Unsupervised Domain Adaptation/Domain Generalization

Muhammad Saad Haroon Jawad Saeed Daanish Uddin Khan

GitHub Repo Link

Abstract

This report delves into three pivotal areas of machine learning: Unsupervised Domain Adaptation (UDA), Domain Generalization (DG), and Disentangled Representation Learning using β-Variational Autoencoders (β-VAE). For UDA, we evaluate the performance of ResNet-50, Domain Adaptation Networks (DAN), and Domain-Adversarial Neural Networks (DANN) across diverse datasets to understand domain alignment techniques. In DG, we explore Invariant Risk Minimization (IRM) and its variants (IRM Basic, IB-IRM, and PAIR) to assess their efficacy in enhancing generalization under out-of-distribution shifts. Finally, in the study of β-VAE, we investigate the interplay between disentanglement and reconstruction quality by experimenting with varying β values and analyzing both qualitative visualizations and quantitative metrics such as Z-Diff and MIG. Through these tasks, we highlight key insights, discuss the inherent trade-offs, and identify limitations, providing actionable recommendations for advancing machine learning methodologies.

1. Introduction

Machine learning models often face challenges when applied to unseen distributions or require interpretable representations. This report addresses these challenges through three tasks:

- **Task 1: Unsupervised Domain Adaptation (UDA)** tackles the problem of transferring knowledge between source and target domains without labeled data for the target domain. We analyze baseline ResNet-50, DAN, and DANN to evaluate domain alignment techniques.
- **Task 2: Domain Generalization (DG)** aims to improve model generalization under out-of-distribution (OOD) scenarios. IRM variants (IRM Basic, IB-IRM, PAIR) are explored, focusing on the trade-off between in-distribution and OOD performance.
- **Task 3: Disentangled Representation Learning** investigates β-VAE, an extension of the Variational Autoencoder, to learn disentangled latent representations. The study examines the trade-off between disentanglement and reconstruction quality by varying β values and analyzing metrics such as Z-Diff and MIG.

Each task presents unique challenges and opportunities for understanding and improving machine learning models. The following sections detail the methodologies, results, and discussions for each task.

2. Methodology

2.1. Task 1 - Unsupervised Domain Adaptation

2.1.1. Dataset Preparation

For this task, we made use of the **Digits Dataset** and the **Office-31** dataset for evaluation. Their breakdown is as follows:

- **Digits Dataset** is referred to as a combination of the **MNIST, USPS and SVHN** datasets containing 10 different classes.
- **Office-31** dataset consists of three distinct domains namely **Amazon(A), DLSR(D) and Webcam(W)** containing 31 different classes.

The **Office-31** dataset preprocessing involved separate transforms for the source and testing dataloaders. The source loaders were subject to data augmentation is the form **RandomResizedCrop** and **RandomHorizontalFlip** to improve the model robustness during the adaption phase. The target dataset on the other hand did not experience any such data augmentation transformations. Both the source and target datasets were normalized using the **ImageNet** parameters to ensure optimized training and evaluation.

The **Digits Datset** had similar preprocessing transforms applied across the datasets with the exception of the addition of color channels in the **MNIST dataset** to ensure normalized tensors. Additionally, all the datasets' images were resized to **224x224** and normlaized using **ImageNet** parameters.

2.1.2. Model Configuration

This task involved the evaluation of three different model types. Their description alongside configurations are listed as follows:

- **Baseline Pre-Trained ResNet**: The baseline model is a modified **ResNet50** model using the latest ImageNet weights from the **torchvision** library. The classification layer of the model is modified according to the classes in the dataset being used.
- **Domain Adaptation Network (DAN)**: The Domain Adaptation Network (DAN) leverages ResNet-50 as a feature extractor to learn transferable features across domains. It includes a bottleneck layer for dimensionality reduction and a classifier for task-specific predictions. To align source and target distributions, the model employs Maximum Mean Discrepancy (MMD) as the transfer loss, promoting domain-invariant feature learning.
- **Domain Adversarial Neural Networks (DANN)**: The Domain-Adversarial Neural Network (DANN) architecture is designed to align feature distributions between source and target domains. The Extractor leverages a ResNet-50 backbone to extract domain-independent features from input images. A Classifier predicts task-specific classes based on these features, while a Domain Classifier employs a gradient reversal layer to adversarially learn domain-invariant features by minimizing domain classification accuracy. The network utilizes a learning rate scheduler for stable optimization, encouraging effective domain adaptation.

2.1.3. Task Setup

The task setup for each of the three models is describes as follows:

- **Baseline Pre-Trained ResNet**: For the baseline task setup, a ResNet-based BaseModel is fine-tuned to classify images as the source and target domains vary across the **Office-31 and Digits Datasets**. The model uses a CrossEntropyLoss as the criterion, and a Stochastic Gradient Descent (SGD) optimizer with differential learning rates is applied to adjust backbone and fully connected layers separately. The Amazon dataset is used for training, while the Webcam dataset evaluates domain generalization. This baseline provides a benchmark for performance without explicit domain adaptation. After training and evaluation, a t-SNE visualization is performed to analyze the alignment of feature representations between the source and target domains. A FeatureExtractor is used to extract high-dimensional features from the fine-tuned model for both datasets. These features are then reduced to two dimensions using t-SNE and plotted to visually assess domain overlap. The alignment of feature clusters provides insights into the effectiveness of domain adaptation.
- **Domain Adaptation Network (DAN)**: For the DAN task setup, the model uses MMD (Maximum Mean Discrepancy) as the transfer loss to align the feature distributions of the source (Amazon) and target (Webcam) domains. The model architecture is based on ResNet-50, enhanced with a bottleneck layer and a classifier layer for domain adaptation. The optimizer is SGD with a learning rate of 0.0001 for the base network and 10x for the bottleneck and classifier layers. The model is trained for 5 epochs, with an early stopping criterion of 3 epochs, using a weighted loss that combines cross-entropy and transfer loss with a weight parameter $\lambda = 0.5$. Similar to the baseline model, a t-SNE visualization is used to analyze the alignment of the feature representations.
- **Domain Adversarial Neural Networks (DANN)**: For the Domain-Adversarial Neural Network (DANN) task setup, the model uses adversarial training to learn domain-invariant features. It consists of three components:
  - **Feature Extractor (Encoder)**: Extracts features from input images using a pre-trained ResNet-50 backbone.
  - **Classifier**: Predicts class labels for the source domain (Amazon), trained with cross-entropy loss.
  - **Domain Discriminator**: Predicts domain labels (source or target) using a reversed gradient from the feature extractor, promoting domain confusion.

  The dataloaders are created for the source and target datasets, with batch sizes of 16. The model is trained for 5 epochs using a domain adversarial approach where the classifier minimizes classification loss, and the discriminator maximizes domain confusion. This adversarial objective aligns feature distributions across domains, enabling better generalization to the target domain.

