A Cognitive Framework for Learning Debiased and Interpretable Representations via Debiasing Global Workspace

Jinyung Hong[1] **Eun Som Jeon**[2] **Changhoon Kim**[1] **Keun Hee Park**[1]

Utkarsh Nath[1] **Yezhou Yang**[1] **Pavan Turaga**[3] **Theodore P. Pavlic**[1,4]

[1]**School of Computing and Augmented Intelligence**
[3]**School of Arts, Media, and Engineering**
[4]**School of Life Sciences**
Arizona State University, Tempe, AZ 85281, USA

[2]**Department of Computer Science and Engineering**
Seoul National University of Science and Technology, Seoul, 01811, Korea

Editors: Marco Fumero, Clementine Domine, Zorah Lähner, Donato Crisostomi, Luca Moschella, Kimberly Stachenfeld

Abstract

When trained on biased datasets, Deep Neural Networks (DNNs) often make predictions based on attributes derived from features spuriously correlated with target labels. This is especially problematic if these irrelevant features are easier for the model to learn than the truly relevant ones. Many existing debiasing methods have been proposed to address this issue, but they often require predefined bias labels and entail significantly increased computational complexity by incorporating additional auxiliary models. Instead, we provide an orthogonal perspective from existing approaches, inspired by cognitive science, specifically Global Workspace Theory (GWT). Our method, *Debiasing Global Workspace* (DGW), is a novel debiasing framework that consists of specialized modules and a shared workspace, allowing for increased modularity and improved debiasing performance. Furthermore, DGW improves the transparency of decision-making processes by visualizing which features of inputs the model focuses on during training and inference through attention masks. We begin by proposing an instantiation of GWT for the debiasing method. We then outline the implementation of each component within DGW. Finally, we validate our method across various biased datasets, proving its effectiveness in mitigating biases and improving model performance.

1 Introduction

Deep Neural Networks (DNNs) have achieved remarkable advancements across various domains, such as image classification (He et al., 2019; Xie et al., 2020), generation (Wang and Gupta, 2016; Kataoka et al., 2016), and segmentation (Luo et al., 2017; Zheng et al., 2014). However, DNNs often show limited generalization capability to out-of-distribution (OOD) data and are susceptible

Proceedings of the II edition of the Workshop on Unifying Representations in Neural Models (UniReps 2024).

to biases present in their training datasets (Torralba and Efros, 2011). These biases occur when irrelevant features, such as background color, correlate with target labels, causing models to rely on these features for making predictions (Geirhos et al., 2020). This reliance on biased features leads to poor performance when the model encounters new data that do not share the same biases. Biased datasets possess many *bias-aligned* samples, where irrelevant features correlate with the labels, and a small number of *bias-conflicting* samples, where these features do not align with the labels. Models trained on such data tend to be disproportionately influenced by bias-aligned samples, leading to poor generalization (Hendrycks et al., 2021b,a).

Various debiasing methods have been proposed to prevent a network from relying on spurious correlations when trained on a biased dataset. Some methods assume that biased features are "easier" to learn than robust ones, leading to the use of auxiliary models that exploit these biased features to guide the main model's training (Nam et al., 2020; Sanh et al., 2020). Strategies such as re-weighting samples (Liu et al., 2021; Nam et al., 2020) and data augmentation (Kim et al., 2021; Lee et al., 2021) are common but often struggle with insufficiently diverse samples. Other approaches involve identifying specific biases prior to training (Hong and Yang, 2021; Kim et al., 2019; Li and Vasconcelos, 2019; Sagawa et al., 2019), allowing the model to ignore or correct these biases. Although effective, this requires accurate bias identification and extensive manual labeling (Bahng et al., 2020; Tartaglione et al., 2021).

In this work, we depart from the above perspectives and focus on a novel and completely different approach to implement a debiasing framework. In modern ML and AI, it has been argued that it is better to build an intelligent system from many interacting specialized modules rather than a single "monolithic" entity to deal with a broad spectrum of conditions and tasks (Goyal and Bengio, 2022; Minsky, 1988; Robbins, 2017). Toward this end, we focus on Global Workspace Theory (GWT), a framework from cognitive science proposed to underlie perception, executive function, and consciousness. GWT is a crucial element of modern cognitive science that models human consciousness arising from integrating and broadcasting information across specialized, unconscious processes in the brain (Baars, 1993, 2005). Many recent studies proposing a deep-learning implementation of GWT (Bengio, 2017; Goyal et al., 2021; VanRullen and Kanai, 2021) have demonstrated their effectiveness in allowing a model to have: general-purpose functionality, increased modularity, improved performance, and interpretable representation learning. This perspective is expected to be well suited for application in implementing debiasing methods.

Therefore, we propose the *Debiasing Global Workspace* (DGW), a novel instantiation of GWT for debiasing to eliminate the negative effect of the misleading correlations. Our debiasing approach involves specialized modules (acting as the specialists in GWT) and an attention-based information bottleneck (acting as the global workspace in GWT). This allows the model to achieve straightforward, functional modularity, and effective debiasing performance while providing interpretable representation by visualizing which attributes are essential for accurate predictions and which are irrelevant and likely to cause errors.

The remainder of this paper is organized as follows. We begin in Section 2 with a review of related work and relevant background literature. Then, in Section 3, we propose a conceptual modification of the GWT to implement a debiasing method. This involves defining specialized modules and the shared global workspace (Section 3.1). Then, we provide a step-by-step framework for defining the essential deep-learning components of our debiasing model within an AI system (Section 3.2). In Section 4, we empirically test our method on biased datasets, including Colored MNIST, Corrupted CIFAR10, and Biased FFHQ, and demonstrate that DGW effectively separates and understands intrinsic and biased features through both performance metrics and visualizations. Finally, we conclude with a discussion of future work and limitations of our approach in Section 5.

2 Related Work

2.1 Debiasing Methods for Deep Neural Networks

Relative to existing debiasing methods for DNNs, our work aims to reduce training complexity while improving generalization performance. We survey those existing methods here.

Debiasing with predefined forms of bias or specific bias labels. One approach to debiasing is to identify specific biases prior to training (Hong and Yang, 2021; Kim et al., 2019; Li and Vasconcelos,

2019; Sagawa et al., 2019). The model then learns to ignore or correct these biases. Although effective, it depends on accurately identifying biases beforehand, which can be challenging. Another approach uses bias labels to tag the data (Bahng et al., 2020; Tartaglione et al., 2021), which allows the model to differentiate between biased and unbiased data during training. This improves learning but requires extensive manual labeling.

Debiasing using the easy-to-learn heuristic. Biases are "easier" for models to learn (Nam et al., 2020) than intrinsic features. Techniques like dynamic training schemes, re-weighting samples, and data augmentation (Geirhos et al., 2018; Lee et al., 2021; Minderer et al., 2020; Li and Vasconcelos, 2019; Lim et al., 2023) help models focus on unbiased features. However, these methods do not perform well if training samples have low diversity. Complex models can learn invariant features or correct representations but are difficult to design and train (Tu et al., 2022; Zhao et al., 2020; Agarwal et al., 2020; Bahng et al., 2020; Geirhos et al., 2018; Goel et al., 2020; Kim et al., 2019; Li et al., 2020; Minderer et al., 2020; Tartaglione et al., 2021; Wang et al., 2020).

