023.

[9] Chaoqi Chen, Luyao Tang, Feng Liu, Gangming Zhao, Yue Huang, and Yizhou Yu. Mix and reason: Reasoning over semantic topology with data mixing for domain generalization. *NeurIPS*, 35:33302–33315, 2022.

[10] Dian Chen, Dequan Wang, Trevor Darrell, and Sayna Ebrahimi. Contrastive test-time adaptation. In *CVPR*, pages 295–305, 2022.

[11] Liang Chen, Yong Zhang, Yibing Song, Ying Shan, and Lingqiao Liu. Improved test-time adaptation for domain generalization. In *CVPR*, 2023.

[12] Liang Chen, Yong Zhang, Yibing Song, Anton Van Den Hengel, and Lingqiao Liu. Domain generalization via rationale invariance. In *ICCV*, pages 1751–1760, 2023.

[13] Liang Chen, Yong Zhang, Yibing Song, Zhen Zhang, and Lingqiao Liu. A causal inspired early-branching structure for domain generalization. *IJCV*, pages 1–21, 2024.

[14] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. Ieee, 2009.

[15] Abhimanyu Dubey, Vignesh Ramanathan, Alex Pentland, and Dhruv Mahajan. Adaptive methods for real-world domain generalization. In *CVPR*, 2021.

[16] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. Toy models of superposition. *arXiv preprint arXiv:2209.10652*, 2022.

[17] Yifan Feng, Haoxuan You, Zizhao Zhang, Rongrong Ji, and Yue Gao. Hypergraph neural networks. In *AAAI*, volume 33, pages 3558–3565, 2019.

[18] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, pages 1126–1135, 2017.

[19] Kexue Fu, Shaolei Liu, Xiaoyuan Luo, and Manning Wang. Robust point cloud registration framework based on deep graph matching. In *CVPR*, pages 8893–8902, 2021.

[20] Yossi Gandelsman, Yu Sun, Xinlei Chen, and Alexei Efros. Test-time training with masked autoencoders. *NeurIPS*, 35:29374–29385, 2022.

[21] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *JMLR*, 17(1):2096–2030, 2016.

[22] Jiahui Gao, Renjie Pi, Yong Lin, Hang Xu, Jiacheng Ye, Zhiyong Wu, Weizhong Zhang, Xiaodan Liang, Zhenguo Li, and Lingpeng Kong. Self-guided noise-free data generation for efficient zero-shot learning, 2023.

[23] Quankai Gao, Fudong Wang, Nan Xue, Jin-Gang Yu, and Gui-Song Xia. Deep graph matching under quadratic constraint. In *CVPR*, pages 5069–5078, 2021.

[24] Taesik Gong, Yewon Kim, Taeckyung Lee, Sorn Chottananurak, and Sung-Ju Lee. Sotta: Robust test-time adaptation on noisy data streams. *Advances in Neural Information Processing Systems*, 36, 2024.

[25] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *ICLR*, 2021.

[26] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019.

[27] Katherine Hermann and Andrew Lampinen. What shapes feature representations? exploring datasets, architectures, and training. *Advances in Neural Information Processing Systems*, 33:9995–10006, 2020.

[28] Zeyi Huang, Haohan Wang, Eric P Xing, and Dong Huang. Self-challenging improves cross-domain generalization. In *ECCV*, pages 124–140, 2020.

[29] Yusuke Iwasawa and Yutaka Matsuo. Test-time classifier adjustment module for model-agnostic domain generalization. *NeurIPS*, 34:2427–2440, 2021.

[30] Minguk Jang and Sae-Young Chung. Test-time adaptation via self-training with nearest neighbor information. In *ICLR*, 2023.

[31] Minguk Jang, Sae-Young Chung, and Hye Won Chung. Test-time adaptation via self-training with nearest neighbor information. In *The Eleventh International Conference on Learning Representations*, 2023.