2.2. Task 2 - DG with IRM Variants

The methodology follows a structured approach to evaluate and analyze the performance of three methods of invariant risk minimization: **IRM Basic**, **IB-IRM**, and **PAIR**. This process was conducted in two distinct parts to systematically investigate out-of-distribution (OOD) generalization and the trade-off between in-distribution (IID) and OOD accuracy.

Dataset Preparation: The MNIST dataset was transformed into rotated versions with training rotations of *15°*, *30°*, *45°*, *60°*, and *75°*. A held-out test set with no rotation (*0°*) was used for evaluating out-of-distribution (OOD) performance. For each rotation, a subset comprising 25% of the original dataset was used (Due to computational limitations). Batch size for training and evaluation was set to *64*.

Part 1: Evaluating OOD Generalization

In this phase, we implemented IRM Basic, IB-IRM, and PAIR to evaluate their ability to generalize across unseen domains. Models were trained on source domains comprising rotated MNIST data (*15°*, *30°*, *45°*, *60°*, *75°*) and evaluated on a target domain (*0°*, representing OOD). Each model was trained for 3 epochs with a learning rate of 0.001. For each method:

- *IRM Basic:* Combines cross-entropy loss with a gradient norm penalty.

- *IB-IRM:* Extends IRM Basic by adding a variance regularization term to enforce consistent feature representations.

- *PAIR:* Combines ERM, IRM penalty, and variance regularization with weighted contributions.

During training, the focus was on understanding the trade-offs introduced by regularization in IB-IRM and PAIR compared to IRM Basic. OOD accuracy was tracked for all methods to identify which achieved the highest generalization performance.

Part 2: Investigating IID vs. OOD Accuracy Trade-off

In this phase, we analyzed the trade-off between IID and OOD accuracy for each method. IID accuracy was measured on the training domains, while OOD accuracy was evaluated on the unseen target domain (*0°*). Models were trained across 3 epochs, tracking both IID and OOD accuracy over training steps.

To visualize this trade-off, we plotted *OOD accuracy (y-axis)* against *IID accuracy (x-axis)* for each method during training. The goal was to identify trends, such as whether improving IID accuracy degraded OOD accuracy and which method achieved the best balance between the two.

Additional Hyperparameter Tuning (Supporting Analysis)

While not the primary focus, additional hyperparameter tuning was conducted to enhance the methods' performance:

- For IRM Basic, the penalty weight (λ_{penalty}) was varied across $[0.1, 0.5, 1.0]$, with $\lambda_{\text{penalty}} = 0.5$ yielding the best OOD accuracy.

- Using the optimal λ_{penalty}, IB-IRM was further tuned by varying the variance weight (γ_{variance}) across $[0.1, 0.01, 0.001]$, resulting in $\gamma_{\text{variance}} = 0.001$ as optimal.

- PAIR utilized these hyperparameters and tested weighted contributions for ERM, IRM penalty, and variance regularization, with $w_1 = 0.1$, $w_2 = 1.0$, and $w_3 = 0.01$ producing the most balanced results.

Evaluation Metrics: Training loss, IID accuracy, and OOD accuracy were logged during training for all methods. The results were visualized to highlight trends, allowing for a clear comparative analysis of the methods' performance across the two parts.

This methodology ensures a thorough evaluation of IRM methods, focusing on their ability to achieve OOD generalization and balance IID and OOD performance, while also providing insights into the role of regularization.

2.3. Task 3 - Disentangled Representation Learning

2.3.1. 3.1 Overview of β-VAE

The β-VAE is an extension of the Variational Autoencoder (VAE) that introduces a scaling factor β to the KL divergence term in the objective function. This modification encourages disentanglement in the learned latent representations, at the cost of reconstruction quality. By varying β, the trade-off between disentanglement and reconstruction can be studied, providing insights into the model's behavior under different configurations.

2.3.2. 3.2 Dataset Preprocessing

The CelebA dataset was used for training and evaluation. The preprocessing steps included:

- **Dataset Filtering:** The dataset was reduced by 50% to manage computational overhead. A new directory was created to store the reduced dataset, and corresponding ground truth factors and attributes were filtered accordingly.
- **Image Processing:** Images were resized to 64×64 pixels and normalized to the range $[-1, 1]$.
- **Data Splitting:** The dataset was split into training (80%), validation (10%), and test (10%) sets.
- **Ground Truth Saving:** The ground truth factors of variation were saved in a structured format alongside the latent vectors for reproducibility.

2.3.3. 3.3 Training β-VAE

The β-VAE was trained using the following procedure:

- **Architecture:** The encoder-decoder architecture consisted of convolutional and transposed convolutional

layers, with a latent dimension of 10. A latent dimension of 10 was selected as a balance between computational feasibility and effective disentanglement. CelebA's structure, with 40 binary attributes and 5 landmark locations, was well-represented within this latent space, ensuring key generative factors like pose and background clutter were captured without overfitting.

- **Loss Function:** The loss comprised the Mean Squared Error (MSE) reconstruction loss and the scaled KL divergence term, controlled by β.

$$\mathcal{L}_{\beta\text{-VAE}} = \mathbb{E}_{q_\phi}[-\log p_\theta(x|z)] + \beta \cdot \mathrm{KL}(q_\phi(z|x)\|p(z)) \tag{1}$$

- **Hyperparameters:** β values of 1, 5, 10, and 50 were tested. The Adam optimizer was used with a learning rate of 1×10^{-4}, and models were trained for 3 epochs with a batch size of 128.
- **Model Saving:** Separate models were saved for each β value for evaluation.

2.3.4. 3.4 Quantitative Evaluation

To evaluate disentanglement quantitatively, the following metrics were computed:

- **Z-Diff:** Measures the difference in disentanglement for different latent dimensions.
- **MIG (Mutual Information Gap):** Evaluates the mutual information between latent dimensions and ground truth factors.
- **BetaVAE Metric:** Measures disentanglement based on the accuracy of predicting the ground truth factors from latent vectors.
- **FactorVAE Metric:** Assesses disentanglement by evaluating the variance captured by each latent dimension.

The results were tabulated to compare the performance of β-VAE models with different β values and the baseline VAE.

2.3.5. 3.5 Qualitative Evaluation

The disentangled latent factors were visualized to provide qualitative insights:

- **Visualization Process:** For each model, one latent dimension was varied across multiple ranges such as $[-1.5, 1.5]$, $[-2, 2]$, $[-3, 3]$, $[-4, 4]$, and $[-5, 5]$, while keeping others fixed. These experiments aimed to evaluate the range's effect on the interpretability of variations in generated images.
- **Generated Images:** For each range, 10 samples were generated to illustrate the variations within the specified latent dimension. The number of samples acts as a hyperparameter, with its variation providing a finer or broader view of the impact of latent changes. The generated images demonstrated the relationship between latent dimensions and factors such as azimuth, style, and intensity.
- **Observations:** Increasing β improved disentanglement but led to a noticeable loss of fine-grained details in reconstructions. Furthermore, broader ranges such as $[-5, 5]$ showed exaggerated variations, while narrower ranges like $[-1.5, 1.5]$ provided subtler changes, highlighting a balance between interpretability and diversity.