Others. SelecMix (Hwang et al., 2022) uses an auxiliary contrastive model with new training samples that mix pairs with similar labels but different biases or different labels but similar biases. This method is effective but adds significant training complexity. χ^2 model (Zhang et al., 2023) learns debiased representations by identifying Intermediate Attribute Samples (IAS) and using a χ-structured metric learning objective. However, its reliance on training dynamics to identify IASs makes it different from our approach and out of the scope of our study.

2.2 Deep Learning and Global Workspace Theory

In neuroscience and cognitive science, there are ongoing efforts to identify neural correlates of consciousness, as reviewed by Seth and Bayne (2022), and to form explanatory theories of consciousness (ToC). One such theory is the Global Workspace Theory (GWT) (Baars, 1993; Dehaene and Changeux, 2011; Mashour et al., 2020), which is inspired by the "blackboard architecture" used in artificial intelligence. In this architecture, a centralized, shared blackboard resource facilitates the exchange of information between specialized processors.

Recent studies have aimed to bridge the gap between neuroscience and deep learning, focusing on practical solutions to implement a GWT using current deep learning components while also incorporating organizational principles from functionally equivalent brain mechanisms (Goyal and Bengio, 2022; Minsky, 1988; Robbins, 2017; Goyal et al., 2021; Hong et al., 2024). Bengio (2017) emphasized learning high-level concepts by selecting key elements through attention, forming a low-dimensional conscious state similar to language, which aids in better representation learning. Mashour et al. (2020) details GWT's implementation in neuroscience, suggesting that consciousness arises from extensive information sharing across brain regions via a central network of neurons.

Inspired by GWT, our Debiasing Global Workspace (DGW) framework manages intrinsic and biased attributes in neural networks. DGW integrates information from intrinsic and bias specialists, ensuring that disentangled representations are used in decision making. Unlike prior works that focus on monolithic architectures or general-purpose learning, our approach uniquely applies these theories to the specific problem of debiasing neural networks.

3 Method

We propose the Debiasing Global Workspace (DGW), an instantiation of GWT for debiasing. DGW learns the composition of attributes in a dataset and provides interpretable explanations for the model's decisions. We introduce the conceptual framework of GWT for debiasing first (Section 3.1), its implementation in a deep learning framework next (Section 3.2), and the training objectives last (Section 3.3).

3.1 The Conceptual Instantiation of Debiasing Global Workspace

Figure 1 shows a conceptual overview of our proposed DGW framework. The conceptual flow of the DGW proceeds through a sequence of steps that we describe in detail here.

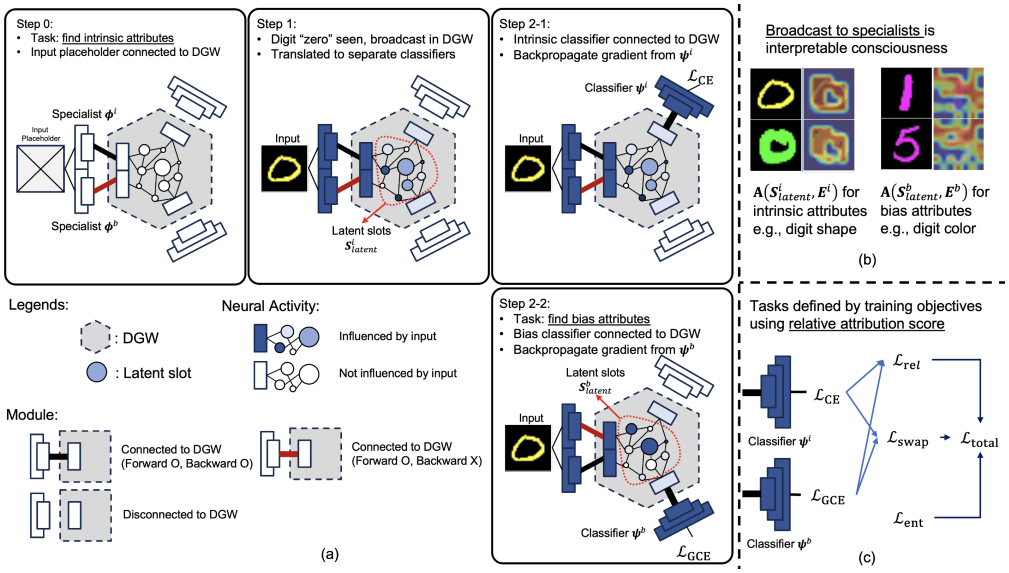


Figure 1: Conceptual framework of Debiasing Global Workspace (DGW). (a) Section 3.1: When attention selects inputs from specialists (Step 0), its latent-space activation is copied into DGW and immediately translated into representations suitable for each module (Step 1). We control which module is mobilized into the workspace to receive and process the corresponding data effectively. For example, upon recognizing digit "zero," the corresponding classifiers are activated in the workspace. The classifier ψ^i is initiated for intrinsic attributes (Step 2-1), and the classifier ψ^b is activated for learning bias attributes (Step 2-2). (b) Broadcast in Section 3.2: The information broadcast in DGW can demonstrate interpretable representation for attribute learning. (c) Section 3.3: Unlike the original GWT, where task definitions can be preset in Step 0, we address them using our training objectives using relative attribution score. Figure inspired by VanRullen and Kanai (2021, Fig. 3)

Step 0. For learning disentangled representations of intrinsic and biased attributes, we introduce two specialists: intrinsic ϕ^i and biased ϕ^b. In the original GWT, the specialists connect to the global workspace before any stimulus appears, coupling their latent spaces bidirectionally with the workspace. We modify this arrangement to so that different information will be backpropagated to the two specialists separately (black and red connections between specialists and DGW in Step 0 in Fig. 1). Specifically, the intrinsic and bias specialists function identically in the forward pass. However, during the backpropagation stage, only the intrinsic attribute encoder updates its parameters and learns, while the bias attribute encoder remains frozen and does not undergo parameter updates when the model is tasked to find intrinsic attributes.

Step 1. The DGW acts as an independent and intermediate shared latent space trained to perform unsupervised neural translation between the C latent spaces of the specialized modules. The translation system is optimized to ensure that successive translation and back translation (e.g., a cycle from A to B, then back to A) return the original input (Goyal et al., 2021; VanRullen and Kanai, 2021). We implement specific operations to mimic the translation system by leveraging residual operations (He et al., 2016) and a variant of mixup (Verma et al., 2019).

Posner (1994) argues that attention determines what information is consciously perceived and what is discarded in brains. In GWT, attention selects the information that enters the workspace. When a specific module is connected to the workspace through attention, its latent space activation vector is copied into the DGW. This internal copy serves as a bidirectional connection interface between the corresponding module and the DGW.