[32] Daehee Kim, Youngjun Yoo, Seunghyun Park, Jinkyu Kim, and Jaekoo Lee. Selfreg: Self-supervised contrastive regularization for domain generalization. In *ICCV*, pages 9619–9628, 2021.

[33] Jaeill Kim, Suhyun Kang, Duhun Hwang, Jungwook Shin, and Wonjong Rhee. Vne: An effective method for improving deep representation by manipulating eigenvalue distribution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3799–3810, 2023.

[34] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *ICML*, pages 5637–5664, 2021.

[35] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *Technical report, University of Toronto*, 2009.

[36] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation. In *International Conference on Machine Learning*, pages 5815–5826, 2021.

[37] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning, ICML*, volume 3, page 2, 2013.

[38] Jonghyun Lee, Dahuin Jung, Saehyung Lee, Junsung Park, Juhyeon Shin, Uiwon Hwang, and Sungroh Yoon. Entropy is not enough for test-time adaptation: From the perspective of disentangled factors. In *ICLR*, 2024.

[39] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *ICCV*, pages 5542–5550, 2017.

 https://doi.org/10.52202/079017-3990

[40] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Learning to generalize: Meta-learning for domain generalization. In *AAAI*, 2018.

[41] Da Li, Jianshu Zhang, Yongxin Yang, Cong Liu, Yi-Zhe Song, and Timothy M Hospedales. Episodic training for domain generalization. In *ICCV*, pages 1446–1455, 2019.

[42] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C Kot. Domain generalization with adversarial feature learning. In *CVPR*, pages 5400–5409, 2018.

[43] Wuyang Li, Xinyu Liu, and Yixuan Yuan. Sigma: Semantic-complete graph matching for domain adaptive object detection. In *CVPR*, 2022.

[44] Wuyang Li, Xinyu Liu, and Yixuan Yuan. Sigma++: Improved semantic-complete graph matching for domain adaptive object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(7):9022–9040, 2023.

[45] Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. Deep domain generalization via conditional invariant adversarial networks. In *ECCV*, pages 624–639, 2018.

[46] Jian Liang, Ran He, and Tieniu Tan. A comprehensive survey on test-time adaptation under distribution shifts. *arXiv preprint arXiv:2303.15361*, 2023.

[47] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *ICML*, pages 6028–6039. PMLR, 2020.

[48] Chang Liu, Xinwei Sun, Jindong Wang, Haoyue Tang, Tao Li, Tao Qin, Wei Chen, and Tie-Yan Liu. Learning causal semantic representation for out-of-distribution prediction. *NeurIPS*, 34, 2021.

[49] Yuejiang Liu, Parth Kothari, Bastien Van Delft, Baptiste Bellot-Gurlet, Taylor Mordan, and Alexandre Alahi. Ttt++: When does self-supervised test-time training fail or thrive? *NeurIPS*, 34:21808–21820, 2021.

[50] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *ICML*, pages 4114–4124, 2019.

[51] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *NeurIPS*, 33:11525–11538, 2020.

[52] Eliane Maria Loiola, Nair Maria Maia De Abreu, Paulo Oswaldo Boaventura-Netto, Peter Hahn, and Tania Querido. A survey for the quadratic assignment problem. *European journal of operational research*, 176(2):657–690, 2007.

[53] Seonwoo Min, Nokyung Park, Siwon Kim, Seunghyun Park, and Jinkyu Kim. Grounding visual representations with texts for domain generalization. In *European Conference on Computer Vision*, pages 37–53. Springer, 2022.

[54] Zachary Nado, Shreyas Padhy, D Sculley, Alexander D'Amour, Balaji Lakshminarayanan, and Jasper Snoek. Evaluating prediction-time batch normalization for robustness under covariate shift. *arXiv preprint arXiv:2006.10963*, 2020.