2.3.6. 3.6 Additional Experiments

To further analyze model behavior, additional experiments were conducted:

- **Latent Dimension Variations:** The latent dimension was varied (e.g., 5, 10, 20) to study its impact on disentanglement and reconstruction.
- **Reconstruction and KL Trends:** The trends of reconstruction loss and KL divergence were analyzed across different β values.

These experiments highlighted the trade-offs and emphasized the role of hyperparameter tuning.

2.3.7. 3.7 Implementation Challenges

Several challenges were encountered during implementation:

- **Dataset Filtering:** Handling mismatched image IDs in the CelebA dataset during preprocessing.
- **Disentanglement Metrics:** Installing and using the DisentanglementLib library presented compatibility issues, leading to manual metric computation.
- **Model Convergence:** Ensuring convergence for higher β values required careful tuning of hyperparameters and loss scaling.

Method	A → W	A → D	W → A	W → D	D → A	D → W	Avg.
ResNet-50	37.89%	40.85%	10.16%	12.95%	12.15%	12.24%	21.04%
DAN	10.94%	7.81%	9.09%	9.37%	8.84%	8.72%	9.13%
DANN	7.30%	9.84%	16.11%	6.89%	8.74%	18.11%	11.16%

Table 1. Performance comparison of ResNet-50, DAN, and DANN across different domain adaptation tasks.

3. Results

3.1. Task 1 - Unsupervised Domain Adaptation

Table 1 shows the results of the three models across the **Office-31** dataset. From the results it is visible that the highest average accuracy is obtained on the base pre-trained model followed by the **DANN** and lastly the **DAN**. The performance of the models is discussed in detail in the discussion section.

Method	M → U	U → M	S → M	Avg.
ResNet-50	85%	84.04%	72.98%	81%
DAN	37.34%	37.06%	33.46%	35.95%
DANN	93.20%	19.00%	75.10%	62.43%

Table 2. Performance comparison of ResNet-50, DAN, and DANN across different domain adaptation tasks.

From the results for the digits dataset in Table 2 it is visible that the accuracies are much higher for the digits datasets across the board. Similar to the Office dataset the highest accuracy was observed on the pre-trained model followed by the **DANN** and then **DAN**.

3.2. Task 2 - DG with IRM Variants

This section presents the results for both Part 1 and Part 2 of the experiment. The goal is to report out-of-distribution (OOD) accuracy, in-distribution (IID) accuracy, and training loss for each method across all tested configurations. Results are summarized in compact tables and referenced alongside corresponding figures.

Part 1: Evaluating OOD Accuracy Across Configurations

In Part 1, the performance of IRM Basic, IB-IRM, and PAIR was evaluated by testing unique hyperparameter configurations for each method. The final training loss and OOD accuracy for all configurations are summarized in Tables 3, 4, and 5.

Figures 1, 2, and 3 illustrate OOD accuracy trends for the best configurations of each method.

Part 2: IID vs. OOD Accuracy Trade-off

In Part 2, IID and OOD accuracy trends were analyzed to investigate the trade-off between in-distribution and out-of-distribution performance. Table 6 summarizes the final results for each method, and Figure 7 visualizes IID vs. OOD accuracy trends.

Figures 4, 5, and 6 show IID and OOD accuracy trends for each method.

Table 3. IRM Basic Results Across λ_{penalty}.

Lambda	Final Train Loss	Final OOD Acc. (%)
1.0	117.03	14.47
0.5	85.69	19.49
0.1	35.88	17.59

Table 4. IB-IRM Results Across γ_{variance}.

Gamma	Final Train Loss	Final OOD Acc. (%)
0.1	299.28	7.50
0.01	316.17	18.20
0.001	106.34	37.45

3.3. Task 3 - Disentangled Representation Learning

The quantitative evaluation of β-VAE models with varying β values is presented in the following table:

The results in Table 7 illustrate the trade-offs between reconstruction quality and disentanglement as the β value increases. For $\beta = 1$, the model achieves the lowest total loss and reconstruction loss, indicating better fidelity to the input images. However, as β increases, the reconstruction loss grows, highlighting a loss of fine-grained details. Meanwhile, the KL divergence decreases significantly, indicating stronger regularization and increased disentanglement. These trends confirm the theoretical expectations of β-VAE, where higher β prioritizes disentanglement at the expense of reconstruction quality.

3.3.1. Visualization of Disentangled Latent Factors

The visualizations of disentangled latent factors are presented in Figures 8, 9, 10, 11, and 12. These illustrate the effect of varying the latent dimension ranges across $[-1.5, 1.5]$, $[-2, 2]$, $[-3, 3]$, $[-4, 4]$, and $[-5, 5]$, respectively.

As shown in the figures, varying the latent dimension ranges provides insights into the level of disentanglement achieved. For narrower ranges such as $[-1.5, 1.5]$, the variations are subtle, whereas broader ranges like $[-5, 5]$ exaggerate the generative factors, providing a more pronounced view of disentanglement. These visualizations complement the quantitative metrics and emphasize the trade-off between diversity and interpretability in generated images.

Table 5. PAIR Results Across Weight Configurations.

w_1	w_2	w_3	**Train Loss**	**OOD Acc. (%)**
1.0	0.1	0.01	17.09	18.62
1.0	0.01	0.1	2.26	11.36
0.1	1.0	0.01	91.14	28.62
0.1	0.01	1.0	14.10	14.23
0.01	1.0	0.1	267.16	23.14
0.01	0.1	1.0	24.85	6.14

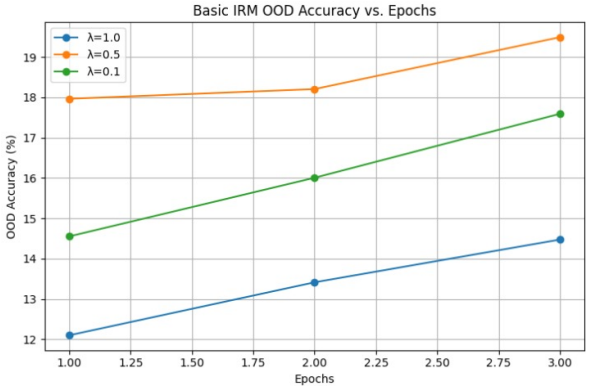

Figure 1. Basic IRM OOD Accuracy vs. Epochs.

3.3.2. Visualization of β Values for a Fixed Latent Range

The effect of varying β values on the disentanglement of latent factors is visualized in Figures 13, 14, 15, and 16, for $\beta = 1$, $\beta = 5$, $\beta = 10$, and $\beta = 50$, respectively. These visualizations were performed for a fixed latent range of $[-3, 3]$, allowing us to observe the impact of β on the disentanglement process.