When a new stimulus, such as the digit "zero," appears, its latent activity is transferred to the corresponding internal copy inside the workspace, initiating a broadcast to all other domains. This shared latent space ($\mathbf{S}^i_{\text{latent}}$ in Fig. 1) uses translations and back translations from all modules to compute and train using error backpropagation. We introduce a recurrent, top-down pathway, which can sometimes be considered a key to account for the global ignition property observed in the brain

when an input reaches consciousness, and the corresponding module is mobilized into the conscious global workspace (VanRullen and Kanai, 2021).

Step 2. The incoming information is then immediately broadcasted and translated (through the shared latent space) into the latent space of all other modules. In GWT, this translation process is automatic. However, we modify this to force the learning of intrinsic and biased attribute representations by using different loss functions. Specifically, we force the classifier $\boldsymbol{\phi}^i$ to learn intrinsic attributes by error backpropagation from specific training objectives (Step 2-1 in Fig. 1). Step 2-2 simultaneously forces the connection to the classifier $\boldsymbol{\psi}^b$ and limits backpropagation to the intrinsic specialist $\boldsymbol{\phi}^i$ to learn the representations of bias attributes.

3.2 Roadmap to Implement Debiasing Global Workspace

Here, we present our deep-learning-based implementation of the DGW. It combines and organizes existing components for effective debiasing frameworks in a way that is consistent with the cognitive-science-inspired DGW framework described above.

Two specialized modules and the shared workspace. DGW uses two independent specialists, the intrinsic attribute encoder $\boldsymbol{\phi}^i$ and the bias attribute encoder $\boldsymbol{\phi}^b$. From these, we derive concatenated features $\mathbf{E} \triangleq [\boldsymbol{\phi}^i(\mathbf{x}); \boldsymbol{\phi}^b(\mathbf{x})] \in \mathbb{R}^{L \times D}$. To connect specialists and the shared workspace, we define $\mathbf{e} \in \{\mathbf{E}^i, \mathbf{E}^b\}$, where $\mathbf{E}^i = [\boldsymbol{\phi}^i(\mathbf{x}); \mathrm{sg}(\boldsymbol{\phi}^b(\mathbf{x}))]$ and $\mathbf{E}^b$ is vice versa, with $\mathrm{sg}(\cdot)$ as the stop-gradient operator. We introduce the Global Latent Attention (GLA) module, which acts as a shared workspace that encourages synchronization within the input feature vector $\mathbf{E}$ through a latent feature representation $\mathbf{S}_{\text{latent}}$.

Latent-slot binding specific to each input. The GLA module uses a set number of latent embeddings or latent slots C. These latent slots represent the learnable embedding vectors in the DGW and provide competitive attention (Vaswani et al., 2017) on input features $\mathbf{e}$. We define $\mathbf{s}_{\text{latent}} \in \{\mathbf{S}^i_{\text{latent}}, \mathbf{S}^b_{\text{latent}}\} \in \mathbb{R}^{C \times D}$ where C^i is the number of slots for intrinsic features and C^b for bias features, with $C = C^i + C^b$. The attention mechanism is such that:

$$\mathbf{A}(\mathbf{e}, \mathbf{s}_{\text{latent}}) = \mathrm{softmax}\left(\frac{k(\mathbf{e}) \cdot q(\mathbf{s}_{\text{latent}})^\top}{\sqrt{D}}\right) \in \mathbb{R}^{C \times L}, \tag{1}$$

where, k, q are linear projection matrices, and the softmax function normalizes the slots, creating competition among them. The slots are refined iteratively using the following:

$$\mathbf{s}^{(n+1)}_{\text{latent}} = \mathrm{GRU}\left(\mathbf{s}^{(n)}_{\text{latent}}, \mathrm{Normalize}\left(\mathbf{A}(\mathbf{e}, \mathbf{s}^{(n)}_{\text{latent}})^\top\right) \cdot v\left(\mathbf{E}\right)\right), \tag{2}$$

where: $\mathbf{s}^{(n)}_{\text{latent}}$ is the latent-slot representation after n iterations, GRU (Cho et al., 2014) is a recurrent neural network, and v is another liner projection matrix. The initial slots $\mathbf{s}^{(0)}_{\text{latent}}$ are initialized with learnable queries following Jia et al. (2022).

The above computations can be considered to implement a shared global workspace (Goyal et al., 2021; Hong et al., 2024) as they allow different parts of the model to compete for attention and integrate and broadcast information similar to GWT.

Broadcast updated information to specialists. Specialists update their states using information from the shared workspace. The inverted cross-attention mechanism allows specialists to query and interact with updated latent slots $\mathbf{s}^{(n+1)}_{\text{latent}}$, updating their states through:

$$\bar{\mathbf{e}} = \mathbf{e} \oplus \left(\mathbf{A}\left(\mathbf{s}^{(n+1)}_{\text{latent}}, \mathbf{e}\right) \cdot v\left(\mathbf{s}^{(n+1)}_{\text{latent}}\right)\right) \in \mathbb{R}^{L \times D}, \tag{3}$$

where v is a linear projection matrix. Here, $\oplus$ can be instantiated with various computational operations that implement different forms of information broadcast, including a residual connection (He et al., 2016). The other way of operation is a modified version of Manifold Mixup (Verma et al., 2019), which interpolates feature embeddings to capture higher-level information:

$$\bar{\mathbf{e}} = \mathrm{Mix}_\alpha\left(\mathbf{e}, \left(\mathbf{A}\left(\mathbf{s}^{(n+1)}_{\text{latent}}, \mathbf{e}\right) \cdot v\left(\mathbf{s}^{(n+1)}_{\text{latent}}\right)\right)\right),$$

where: $\text{Mix}_\alpha(a, b) = \alpha \cdot a + (1 - \alpha) \cdot b$ and $\alpha \sim \text{Beta}(\beta, \beta)$. The updated feature vector $\bar{\mathbf{e}}$ is then fed to the classifier $\boldsymbol{\psi}^i$ and $\boldsymbol{\psi}^b$. We compare the performance of using residual connections versus our modified Manifold Mixup in Section 4.1.

In GWT, the information broadcast through the global workspace is a necessary and sufficient condition for conscious perception (VanRullen and Kanai, 2021). Intuitively, the attention mask $\mathbf{A}(\mathbf{s}_{\text{latent}}^{(n+1)}, \mathbf{e})$ can be seen as artificial phenomenal consciousness, indicating the immediate subjective experience of sensations and perceptions. These non-negative relevance scores depend on $\mathbf{x}$ through the averaged attention weight, allowing us to show interpretable representations for intrinsic and biased attributes in our analysis (Section 4.2).

3.3 Training Objectives

Here, we summarize the objective functions to train our framework. We have two linear classifiers $\boldsymbol{\psi}^i$ and $\boldsymbol{\psi}^b$ that take the updated concatenated vector $\bar{\boldsymbol{e}}$ from the previous module as input to predict the target label y. Our training objectives consist of: i) the relative attribute score learning phase, and ii) the attribute composition phase.