[55] Hyeonseob Nam, HyunJae Lee, Jongchan Park, Wonjun Yoon, and Donggeun Yoo. Reducing domain gap by reducing style bias. In *CVPR*, pages 8690–8699, 2021.

[56] A Tuan Nguyen, Thanh Nguyen-Tang, Ser-Nam Lim, and Philip HS Torr. Tipi: Test time adaptation with transformation invariance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24162–24171, 2023.

[57] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *ICML*, pages 16888–16905. PMLR, 2022.

[58] Prashant Pandey, Mrigank Raman, Sumanth Varambally, and Prathosh Ap. Generalization on unseen domains via inference-time label-preserving target projections. In *CVPR*, pages 12924–12933, 2021.

[59] Renjie Pi, Tianyang Han, Wei Xiong, Jipeng Zhang, Runtao Liu, Rui Pan, and Tong Zhang. Strengthening multimodal large language model with bootstrapped preference optimization, 2024.

[60] Renjie Pi, Jianshu Zhang, Jipeng Zhang, Rui Pan, Zhekai Chen, and Tong Zhang. Image textualization: An automatic framework for creating accurate and detailed image descriptions, 2024.

[61] Vihari Piratla, Praneeth Netrapalli, and Sunita Sarawagi. Efficient domain generalization via common-specific low-rank decomposition. In *ICML*, pages 7728–7738, 2020.

[62] Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations*, 2020.

[63] Wojciech Samek, Grégoire Montavon, Andrea Vedaldi, Lars Kai Hansen, and Klaus-Robert Müller. *Explainable AI: interpreting, explaining and visualizing deep learning*, volume 11700. Springer Nature, 2019.

[64] Maximilian Seitzer, Max Horn, Andrii Zadaianchuk, Dominik Zietlow, Tianjun Xiao, Carl-Johann Simon-Gabriel, Tong He, Zheng Zhang, Bernhard Schölkopf, Thomas Brox, et al. Bridging the gap to real-world object-centric learning. *arXiv preprint arXiv:2209.14860*, 2022.

[65] Ruoqi Shen, Sébastien Bubeck, and Suriya Gunasekar. Data augmentation as feature manipulation. In *International conference on machine learning*, pages 19773–19808. PMLR, 2022.

[66] Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information. *arXiv preprint arXiv:1703.00810*, 2017.

[67] Richard Sinkhorn. A relationship between arbitrary positive matrices and doubly stochastic matrices. *The annals of mathematical statistics*, 35(2):876–879, 1964.

[68] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *ECCV*, pages 443–450, 2016.

[69] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *ICML*, pages 9229–9248. PMLR, 2020.

[70] Devavrat Tomar, Guillaume Vray, Jean-Philippe Thiran, and Behzad Bozorgtabar. Un-mixing test-time normalization statistics: Combatting label temporal correlation. *arXiv preprint arXiv:2401.08328*, 2024.

[71] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 9(11), 2008.

[72] Vladimir N Vapnik. An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5):988–999, 1999.

[73] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *CVPR*, pages 5018–5027, 2017.

[74] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. In *NeurIPS*, pages 5334–5344, 2018.

[75] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *ICLR*, 2021.

[76] Jindong Wang, Cuiling Lan, Chang Liu, Yidong Ouyang, Tao Qin, Wang Lu, Yiqiang Chen, Wenjun Zeng, and Philip Yu. Generalizing to unseen domains: A survey on domain generalization. *TKDE*, 2022.

[77] Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7201–7211, 2022.

[78] Runzhong Wang, Junchi Yan, and Xiaokang Yang. Neural graph matching network: Learning lawler's quadratic assignment problem with extension to hypergraph and multiple-graph matching. *IEEE TPAMI*, 44(9):5261–5279, 2021.

[79] Shuai Wang, Daoan Zhang, Zipei Yan, Jianguo Zhang, and Rui Li. Feature alignment and uniformity for test time adaptation. In *CVPR*, 2023.