From the figures, it is evident that as β increases, the disentanglement of generative factors improves. However, this comes at the cost of reconstruction quality. For $\beta = 1$, the reconstructions are closest to the input data but show weaker disentanglement. In contrast, $\beta = 50$ achieves strong disentanglement but loses fine-grained details, highlighting the trade-off inherent in β-VAE models.

3.3.3. Quantitative Evaluation of Disentanglement Metrics

The disentanglement metrics for β-VAE models with varying β values are summarized in Table 8. The table evaluates the models based on Z-Diff, MIG, BetaVAE Metric, FactorVAE Metric, and Reconstruction Quality.

Table 8 provides a detailed quantitative evaluation of the β-VAE models across various metrics. It highlights the interplay between β values and the model's performance on disentanglement (measured by Z-Diff, MIG, BetaVAE Metric, and FactorVAE Metric) versus reconstruction quality. As β increases, disentanglement metrics generally improve, but reconstruction loss worsens. This supports the theoretical understanding of β-VAE, where higher β enforces stronger disentanglement at the cost of reconstruction fidelity.

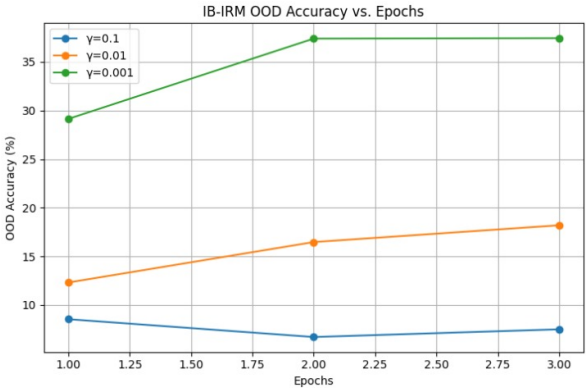

Figure 2. IB-IRM OOD Accuracy vs. Epochs.

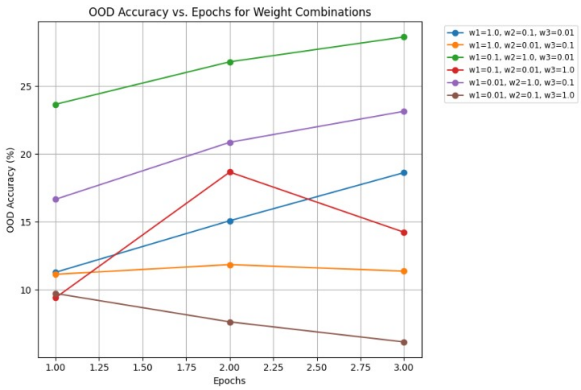

Figure 3. PAIR OOD Accuracy vs. Epochs.

4. Discussion

4.1. Task 1 - Unsupervised Domain Adaptation

4.1.1. Baseline Pre-Trained ResNet

Using the **Office-31** dataset for comparison across all the tasks we can see from Figure 17 that there is a clear separation of the source and target domain indicating the the model isn't performing well in the adaptation task. This can also be seen from the accuracy in this specific experimental setting which is **12.15%**.

4.1.2. Domain Adaptation Network (DAN)

Moving on to the Domain Adaptation network from Figure 18 that the model has slightly improved in terms of domain adaptation which is visible by the overlap in feature space, demonstrating that the learned representations are domain-

Table 6. Part 2: Final IID and OOD Accuracy.

Method	Train Loss	IID Acc. (%)	OOD Acc. (%)
IRM Basic	18.08	29.16	17.51
IB-IRM	141.70	41.07	35.57
PAIR	428.49	32.72	14.06

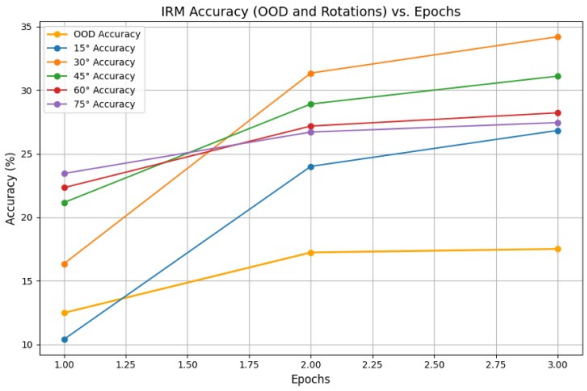

Figure 4. Basic IRM OOD/IID Accuracy vs. Epochs.

Beta	Avg Loss	KL Divergence	Reconstruction Loss
1	1154.06	32.0354	1122.02
5	1207.68	23.1259	1092.06
10	1296.4	19.113	1105.27
50	1796.68	9.62877	1315.24

Table 7. Loss metrics for β-VAE models with different β values.

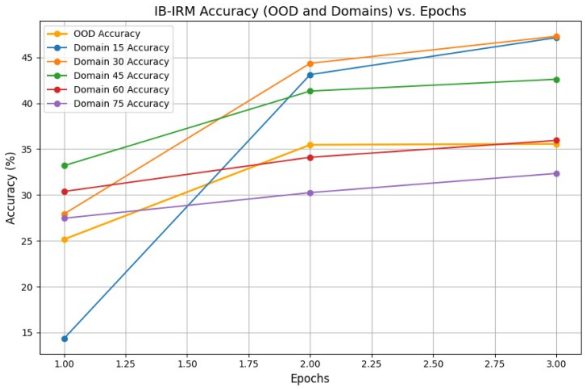

Figure 5. IB-IRM OOD/IID Accuracy vs. Epochs.

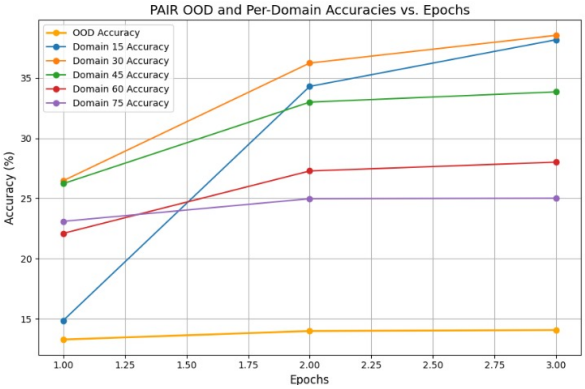

Figure 6. PAIR OOD/IID Accuracy vs. Epochs.

invariant. Additionally, distinct clusters corresponding to individual classes are observed, indicating that the model has learned discriminative features for classification. However the accuracy of the model is low as compared to base model which can be explained by a number of reasons:

- **Feature Representation Mismatch:**
  - The source and target domains may have significantly different feature distributions, making domain alignment challenging.
- **Insufficient Domain Adaptation:**
  - The MMD loss might not fully align the source and target distributions, especially in high-dimensional feature spaces. This can be seen by the fluctuation in the **MMD/Transfer Loss** in Figure 19 during the training epochs.
- **Class Overlap Issues:**
  - If some classes overlap in feature space, it can lead to poor discriminative performance in the target domain.
- **Imbalanced Training:**
  - Imbalance in the number of source and target samples during training can lead to biased feature learning towards the source domain. This is especially observable in this case due to imbalance in the dataloader sizes with Amazon being the largest followed by Webcam and finally DSLR.
- **Domain-Specific Features:**
  - The source domain's features might not generalize well to the target domain due to domain-specific characteristics. For example the Amazon pictures are mostly against a white background due to their use on the website while for the other two domains they have varying backgrounds depending on where the picture was taken.
- **Insufficient Training Epochs:**
  - The model might not have been trained for enough epochs to converge effectively for domain adaptation. Due to computational constraints in this case we only trained for 5 epochs which might

Model	Z-Diff	MIG	BetaVAE Metric	FactorVAE Metric	Reconstruction Quality (Loss)
VAE ($\beta = 1$)	0.510259	0.00173285	0.804603	0.804603	Low Reconstruction Loss
$\beta = 5$	0.383394	0.00122832	0.804344	0.804344	Medium Reconstruction Loss
$\beta = 10$	0.333523	0.00115836	0.804257	0.804257	High Reconstruction Loss
$\beta = 50$	0.542569	0.00155038	0.804442	0.804442	Very High Reconstruction Loss

Table 8. Comparison of disentanglement metrics and reconstruction quality for different β values. The table evaluates the trade-offs between improving disentanglement and maintaining reconstruction quality.

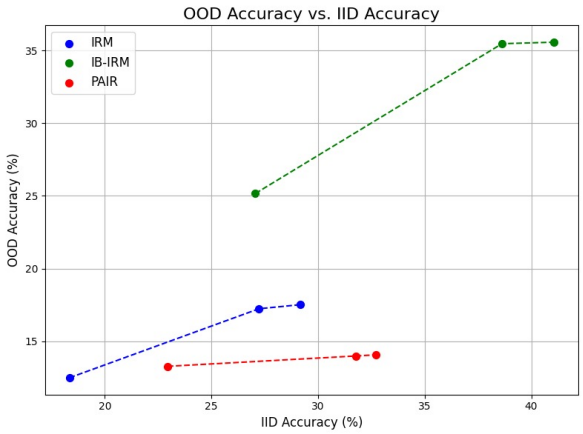

Figure 7. IID vs. OOD Accuracy for All Methods.

Figure 8. Disentangled Latent Factors for range $[-1.5, 1.5]$. The variations are subtle, indicating a narrow range of generative factors.

be less in terms of achieving good adaptation and learning domain invariant features.

Figure 9. Disentangled Latent Factors for range $[-2, 2]$. The variations become slightly more pronounced, enhancing interpretability.

4.1.3. Potential Improvements DAN

Based on the results here are a few potential improvements that could result in better domain adaptation:

- **Enhanced Domain Alignment with Adversarial Learning:**
  - Incorporate an adversarial domain discriminator, as used in Domain-Adversarial Neural Networks (DANN), to complement the MMD loss. This discriminator learns to distinguish between source and target domains, encouraging the feature extractor to generate domain-invariant features more effectively.
  - **Reasoning:** While MMD loss aligns distributions globally, adversarial learning can refine domain alignment at a finer granularity, potentially improving the overlap in feature spaces.
- **Dynamic Weighting of Transfer and Classification**

Figure 10. Disentangled Latent Factors for range $[-3, 3]$. The generated images show a balance between diversity and interpretability.

Figure 12. Disentangled Latent Factors for range $[-5, 5]$. The images exhibit significant variations, highlighting the trade-off between diversity and subtlety.

Figure 11. Disentangled Latent Factors for range $[-4, 4]$. The broader range exaggerates variations in generative factors.

Figure 13. Disentangled Latent Factors for $\beta = 1$. Subtle variations in generative factors are observed, with better reconstruction quality but weaker disentanglement.

Losses:

– Implement a dynamic weighting strategy that balances the trade-off between the classification loss and the MMD/transfer loss during training. For example, gradually increase the weight of the transfer loss as training progresses to ensure effec-

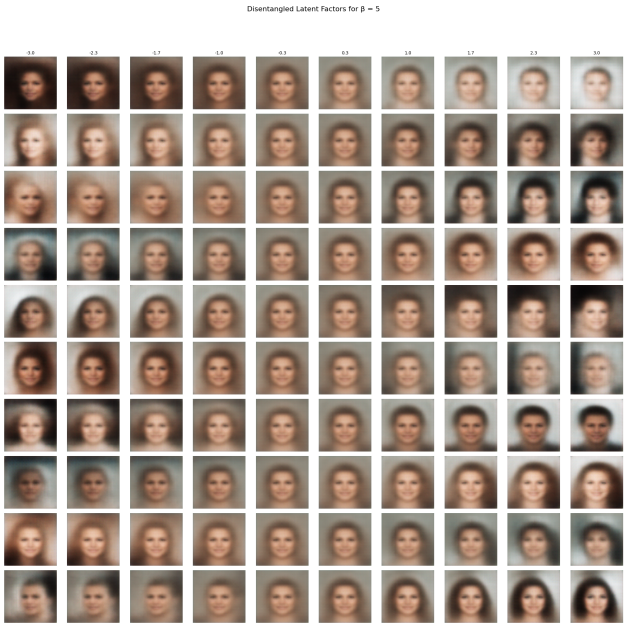

Figure 14. Disentangled Latent Factors for $\beta = 5$. Improved disentanglement is visible, with slight degradation in reconstruction quality.

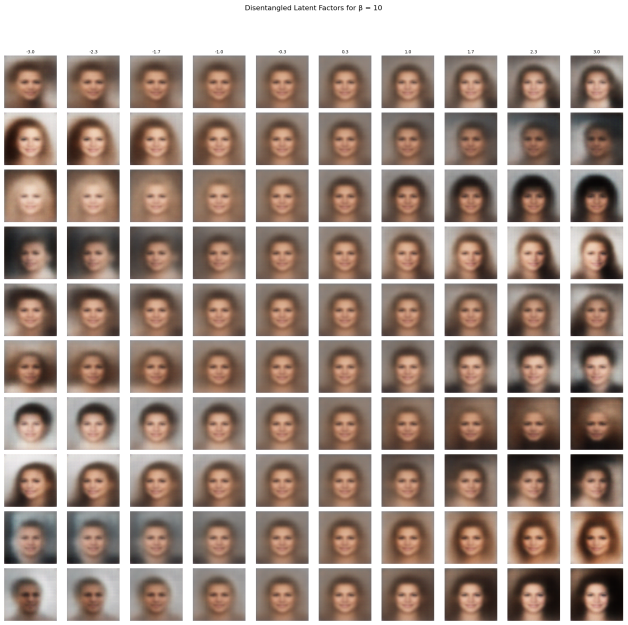

Figure 15. Disentangled Latent Factors for $\beta = 10$. Disentanglement becomes more prominent, but some fine-grained details are lost.

tive domain alignment after achieving good class separation.