Relative attribute score learning phase. In this phase, we define two tasks within the conceptual framework: identification of intrinsic attributes and identification of biased attributes. Without specific information about bias types, we utilize the relative difficulty score of each data sample, as proposed by Nam et al. (2020). Specifically, we train $\boldsymbol{\phi}^b$, $\mathbf{S}_{\text{latent}}^b$, and $\boldsymbol{\psi}^b$ to focus on bias attributes using generalized cross entropy (GCE) (Zhang and Sabuncu, 2018), while $\boldsymbol{\phi}^i$, $\mathbf{S}_{\text{latent}}^i$ and $\boldsymbol{\psi}^i$ are trained with the cross entropy (CE) loss. Samples with high CE loss from $\boldsymbol{\psi}^b$ are considered bias conflicting compared to those with low CE loss. We define the relevance-score function:

$$\text{Score}(\bar{\boldsymbol{e}}, y) \triangleq CE(\boldsymbol{\psi}^b(\bar{\boldsymbol{e}}), y) \Big/ \Big(CE(\boldsymbol{\psi}^i(\bar{\boldsymbol{e}}), y) + CE(\boldsymbol{\psi}^b(\bar{\boldsymbol{e}}), y) \Big). \tag{4}$$

Thus, the objective function is defined using the above relative difficulty score for each data sample:

$$\mathcal{L}_{\text{rel}} \triangleq \text{Score}(\bar{\boldsymbol{e}}, y) \cdot CE(\boldsymbol{\psi}^i(\bar{\boldsymbol{e}}), y) + \lambda_{\text{rel}} GCE(\boldsymbol{\psi}^b(\bar{\boldsymbol{e}}), y),$$

where weight λ_{rel} adjusts the balance between the two loss terms. This loss function balances learning between intrinsic and biased attributes, ensuring effective identification and separation of these attributes during the training phase.

Attribute-composition phase. We swap the disentangled latent vectors among the training sets (Lee et al., 2021). We randomly permute the intrinsic and bias features in each mini-batch, creating $\mathbf{E}_{\text{swap}} = [\boldsymbol{\phi}^i(\mathbf{x}); \boldsymbol{\phi}_{\text{swap}}^b(\mathbf{x})]$ where $\boldsymbol{\phi}_{\text{swap}}^b(\mathbf{x})$ denotes the randomly permuted bias attributes. This process produces augmented bias-conflicting latent vectors. As in the definition of $\mathbf{e}$, we define $\mathbf{e}_{\text{swap}} \in \{\mathbf{E}_{\text{swap}}^i, \mathbf{E}_{\text{swap}}^b\}$ and generate $\bar{\mathbf{e}}_{\text{swap}}$ following the same process described in eqs 1, 2 and 3. The objective function for this phase is:

$$\mathcal{L}_{\text{swap}} \triangleq \text{Score}(\bar{\boldsymbol{e}}, y) \cdot CE(\boldsymbol{\psi}^i(\bar{\boldsymbol{e}}_{\text{swap}}), y) + \lambda_{\text{swap}} GCE(\boldsymbol{\psi}^b(\bar{\boldsymbol{e}}_{\text{swap}}), \tilde{y}),$$

where: $\tilde{y}$ denotes the target labels for the permuted bias attributes $\boldsymbol{\phi}_{\text{swap}}^b(\mathbf{x})$, the weight λ_{swap} adjusts the balance between two loss terms, and the relevance score $\text{Score}(\bar{\boldsymbol{e}}, y)$ from eq. 4 is reused from $\mathcal{L}_{\text{rel}}$ to reduce computational complexity. This loss function swaps bias features so that the model learns to handle a wider variety of bias-conflicting samples, improving its ability to generalize beyond the specific biases present in the training data. Consequently, the model becomes more robust as it learns to focus on intrinsic features while disregarding spurious correlations, resulting in better performance on unbiased data. Furthermore, augmenting the training data in this manner helps the model generalize better to new, unseen data by exposing it to a wider range of possible biases during training.

Entropy regularization. We empirically incorporate an additional regularization term on the latent slot attention mask to enhance performance:

$$\mathcal{L}_{\text{ent}} \triangleq H(\mathbf{A}(\mathbf{s}_{\text{latent}}^{(n)}, \mathbf{e})) + H(\mathbf{A}(\mathbf{s}_{\text{latent}}^{(n)}, \mathbf{e}_{\text{swap}})),$$

where $\mathbf{A}(\mathbf{s}_{\text{latent}}^{(n)}, \mathbf{e})$ and $\mathbf{A}(\mathbf{s}_{\text{latent}}^{(n)}, \mathbf{e}_{\text{swap}})$ are attention masks from the last iteration of eq. 2. Minimizing entropy $H(\mathbf{A}) = H(a_1, \ldots, a_{|\mathbf{A}|}) = (1/|\mathbf{A}|) \sum_i -a_i \cdot \log(a_i)$ encourages the attention masks to be consistent across the input features captured by the latent slots. This regularization ensures that the model's attention remains focused and interpretable across different input scenarios.

Final loss. The total loss function $\mathcal{L}_{\text{total}} \triangleq \mathcal{L}_{\text{rel}} + \lambda_{\text{swap}} \cdot \mathcal{L}_{\text{swap}} + \lambda_{\text{ent}} \cdot \mathcal{L}_{\text{ent}}$ is a weighted combination of the above components, where the weights λ_{swap} and λ_{ent} adjust the relative importance of feature augmentation and entropy regularization, respectively. This comprehensive loss function ensures balanced training that enhances the model's ability to learn and generalize effectively while maintaining interpretability and robustness.

4 Experiments

Here, we present our experimental results, focusing on performance evaluation on various biased datasets (Section 4.1), interpretable analysis for attribute-centric representation learning (Section 4.2), and additional qualitative and quantitative analyses (Section 4.3).

Datasets. Following the previous work (Lee et al., 2021), we used the following three well-known benchmark datasets for debiasing methods to evaluate the performance and interpretability of DGW.

- **Colored MNIST (C-MNIST)** and **Corrupted CIFAR10 (C-CIFAR-10)**: These synthetic datasets are designed to test model generalization on unbiased test sets by varying the ratio of bias-conflicting samples (0.5%, 1%, 2%, and 5%).
- **Bias FFQH (BFFHQ)**: This real-world dataset from FFHQ (Karras et al., 2019) contains face images annotated with age (intrinsic attribute) and gender (bias attribute). Most of the samples are from young women and old men, creating a high correlation between age and gender. For BFFHQ, we included 0.5% bias-conflicting samples in the training set and used a bias-conflicting test set to ensure robust evaluation.

At inference time, we evaluated the models on clean data containing no bias-conflicting samples.

4.1 Performance Evaluation

Baselines. Our set of debiasing baselines includes the following six different approaches[1]: Vanilla network, HEX (Wang et al., 2018), EnD (Tartaglione et al., 2021), ReBias (Bahng et al., 2020), LfF (Nam et al., 2020), and LFA (Lee et al., 2021). Vanilla refers to the classification model trained only with the original cross-entropy (CE) loss without debiasing strategies. EnD leverages explicit bias labels, such as color labels in the C-MNIST dataset, during the training phase. HEX and ReBias assume an image's texture as a bias type, whereas LfF, LFA, and our method do not require any prior knowledge about the bias type. Furthermore, we configure a naive debiasing approach integrated with GWT implementation: V+CCT. CCT (Hong et al., 2024) proposed an instantiation of GWT applicable to implement an interpretable model. To compare our DGW, we simply configure the direction fusion of the Vanilla network with CCT as a GWT debiasing method.