[80] Yufei Wang, Haoliang Li, and Alex C Kot. Heterogeneous domain generalization via domain mixup. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3622–3626, 2020.

[81] Zehao Xiao, Xiantong Zhen, Shengcai Liao, and Cees GM Snoek. Energy-based test sample adaptation for domain generalization. In *ICLR*, 2023.

[82] Minghao Xu, Jian Zhang, Bingbing Ni, Teng Li, Chengjie Wang, Qi Tian, and Wenjun Zhang. Adversarial domain adaptation with domain mixup. In *AAAI Conference on Artificial Intelligence*, pages 6502–6509, 2020.

 https://doi.org/10.52202/079017-3990

[83] Zhenlin Xu, Deyi Liu, Junlin Yang, Colin Raffel, and Marc Niethammer. Robust and generalizable visual representation learning via random convolutions. In *ICLR*, 2021.

[84] Shen Yan, Huan Song, Nanxiang Li, Lincan Zou, and Liu Ren. Improve unsupervised domain adaptation with mixup training. *arXiv preprint arXiv:2001.00677*, 2020.

[85] Hanlin Zhang, Yi-Fan Zhang, Weiyang Liu, Adrian Weller, Bernhard Schölkopf, and Eric P Xing. Towards principled disentanglement for domain generalization. In *CVPR*, pages 8024–8034, 2022.

[86] Lei Zhang, Ji-Fu Li, and Wei Wang. Semi-supervised domain generalization with known and unknown classes. *NeurIPS*, 36, 2023.

[87] Marvin Mengxin Zhang, Sergey Levine, and Chelsea Finn. Memo: Test time robustness via adaptation and augmentation. In *NeurIPS*, 2022.

[88] Marvin Mengxin Zhang, Henrik Marklund, Nikita Dhawan, Abhishek Gupta, Sergey Levine, and Chelsea Finn. Adaptive risk minimization: Learning to adapt to domain shift. In *Advances in Neural Information Processing Systems*, 2021.

[89] Yabin Zhang, Bin Deng, Hui Tang, Lei Zhang, and Kui Jia. Unsupervised multi-class domain adaptation: Theory, algorithms, and practice. *IEEE TPAMI*, 2020.

[90] Yabin Zhang, Minghan Li, Ruihuang Li, Kui Jia, and Lei Zhang. Exact feature distribution matching for arbitrary style transfer and domain generalization. In *CVPR*, 2022.

[91] Yabin Zhang, Wenjie Zhu, Hui Tang, Zhiyuan Ma, Kaiyang Zhou, and Lei Zhang. Dual memory networks: A versatile adaptation approach for vision-language models. In *CVPR*, 2024.

[92] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *TPAMI*, 2022.

[93] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Deep domain-adversarial image generation for domain generalisation. In *AAAI*, pages 13025–13032, 2020.

[94] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Learning to generate novel domains for domain generalization. In *ECCV*, pages 561–578, 2020.

[95] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. In *ICLR*, 2021.

[96] Xiao Zhou, Yong Lin, Weizhong Zhang, and Tong Zhang. Sparse invariant risk minimization. In *ICML*, pages 27222–27244, 2022.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims in the abstract and introduction accurately represent the paper's contributions and scope.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We provide this part in the supplementary document.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: All assumptions and definitions have been clearly stated.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.

https://doi.org/10.52202/079017-3990

- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We have fully disclosed all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: After internal review and patent approval, we will release the code.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.

- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We have provided all the training and test details.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We have provided the standard deviation.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We have provided these details in the supplementary.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.

https://doi.org/10.52202/079017-3990

- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: We have read the Code of Ethics and will always adhere to the principles it requires.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We have provided this part in the supplementary.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: [NA]

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We use publicly available datasets and code, and we have properly cited their papers or project addresses.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: [NA]

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: [NA]

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: [NA]

https://doi.org/10.52202/079017-3990

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.