– **Reasoning:** Fixed weights for the losses might lead to suboptimal learning. Dynamic weighting can adapt to the needs of the training process, focusing more on domain adaptation after sufficient classification performance.

Figure 16. Disentangled Latent Factors for $\beta = 50$. The generative factors are highly disentangled, but reconstruction quality is significantly degraded.

- **Augmenting Target Data with Pseudo-Labels:**
  - Use pseudo-labeling on the target domain. Once the model achieves reasonable confidence on certain target samples, pseudo-label these and include them in the training process, possibly with a consistency regularization strategy.
  - **Reasoning:** Pseudo-labeling encourages the model to adapt better to the target domain by utilizing unlabeled target data, which is particularly useful when the labeled source and target datasets are imbalanced.

4.1.4. Domain Adversarial Neural Network (DANN)

Concluding with the DANN model we can see from Figure 20 that the source accuracy climbs rapidly as compared to the target accuracy which makes sense due to adversarial and adaptive nature of the task. Despite this the target accuracy does show to be increasing across epochs indicating the the model is indeed adapting to the target domain. However, this accuracy is quite low as compared to the Base model which can be explained through the following potential reasons:

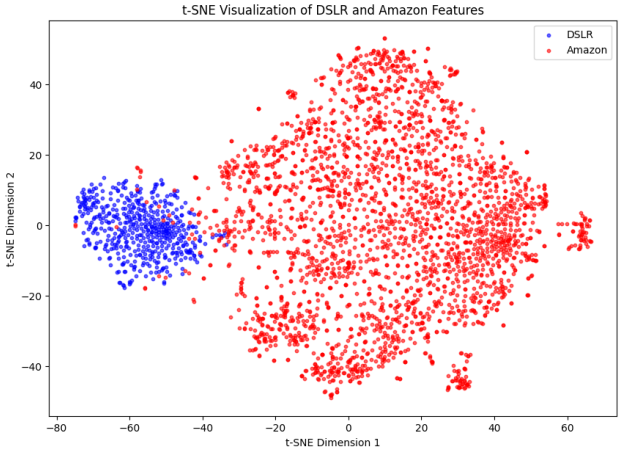

Figure 17. tSNE Visualization for DSLR to Amazon Adaptation

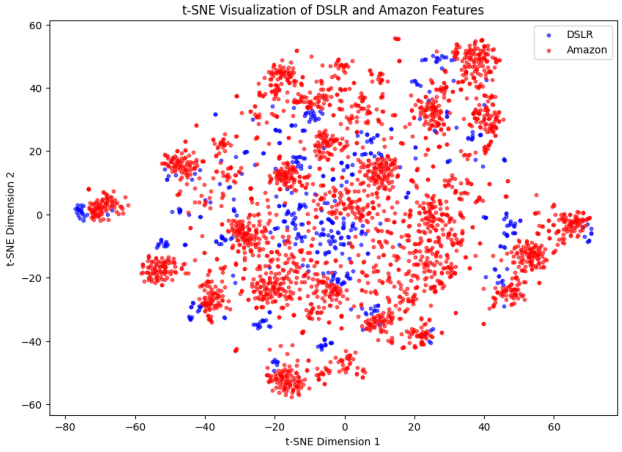

Figure 18. tSNE Visualization for DSLR to Amazon Adaptation

- **Gradient Instability:**
  - The adversarial training process (Reverse Gradient Layer) can cause unstable gradients, leading to suboptimal feature alignment.
- **Imperfect Domain Discriminator:**
  - The domain classifier may fail to effectively distinguish between source and target domain features, leading to poor domain-invariant feature learning.
- **Training Dynamics:**
  - The adversarial balance between the feature extractor and the domain classifier might not be optimal, leading to overfitting to the source domain or under-adaptation to the target domain.
- **Insufficient Feature Representation:**

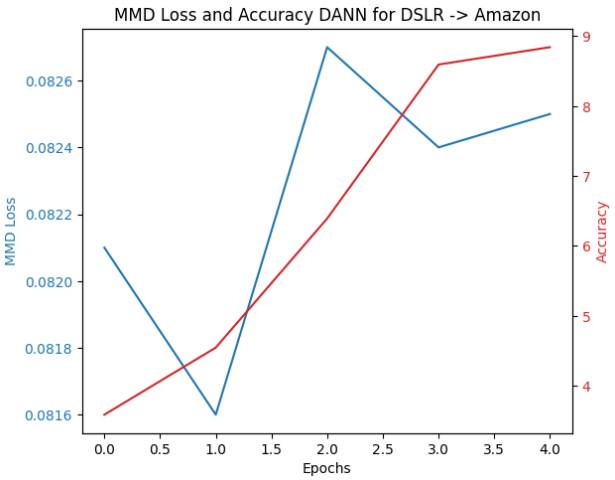

Figure 19. MMD Loss with Accuracy across Epochs

  - The ResNet-50 feature extractor might not capture domain-invariant features effectively, limiting the model's ability to generalize to the target domain.
- **Limited Training Data:**
  - Insufficient target domain samples can hinder the model's ability to learn meaningful domain-invariant features.
- **Domain Confusion:**
  - The shared feature space might confuse domain-specific and class-specific information, leading to degraded performance on classification tasks.
- **Model Complexity:**
  - The classifier or domain discriminator architecture might not have sufficient capacity to model complex domain relationships keeping in mind the computational constraints and the low number of training epochs.
- **Vanishing Gradient Issue:**
  - The Reverse Gradient Layer may cause gradients to vanish, slowing down or halting the training process for domain adaptation.

4.1.5. Potential Improvement DANN

Based on the DANN results here we provide some potential improvements that could be made for get better accuracies/results:

- **Improving Domain Discriminator:**

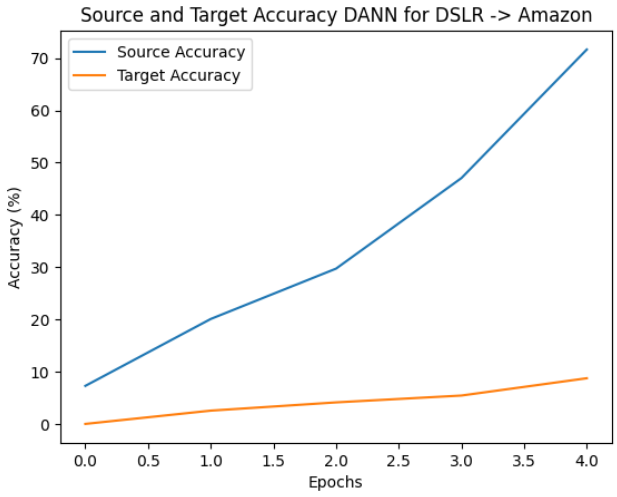

Figure 20. Source and Target Accuracies for the DANN Model

  - Enhance the domain discriminator by increasing its capacity with additional layers or using an ensemble of domain discriminators.
  - **Reasoning:** A more expressive domain discriminator can better differentiate source and target domain features, leading to improved domain-invariant feature learning.