Implementation details. Following the implementation details of Lee et al. (2021), we used a fully connected network for attribute encoders with three hidden layers for C-MNIST and ResNet-18 for C-CIFAR-10 and BFFHQ. We used a fully connected classifier with twice the hidden units to handle the combined output of the intrinsic attribute encoder ϕ^i and the bias attribute encoder ϕ^b.

During testing, only the intrinsic classifier $\psi^i(\mathbf{e})$ was used for the final predictions. We used batch sizes of 256 for C-MNIST and C-CIFAR-10, and 64 for BFFHQ, respectively. Two concepts and a size of 8 were used for C-MNIST, 5 and 16 for C-CIFAR-10, and 10 and 32 for BFFHQ, respectively.

[1]We only establish baselines that can be directly tested. For example, χ^2 (Zhang et al., 2023) is not included because its code is not publicly available, and SelecMix (Hwang et al., 2022) is not included because it is a data-augmentation method that differs from our method category and has high training complexity, taking over approximately four times longer than our method. Additionally, although the authors of SelecMix claim it runs on an RTX 3090, we found that our environment with a 24GB RTX A6000 could not handle the real-life dataset BFFHQ, indicating significant computational resource requirements.

Table 1: Test accuracy (%) on unbiased test sets of C-MNIST and C-CIFAR-10, and the bias-conflicting test set of BFFHQ with varying ratio of bias-conflicting samples. ($*$) denotes methods tailored to predefined forms of bias, ($^\circ$) methods using bias labels, ($\dagger$) methods relying on the easy-to-learn heuristic, and ($\ddagger$) methods combined with GWT. V+CCT indicates the direct integration of Vanilla and CCT. DGW+M refers to DGW with our mixup strategy, and DGW+R refers to DGW with residual connection. Performance for HEX and EnD is from (Lee et al., 2021), while results for Vanilla, ReBias, LfF, LFA, V+CCT and DGW are from our evaluation. The best-performing results are shown in bold, and the second-best results are underlined.

Dataset	Ratio (%)	Vanilla	HEX*	EnD$^\circ$	ReBias*	LfF†	LFA†	V+CCT‡	DGW+M‡	DGW+R‡
C-MNIST	0.5	$36.2_{\pm 1.8}$	$30.3_{\pm 0.8}$	$34.3_{\pm 1.2}$	**72.2**$_{\pm 1.5}$	$47.5_{\pm 3.0}$	$67.4_{\pm 1.7}$	$26.3_{\pm 1.1}$	$68.9_{\pm 2.8}$	$\underline{70.3}_{\pm 1.2}$
	1.0	$50.8_{\pm 2.3}$	$43.7_{\pm 5.5}$	$49.5_{\pm 2.5}$	**86.6**$_{\pm 0.6}$	$64.6_{\pm 2.5}$	$79.0_{\pm 1.0}$	$40.1_{\pm 2.1}$	$\underline{81.3}_{\pm 1.2}$	$77.4_{\pm 0.4}$
	2.0	$65.2_{\pm 2.1}$	$56.9_{\pm 2.6}$	$68.5_{\pm 2.2}$	**92.7**$_{\pm 0.3}$	$74.9_{\pm 3.7}$	$85.0_{\pm 0.8}$	$56.2_{\pm 1.8}$	$84.6_{\pm 1.5}$	$\underline{85.3}_{\pm 0.7}$
	5.0	$81.6_{\pm 0.6}$	$74.6_{\pm 3.2}$	$81.2_{\pm 1.4}$	**97.1**$_{\pm 0.6}$	$80.2_{\pm 0.9}$	$88.7_{\pm 1.3}$	$73.4_{\pm 0.8}$	$88.9_{\pm 0.2}$	$\underline{89.1}_{\pm 0.6}$
C-CIFAR-10	0.5	$22.8_{\pm 0.3}$	$13.9_{\pm 0.1}$	$22.9_{\pm 0.3}$	$20.8_{\pm 0.2}$	$25.0_{\pm 1.5}$	$27.9_{\pm 1.0}$	$15.2_{\pm 0.3}$	$\underline{29.6}_{\pm 0.5}$	**30.4**$_{\pm 2.2}$
	1.0	$26.2_{\pm 0.5}$	$14.8_{\pm 0.4}$	$25.5_{\pm 0.4}$	$24.4_{\pm 0.4}$	$31.0_{\pm 0.4}$	$34.3_{\pm 0.6}$	$20.6_{\pm 0.4}$	**34.9**$_{\pm 0.4}$	$\underline{33.6}_{\pm 2.4}$
	2.0	$31.1_{\pm 0.6}$	$15.2_{\pm 0.5}$	$31.3_{\pm 0.4}$	$29.6_{\pm 2.9}$	$38.3_{\pm 0.4}$	$40.3_{\pm 2.4}$	$24.6_{\pm 0.5}$	$\underline{41.3}_{\pm 1.0}$	**42.0**$_{\pm 1.9}$
	5.0	$42.0_{\pm 0.3}$	$16.0_{\pm 0.6}$	$40.3_{\pm 0.9}$	$41.1_{\pm 0.2}$	$48.8_{\pm 0.9}$	$\underline{50.3}_{\pm 1.1}$	$35.6_{\pm 0.8}$	**52.3**$_{\pm 0.8}$	$\underline{50.3}_{\pm 1.9}$
BFFHQ	0.5	$54.5_{\pm 0.6}$	$52.8_{\pm 0.9}$	$56.9_{\pm 1.4}$	$58.0_{\pm 0.2}$	$63.6_{\pm 2.9}$	$59.5_{\pm 3.8}$	$52.6_{\pm 1.1}$	**66.9**$_{\pm 1.0}$	$\underline{65.6}_{\pm 3.3}$

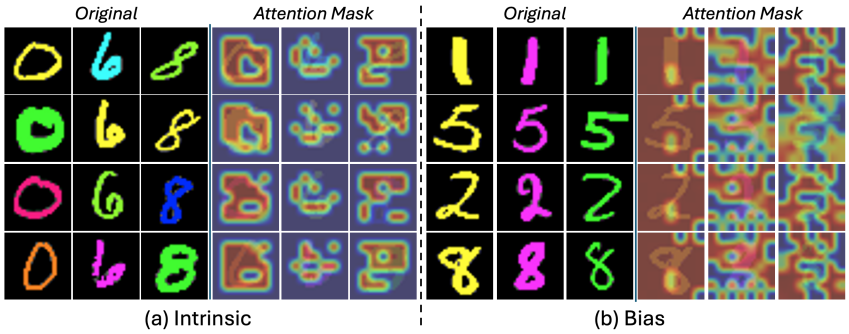


Figure 2: Visualization of $\mathbf{A}^i$ and $\mathbf{A}^b$ for the C-MNIST dataset

We trained our model and baselines with three trials and reported the averaged accuracy and standard deviation. More details of the experimental settings are explained in Appendix C.4.

Performance Comparison. Table 1 shows that ReBias outperforms DGW on C-MNIST because it uses additional predefined bias labels. This gives ReBias a specific advantage. However, DGW excels without needing predefined bias labels, making it more versatile. DGW, with all operators, also outperforms LFA in all datasets, demonstrating its robustness and flexibility in debiasing image classification tasks. Furthermore, the poor performance of V+CCT highlights the importance of finding the proper configuration for debiasing methods, indicating the effectiveness of our DGW configuration as a debiasing method.

4.2 Analysis for Interpretable Attribute Representation

To make the analysis of interpretable attribute representation learning in our model more intuitive, let us explore the attention mask patterns $\mathbf{A}(\mathbf{s}_{\text{latent}}^{(n+1)}, \mathbf{e})$ for the C-MNIST and C-CIFAR-10 datasets. In the broadcast in our formulation ($\mathbf{A}(\mathbf{s}_{\text{latent}}^{(n+1)}, \mathbf{e})$ in eq. 3)), DGW generates two attention masks: $\mathbf{A}^i = \mathbf{A}(\mathbf{S}_{\text{latent}}^i, \mathbf{E}^i)$ for intrinsic attributes, focusing on essential features like shape, and $\mathbf{A}^b = \mathbf{A}(\mathbf{S}_{\text{latent}}^b, \mathbf{E}^b)$ for biased attributes, capturing non-essential features like color.

For the C-MNIST dataset, intrinsic attention masks highlight the shapes of the digits, ignoring colors. For example, the digits "0," "6," and "8" consistently highlight shape regions (Fig. 2(a)), showing that the model focuses on shape for classification. In contrast, bias-attention masks highlight color regions, not shapes. Digits "1," "5," "2," and "8" in yellow/magenta/green show nearly identical masks (Fig. 2(b)), indicating a focus on color. This confirms that the biased components of DGW capture color information, which is irrelevant for digit recognition.

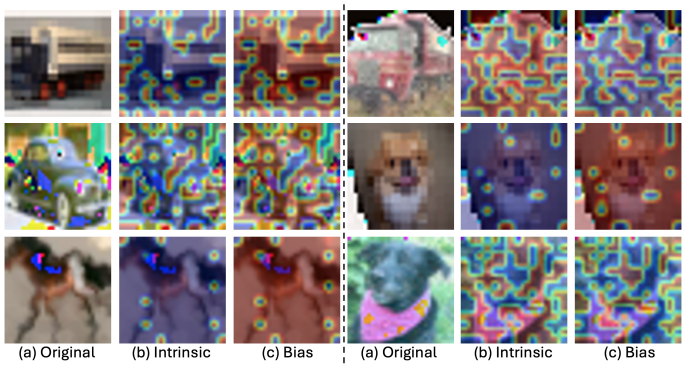


Figure 3: Visualization of $\mathbf{A}^i$ and $\mathbf{A}^b$ for the C-CIFAR10 dataset

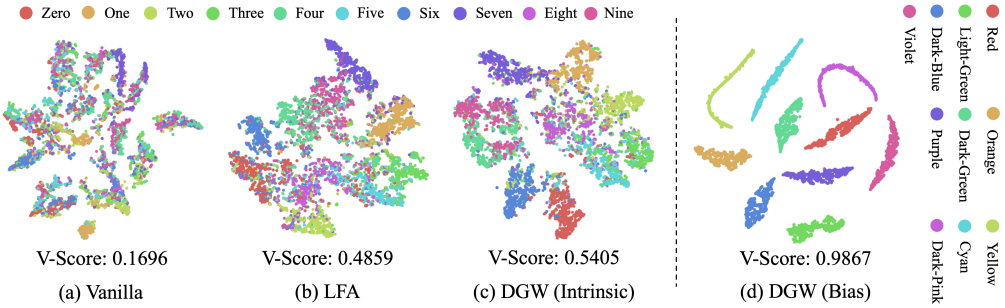


Figure 4: t-SNE plots for intrinsic and bias features on C-MNIST (with 0.5% setting).

For the C-CIFAR-10 dataset, intrinsic masks focus on uncorrupted parts of the images (Fig. 3(b)), highlighting true object features. For example, masks for a truck, car, dog, and horse highlight uncorrupted areas, avoiding noise. The bias masks, on the other hand, focus on corrupted areas, showing no overlap with intrinsic masks (Fig. 3(c)). This complementary relationship illustrates the effective segregation of essential (intrinsic) and non-essential (biased) information.

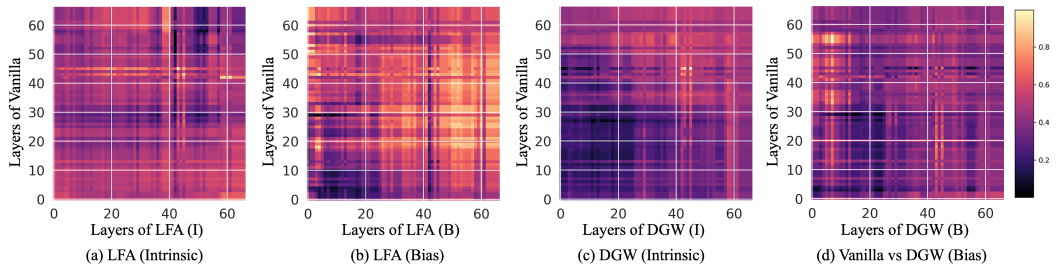
In summary, for C-MNIST, intrinsic masks focus on digit shapes, while bias masks focus on colors. For C-CIFAR-10, intrinsic masks highlight uncorrupted parts, and bias masks cover corrupted parts. This clear separation supports the model's robustness and interpretability, ensuring decisions are based on relevant features while ignoring spurious correlations. More visualization results can be found in Appendix C.5.

4.3 Quantitative and Qualitative Analysis

We provide additional analysis to compare our DGW (DGW+M in Table 1) method with Vanilla and LFA (Lee et al., 2021). More experimental results with different settings can be found in Appendix C.6.

t-SNE and Clustering. We measured clustering performance using t-SNE (van der Maaten and Hinton, 2008) and V-Score (Rosenberg and Hirschberg, 2007) on features from various models capturing intrinsic and bias attributes on C-MNIST. V-Score represents homogeneity and completeness, with higher values indicating better clustering. In Fig. 4, our DGW's ϕ^i captures intrinsic attributes effectively, resulting in tighter clusters and better separation, as indicated by the V-Score. The bias attributes are well captured by ϕ^b, as shown in Fig. 4(d).

Model Similarity. We visualize model similarity using Centered Kernel Alignment (CKA) (Raghu et al., 2021; Kornblith et al., 2019; Cortes et al., 2012), comparing similarities between all pairs of layers for different models. In this analysis, I and B denote ϕ^i and ϕ^b. As shown in Fig. 5, Vanilla and LFA have similar weights across many layers, whereas DGW shows fewer similarities in both initial and deeper layers, indicating different behavior across layers compared to baselines.