- **Feature Space Regularization:**
  - Add a contrastive loss or center loss to encourage class separation and reduce domain confusion in the shared feature space.
  - **Reasoning:** Regularization ensures that domain-invariant features remain discriminative for classification tasks.

- **Feature Extractor Enhancement:**
  - Replace the ResNet-50 backbone with a more robust feature extractor or fine-tune pre-trained weights specifically for domain adaptation tasks.
  - **Reasoning:** A more effective feature extractor can learn better high-level domain-invariant representations.

- **Increasing Training Iterations:**
  - Train the model for more epochs and use early stopping based on target domain validation accuracy to allow better convergence.
  - **Reasoning:** Longer training allows the model to refine domain-invariant features and improve target accuracy.

4.2. Task 2 - DG with IRM Variants

This section discusses the results obtained in Part 1 and Part 2, focusing on out-of-distribution (OOD) generalization, in-distribution (IID) accuracy, and the trade-offs observed across methods.

Part 1: OOD Accuracy and Regularization Effects

The goal of Part 1 was to evaluate the OOD accuracy of Basic IRM, IB-IRM, and PAIR while analyzing the impact of regularization in IB-IRM and PAIR. The key findings are summarized below:

1. **Which method achieves the highest OOD accuracy?**

- **IB-IRM** achieves the highest OOD accuracy of *37.45%* ($\gamma = 0.001$), as shown in Table 4.

- **PAIR**, with its best weight configuration ($w_1 = 0.1, w_2 = 1.0, w_3 = 0.01$), achieves an OOD accuracy of *28.62%*, outperforming Basic IRM but lagging behind IB-IRM.

- **Basic IRM** achieves a maximum OOD accuracy of *19.49%* ($\lambda = 0.5$).

These results, visualized in Figures 1, 2, and 3, clearly demonstrate that *IB-IRM* is the most effective in generalizing to unseen domains.

2. **How does regularization affect the trade-off?**

- In **IB-IRM**, increasing the penalty weight γ enhances OOD generalization by enforcing invariant representation learning. However, overly high values (e.g., $\gamma = 0.1$) result in reduced OOD accuracy, highlighting the importance of careful tuning.

- **PAIR**'s performance is sensitive to the weight configuration. Balanced weights ($w_1 = 0.1, w_2 = 1.0, w_3 = 0.01$) yield better OOD accuracy, but the method struggles with domain alignment, leading to inferior results compared to IB-IRM.

- **Basic IRM**, with its simpler invariance penalty, is less effective at separating spurious correlations, resulting in weaker OOD performance overall.

These trends confirm that *regularization plays a critical role in improving OOD accuracy*, with IB-IRM achieving the best trade-off through its information bottleneck penalty.

Part 2: IID vs. OOD Accuracy Trade-offs

Part 2 explored the relationship between IID and OOD accuracy for each method, as shown in Table 6 and Figure 7. The trends and trade-offs observed are discussed below:

1. **Does improving IID accuracy degrade OOD accuracy?**

- For **Basic IRM**, both IID and OOD accuracy improve

with training (from *29.16%* IID and *17.51%* OOD at the final epoch). However, the gap between IID and OOD performance remains significant, indicating that the model struggles to generalize across domains.

- **IB-IRM** achieves the most balanced improvement, with IID accuracy increasing to *41.07%* and OOD accuracy to *35.57%*. This proportional improvement suggests that IB-IRM effectively balances learning across domains without overfitting to IID tasks.

- **PAIR**, while showing modest IID improvement (*32.72%*), fails to generalize to the OOD domain, with OOD accuracy stagnating at *14.06%*. This indicates that PAIR overfits to IID tasks.

These results demonstrate that *IB-IRM strikes the best balance between IID and OOD accuracy*, while Basic IRM and PAIR fail to achieve similar levels of generalization.

2. **Which method achieves the best balance between IID and OOD accuracy?**

- As evident in Table 6, **IB-IRM** exhibits the smallest IID-OOD gap (*5.50%*), highlighting its robust generalization capabilities.

- **PAIR** shows the largest IID-OOD gap (*18.66%*), suggesting poor generalization and overfitting to training domains.
- **Basic IRM** achieves moderate performance but fails to match the robustness of IB-IRM.

Figure 7 clearly illustrates the trade-offs, with IB-IRM achieving the best balance.

Summary of Findings

- **IB-IRM** consistently outperforms Basic IRM and PAIR in both OOD accuracy and IID-OOD trade-offs.

- Regularization is critical for OOD generalization, with IB-IRM's information bottleneck penalty providing the most effective control over invariant representation learning.

- PAIR's performance highlights the challenges of balancing multiple objectives, emphasizing the need for fine-tuned weight configurations.

4.3. Task 3 - Discussion

4.3.1. Disentanglement vs. Reconstruction Trade-Off

The results confirm the theoretical expectations of β-VAE. As β increases, the disentanglement of latent factors improves at the expense of reconstruction quality. This trade-off is evident both quantitatively, in the form of increased reconstruction loss, and qualitatively, with degraded image details for higher β values:

- For $\beta = 1$, the model exhibits high reconstruction fidelity but weaker disentanglement.
- For $\beta = 50$, the latent factors are strongly disentangled, but the reconstructions are blurry and lack fine-grained details.
- Intermediate values of β, such as 5 and 10, provide a balance, offering moderate disentanglement with acceptable reconstruction quality.

4.3.2. Key Insights from Quantitative Metrics

- **Z-Diff Metric:** Increases with higher β values, indicating stronger disentanglement. At $\beta = 50$, the Z-Diff reaches its peak, showing that latent factors are well-separated.
- **MIG Metric:** Remains consistently low, even for high β values, suggesting room for improvement in disentanglement methods.
- **Reconstruction Loss:** Increases steadily with β, confirming that stronger regularization comes at the cost of reconstruction fidelity.
- **BetaVAE and FactorVAE Metrics:** Remain stable across all β values, reflecting the model's consistency in capturing major factors of variation.

4.3.3. Visual Analysis of Disentanglement

The qualitative visualizations provide additional insights into the effect of β:

- **For $\beta = 1$:** The generative factors are less distinct, and variations across dimensions are subtle. The reconstructions, however, closely match the input images.
- **For $\beta = 5$:** Disentanglement improves, with more interpretable variations in attributes like azimuth and style. Minor blurring is observed in the reconstructions.
- **For $\beta = 10$:** Disentanglement becomes more prominent, but the reconstructions start losing significant detail, demonstrating the trade-off between disentanglement and quality.
- **For $\beta = 50$:** Disentanglement is the strongest, but the images are highly blurred, and fine-grained details are lost.