(a) LFA (Intrinsic) (b) LFA (Bias) (c) DGW (Intrinsic) (d) Vanilla vs DGW (Bias)

Figure 5: Representations of similarities for vanilla and different methods with all pairs of layers on C-CIFAR-10 (0.5% setting). High similarity score denotes high values.

Table 2: ECE (%) and NLL under different settings on C-CIFAR-10.

Ratio (%):	0.5		1.0		2.0		5.0	
Model	ECE	NLL	ECE	NLL	ECE	NLL	ECE	NLL
Vanilla	13.75	5.99	13.14	9.87	12.25	6.65	13.76	5.99
LFA	12.09	5.81	**11.45**	7.27	10.25	5.14	7.56	3.09
DGW (Ours)	**11.85**	**5.71**	11.53	**6.88**	**9.96**	**4.41**	**7.55**	**3.01**

Model Reliability. We evaluate model generalizability using Expected Calibration Error (ECE) and Negative Log Likelihood (NLL) (Guo et al., 2017). ECE measures calibration error, and NLL assesses probabilistic quality. As shown in Table 2, DGW consistently has the lowest ECE and NLL, indicating better generalizability compared to baselines.

5 Conclusion

In this work, we introduced Debiasing Global Workspace (DGW), a framework designed to learn debiased representations of attributes in neural networks. By leveraging attention mechanisms inspired by the Global Workspace Theory, our method effectively differentiates between intrinsic and biased attributes, enhancing both performance and interpretability. Comprehensive evaluations across various biased datasets demonstrated that DGW improves model robustness and generalizability on biased data and provides interpretable insights into the model's decision-making process. Our approach results in tighter clusters and better model separation, indicating superior performance in both intra- and inter-classification tasks. Furthermore, DGW shows improved model reliability and generalizability, making it a better solution to address biases in real-world applications. Future work could focus on reducing this complexity, exploring the scalability of DGW to even larger and more diverse datasets, and extending the framework into a general-purpose drop-in layer to enhance robust performance across a wider range of image recognition tasks.

Limitations. We acknowledge that the introduction of our modules can increase the complexity of the training, including the size of the model and the training time. This represents a trade-off between performance and transparency in decision making. Although our additional overhead is minimal, further analysis is necessary to optimize and streamline the process.

Acknowledgments and Disclosure of Funding

This work was supported by NSF EFMA-2223839.

References

Agarwal, V., Shetty, R., Fritz, M.: Towards causal vqa: Revealing and reducing spurious correlations by invariant and covariant semantic editing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9690–9698 (2020)

Baars, B.J.: A cognitive theory of consciousness. Cambridge University Press (1993)

Baars, B.J.: Global workspace theory of consciousness: toward a cognitive neuroscience of human experience. Progress in brain research **150**, 45–53 (2005)

Bahng, H., Chun, S., Yun, S., Choo, J., Oh, S.J.: Learning de-biased representations with biased representations. In: Proceedings of the International Conference on Machine Learning, pp. 528–539, PMLR (2020)

Bengio, Y.: The consciousness prior. CoRR **abs/1709.08568** (2017), URL `http://arxiv.org/abs/1709.08568`

Burgess, C.P., Matthey, L., Watters, N., Kabra, R., Higgins, I., Botvinick, M., Lerchner, A.: Monet: Unsupervised scene decomposition and representation. arXiv preprint arXiv:1901.11390 (2019)

Chang, M., Griffiths, T., Levine, S.: Object representations as fixed points: Training iterative refinement algorithms with implicit differentiation. Advances in Neural Information Processing Systems **35**, 32694–32708 (2022)

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder–decoder for statistical machine translation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1724–1734 (2014)

Cortes, C., Mohri, M., Rostamizadeh, A.: Algorithms for learning kernels based on centered alignment. The Journal of Machine Learning Research **13**(1), 795–828 (2012)

Darlow, L., Jastrzębski, S., Storkey, A.: Latent adversarial debiasing: Mitigating collider bias in deep neural networks. arXiv preprint arXiv:2011.11486 (2020)

Dehaene, S., Changeux, J.P.: Experimental and theoretical approaches to conscious processing. Neuron **70**(2), 200–227 (2011)

Didolkar, A.R., Goyal, A., Bengio, Y.: Cycle consistency driven object discovery. In: The Twelfth International Conference on Learning Representations (2023)

Fodor, J.A., Pylyshyn, Z.W.: Connectionism and cognitive architecture: A critical analysis. Cognition **28**(1-2), 3–71 (1988)

Geirhos, R., Jacobsen, J.H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., Wichmann, F.A.: Shortcut learning in deep neural networks. Nature Machine Intelligence **2**(11), 665–673 (2020)

Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F.A., Brendel, W.: Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. In: Proceedings of the International Conference on Learning Representations (2018)

Goel, K., Gu, A., Li, Y., Re, C.: Model patching: Closing the subgroup performance gap with data augmentation. In: Proceedings of the International Conference on Learning Representations (2020)

Goyal, A., Bengio, Y.: Inductive biases for deep learning of higher-level cognition. Proceedings of the Royal Society A **478**(2266), 20210068 (2022)

Goyal, A., Didolkar, A.R., Lamb, A., Badola, K., Ke, N.R., Rahaman, N., Binas, J., Blundell, C., Mozer, M.C., Bengio, Y.: Coordination among neural modules through a shared global workspace. In: Proceedings of the International Conference on Learning Representations (2021)

Greff, K., Kaufman, R.L., Kabra, R., Watters, N., Burgess, C., Zoran, D., Matthey, L., Botvinick, M., Lerchner, A.: Multi-object representation learning with iterative variational inference. In: International conference on machine learning, pp. 2424–2433, PMLR (2019)

Greff, K., Van Steenkiste, S., Schmidhuber, J.: On the binding problem in artificial neural networks. arXiv preprint arXiv:2012.05208 (2020)

Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q.: On calibration of modern neural networks. In: Proceedings of the International Conference on Machine Learning (ICML), pp. 1321–1330 (2017)

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

He, T., Zhang, Z., Zhang, H., Zhang, Z., Xie, J., Li, M.: Bag of tricks for image classification with convolutional neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 558–567 (2019)

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

Hendrycks, D., Dietterich, T.: Benchmarking neural network robustness to common corruptions and perturbations. In: International Conference on Learning Representations (2018)

Hendrycks, D., Zhao, K., Basart, S., Steinhardt, J., Song, D.: Natural adversarial examples. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 15262–15271 (2021b)

Hong, J., Park, K.H., Pavlic, T.P.: Concept-centric transformers: Enhancing model interpretability through object-centric concept learning within a shared global workspace. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp. 4880–4891 (January 2024)

Hong, Y., Yang, E.: Unbiased classification through bias-contrastive and bias-balanced learning. Advances in Neural Information Processing Systems **34**, 26449–26461 (2021)

Hwang, I., Lee, S., Kwak, Y., Oh, S.J., Teney, D., Kim, J.H., Zhang, B.T.: Selecmix: Debiased learning by contradicting-pair sampling. Advances in Neural Information Processing Systems **35**, 14345–14357 (2022)