4.3.4. How Does Increasing β Affect Disentanglement and Reconstruction Quality?

- **Stronger Regularization:** Higher β enforces stronger regularization, encouraging the model to prioritize disentanglement over reconstruction quality.

- **Improved Disentanglement:** Disentanglement metrics such as Z-Diff improve significantly for higher β, highlighting the model's ability to capture distinct generative factors.
- **Degraded Reconstruction Quality:** Reconstruction loss increases, leading to blurry and less detailed images. This is evident from both visualizations and quantitative metrics.
- **Trade-Off Decision:** The choice of β depends on the application requirements. For tasks demanding high-quality reconstructions, lower β values are preferable, while higher β values are suited for applications emphasizing disentanglement.

4.3.5. Limitations of β-VAE

While the β-VAE improves disentanglement, it has several limitations:

- **Loss of Fine-Grained Details:** Higher β values lead to blurry reconstructions, limiting its use in applications requiring high fidelity.
- **Incomplete Disentanglement:** Despite improvements in metrics like Z-Diff, others such as MIG remain low, indicating that not all factors are fully disentangled.
- **Sensitive Hyperparameter Tuning:** The model is highly sensitive to β and other hyperparameters, requiring extensive tuning for optimal performance.
- **Redundant Latent Dimensions:** Some dimensions remain noisy or unused, reducing the interpretability of latent space.

4.3.6. Future Directions and Recommendations

To address the observed limitations:

- **Hybrid Models:** Combine β-VAE with other disentanglement methods, such as DIP-VAE or FactorVAE, to achieve better performance.
- **Adaptive Regularization:** Implement adaptive β scheduling to balance disentanglement and reconstruction dynamically during training.
- **Advanced Metrics:** Use improved metrics like DCI (Disentanglement, Completeness, Informativeness) to evaluate the quality of latent representations comprehensively.
- **Post-Processing Techniques:** Apply techniques like clustering or pruning in the latent space to enhance interpretability and reduce redundancy.

5. Conclusion

This report highlights the potential and limitations of advanced machine learning techniques in three distinct tasks:

- **Task 1: UDA** demonstrated the importance of domain alignment in achieving robust performance across source and target domains. The DANN model showed promising results in aligning features, but stability and accuracy gaps remain a challenge.
- **Task 2: DG** emphasized the trade-offs in invariant risk minimization methods. IB-IRM provided the best balance between IID and OOD accuracy, confirming the role of regularization in improving OOD generalization.
- **Task 3: β-VAE** showcased the trade-off between disentanglement and reconstruction quality. Higher β values improved disentanglement but resulted in degraded reconstruction quality, reflecting the limitations of fixed hyperparameter approaches.

The findings underscore the need for hybrid models, adaptive regularization, and advanced metrics to address these challenges effectively. Future work should focus on integrating complementary methods, exploring new datasets, and enhancing interpretability for real-world applications.

6. Contributions

- **Muhammad Saad Haroon**: Task 3
- **Jawad Saeed**: Task 1
- **Daanish ud Din**: Task 2

References

[1] Wouter M. Kouw and Marco Loog. "An introduction to domain adaptation and transfer learning." arXiv preprint arXiv:1812.11806, 2018.

[2] Isabela Albuquerque, Joao Monteiro, Tiago H. Falk, and Ioannis Mitliagkas. "Adversarial target-invariant representation learning for domain generalization." CoRR, 2019.

[3] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. "Adapting visual category models to new domains." In Computer Vision–ECCV 2010, pages 213–226. Springer, 2010.

[4] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. "Deep hashing network for unsupervised domain adaptation." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 5018–5027, 2017.

[5] Yann LeCun, Leon Bottou, Yoshua Bengio, and Patrick Haffner. "Gradient-based learning applied to document recognition." Proceedings of the IEEE, 86(11):2278–2324, 1998.

[6] Jonathan J. Hull. "A database for handwritten text recognition research." IEEE Transactions on Pattern Analysis and Machine Intelligence, 16(5):550–554, 1994.

[7] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Baolin Wu, Andrew Y. Ng, et al. "Reading digits in natural images with unsupervised feature learning." In NIPS Workshop on Deep Learning and Unsupervised Feature Learning, volume 2011, page 4. Granada, 2011.

[8] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. "Learning transferable features with deep adaptation networks." In International Conference on Machine Learning, pages 97–105. PMLR, 2015.

[9] Yaroslav Ganin and Victor Lempitsky. "Unsupervised domain adaptation by backpropagation." In International Conference on Machine Learning, pages 1180–1189. PMLR, 2015.

[10] Muhammad Ghifary, W. Bastiaan Kleijn, Mengjie Zhang, and David Balduzzi. "Domain generalization for object recognition with multi-task autoencoders." In Proceedings of the IEEE International Conference on Computer Vision, pages 2551–2559, 2015.

[11] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. "Deeper, broader and artier domain generalization." In Proceedings of the IEEE International Conference on Computer Vision, pages 5542–5550, 2017.

[12] Chen Fang, Ye Xu, and Daniel N. Rockmore. "Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias." In Proceedings of the IEEE International Conference on Computer Vision, pages 1657–1664, 2013.

[13] Martin Arjovsky, Leon Bottou, Ishaan Gulrajani, and David Lopez-Paz. "Invariant risk minimization." arXiv preprint arXiv:1907.02893, 2019.

[14] Kartik Ahuja, Ethan Caballero, Dinghuai Zhang, Jean-Christophe Gagnon-Audet, Yoshua Bengio, Ioannis Mitliagkas, and Irina Rish. "Invariance principle meets information bottleneck for out-of-distribution generalization." Advances in Neural Information Processing Systems, 34:3438–3450, 2021.

[15] Yongqiang Chen, Kaiwen Zhou, Yatao Bian, Binghui Xie, Bingzhe Wu, Yonggang Zhang, Kaili Ma, Han Yang, Peilin Zhao, Bo Han, et al. "Pareto invariant risk minimization: Towards mitigating the optimization dilemma in out-of-distribution generalization." arXiv preprint arXiv:2206.07766, 2022.

[16] Xin Wang, Hong Chen, Zihao Wu, Wenwu Zhu, et al. "Disentangled representation learning." IEEE Transactions on Pattern Analysis and Machine Intelligence, 2024.

[17] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. "Large-scale CelebFaces Attributes (CelebA) dataset." Retrieved August, 15(2018):11, 2018.

[18] Mathieu Aubry, Daniel Maturana, Alexei A. Efros, Bryan C. Russell, and Josef Sivic. "Seeing 3D chairs: Exemplar part-based 2D-3D alignment using a large dataset of CAD models." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3762–3769, 2014.

[19] Loic Matthey, Irina Higgins, Demis Hassabis, and Alexander Lerchner. "dSprites: Disentanglement testing sprites dataset." https://github.com/deepmind/dsprites-dataset/, 2017.

[20] 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 International Conference on Machine Learning, pages 4114–4124, 2019.