Jia, B., Liu, Y., Huang, S.: Improving object-centric learning with query optimization. In: Proceedings of the International Conference on Learning Representations (2022)

Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 4401–4410 (2019)

Kataoka, Y., Matsubara, T., Uehara, K.: Image generation using generative adversarial networks and attention mechanism. In: Proceedings of the IEEE/ACIS International Conference on Computer and Information Science (ICIS), pp. 1–6, IEEE (2016)

Kim, B., Kim, H., Kim, K., Kim, S., Kim, J.: Learning not to learn: Training deep neural networks with biased data. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 9012–9020 (2019)

Kim, E., Lee, J., Choo, J.: Biaswap: Removing dataset bias with bias-tailored swapping augmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 14992–15001 (2021)

Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)

Kornblith, S., Norouzi, M., Lee, H., Hinton, G.: Similarity of neural network representations revisited. In: Proceedings of the International Conference on Machine Learning, pp. 3519–3529, PMLR (2019)

Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. Tech. Rep. TR-2009, University of Toronto, Toronto, Ontario (2009)

Lee, J., Kim, E., Lee, J., Lee, J., Choo, J.: Learning debiased representation via disentangled feature augmentation. Advances in Neural Information Processing Systems **34**, 25123–25133 (2021)

Li, Y., Vasconcelos, N.: Repair: Removing representation bias by dataset resampling. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 9572–9581 (2019)

Li, Y., Yu, Q., Tan, M., Mei, J., Tang, P., Shen, W., Yuille, A., Xie, C.: Shape-texture debiased neural network training. arXiv preprint arXiv:2010.05981 (2020)

Lim, J., Kim, Y., Kim, B., Ahn, C., Shin, J., Yang, E., Han, S.: Biasadv: Bias-adversarial augmentation for model debiasing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3832–3841 (2023)

Liu, E.Z., Haghgoo, B., Chen, A.S., Raghunathan, A., Koh, P.W., Sagawa, S., Liang, P., Finn, C.: Just train twice: Improving group robustness without training group information. In: International Conference on Machine Learning, pp. 6781–6792, PMLR (2021)

Locatello, F., Weissenborn, D., Unterthiner, T., Mahendran, A., Heigold, G., Uszkoreit, J., Dosovitskiy, A., Kipf, T.: Object-centric learning with slot attention. Advances in Neural Information Processing Systems **33**, 11525–11538 (2020)

Luo, P., Wang, G., Lin, L., Wang, X.: Deep dual learning for semantic image segmentation. In: Proceedings of the IEEE international conference on computer vision, pp. 2718–2726 (2017)

van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of Machine Learning Research **9**(86), 2579–2605 (2008)

Mashour, G.A., Roelfsema, P., Changeux, J.P., Dehaene, S.: Conscious processing and the global neuronal workspace hypothesis. Neuron **105**(5), 776–798 (2020)

Minderer, M., Bachem, O., Houlsby, N., Tschannen, M.: Automatic shortcut removal for self-supervised representation learning. In: Proceedings of the International Conference on Machine Learning, pp. 6927–6937, PMLR (2020)

Minsky, M.: Society of mind. Simon and Schuster (1988)

Nam, J., Cha, H., Ahn, S., Lee, J., Shin, J.: Learning from failure: De-biasing classifier from biased classifier. Advances in Neural Information Processing Systems **33**, 20673–20684 (2020)

Posner, M.I.: Attention: the mechanisms of consciousness. Proceedings of the National Academy of Sciences **91**(16), 7398–7403 (1994)

Raghu, M., Unterthiner, T., Kornblith, S., Zhang, C., Dosovitskiy, A.: Do vision transformers see like convolutional neural networks? Advances in Neural Information Processing Systems **34**, 12116–12128 (2021)

Robbins, P.: Modularity of mind. In: Zalta, E.N. (ed.) The Stanford Encyclopedia of Philosophy, Metaphysics Research Lab, Stanford University, Winter 2017 edn. (2017), URL `https://plato.stanford.edu/archives/win2017/entries/modularity-mind/`

Rosenberg, A., Hirschberg, J.: V-measure: A conditional entropy-based external cluster evaluation measure. In: Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 410–420 (2007)

Sagawa, S., Koh, P.W., Hashimoto, T.B., Liang, P.: Distributionally robust neural networks. In: International Conference on Learning Representations (2019)

Sanh, V., Wolf, T., Belinkov, Y., Rush, A.M.: Learning from others' mistakes: Avoiding dataset biases without modeling them. In: International Conference on Learning Representations (2020)

Seth, A.K., Bayne, T.: Theories of consciousness. Nature Reviews Neuroscience **23**(7), 439–452 (2022)

Tartaglione, E., Barbano, C.A., Grangetto, M.: End: Entangling and disentangling deep representations for bias correction. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 13508–13517 (2021)

Torralba, A., Efros, A.A.: Unbiased look at dataset bias. In: CVPR 2011, pp. 1521–1528, IEEE (2011)

Tu, B., Zhou, C., Kuang, W., Chen, S., Plaza, A.: Multiattribute sample learning for hyperspectral image classification using hierarchical peak attribute propagation. IEEE Transactions on Instrumentation and Measurement **71**, 1–17 (2022)

VanRullen, R., Kanai, R.: Deep learning and the global workspace theory. Trends in Neurosciences **44**(9), 692–704 (2021)

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in Neural Information Processing Systems **30** (2017)

Verma, V., Lamb, A., Beckham, C., Najafi, A., Mitliagkas, I., Lopez-Paz, D., Bengio, Y.: Manifold mixup: Better representations by interpolating hidden states. In: International conference on machine learning, pp. 6438–6447, PMLR (2019)

Wang, H., He, Z., Lipton, Z.C., Xing, E.P.: Learning robust representations by projecting superficial statistics out. In: Proceedings of the International Conference on Learning Representations (2018)

Wang, X., Gupta, A.: Generative image modeling using style and structure adversarial networks. In: Proceedings of the European Conference on Computer Vision, pp. 318–335, Springer (2016)

Wang, Z., Qinami, K., Karakozis, I.C., Genova, K., Nair, P., Hata, K., Russakovsky, O.: Towards fairness in visual recognition: Effective strategies for bias mitigation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 8919–8928 (2020)

Xie, Q., Luong, M.T., Hovy, E., Le, Q.V.: Self-training with noisy student improves imagenet classification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10687–10698 (2020)

Zhang, Y.K., Wang, Q.W., Zhan, D.C., Ye, H.J.: Learning debiased representations via conditional attribute interpolation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7599–7608 (2023)

Zhang, Z., Sabuncu, M.: Generalized cross entropy loss for training deep neural networks with noisy labels. Advances in neural information processing systems **31** (2018)

Zhao, J., Peng, Y., He, X.: Attribute hierarchy based multi-task learning for fine-grained image classification. Neurocomputing **395**, 150–159 (2020)

Zheng, S., Cheng, M.M., Warrell, J., Sturgess, P., Vineet, V., Rother, C., Torr, P.H.: Dense semantic image segmentation with objects and attributes. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3214–3221 (2014)