CS6304

Mitigating Color-Induced Spurious Correlations in Knowledge Distillation for Robust Image Classification

Ahsan Mir
Department of Computer Science
Lahore University of Management Sciences
25100325@lums.edu.pk

Abdulhaseeb Khan
Department of Computer Science
Lahore University of Management Sciences
25100077@lums.edu.pk

Aazen Saleem
Department of Computer Science
Lahore University of Management Sciences
25100031@lums.edu.pk

Abstract

Deep image classifiers often rely on spurious correlations, such as characteristic background colors associated with certain classes. Such biases are problematic in Knowledge Distillation (KD), where a student model inherits the teacher's biases. We propose three remedial techniques, Adaptive Color Knowledge Distillation (ACKD), Adversarial Color Training, and Feature Decorrelation, to reduce color-induced biases in KD. We use the Waterbirds benchmark, known for color-based spurious correlations, and evaluate our approach using state-of-the-art measurements of color bias such as Mutual Information-based Color Reliance, Color Sensitivity Score, and Feature Independence Score. Through tracking these measurements and providing qualitative visualizations such as GradCAM in our experiments, we show improved out-of-distribution (OOD) performance and reduced color dependency, surpassing simple color jitter augmentation and naive KD baselines.

1 Introduction

Models trained on large-scale datasets usually learn shortcut features not aligned with semantic content. One prevalent shortcut is color bias, where a model correlates specific color backgrounds or hues with target classes. Geirhos et al. (1) have shown that ImageNet-trained CNNs rely more on texture and color than on shape, causing performance drops under stylized or color-shifted conditions. In the Waterbirds dataset (2), models exploit the correlation that "waterbirds" appear primarily over bluish water backgrounds. Consequently, if a waterbird appears on a non-blue background, the model often fails. This over-reliance on color is a "spurious correlation" that does not hold under distributional changes, reducing model robustness and trustworthiness.

In KD, we transfer knowledge from a large, complex teacher model to a smaller student model. However, if the teacher is biased, naive KD can propagate and reinforce these spurious correlations. While simple data augmentation such as color jittering (3) can partially mitigate bias, it may not directly address representation-level dependencies that cause the bias. We thus need methods that

CS-6304 Advanced Topics in Machine Learning
Template from NeurIPS 2024

integrate bias reduction mechanisms into the KD process, tackling the issue at multiple levels: input transformation, adversarial training, and feature-space regularization.

2 Related Work

Spurious Correlations and Debiasing: Previous works have identified that standard training regimes often learn superficial cues. Studies like Sagawa et al. (2) show that standard ERM-trained models struggle on minority groups that break these spurious correlations. Proposed solutions include Invariant Risk Minimization (4) or targeted data augmentations (5). Yet, such methods often assume direct access to debiasing techniques or focus on re-training from scratch. Moreover, recent advances in domain generalization have highlighted how classifiers overfit to domain-specific features such as texture or color, leading to poor robustness when test conditions deviate from training distributions (11). Approaches such as GroupDRO (2) or specialized adversarial training (6) have shown promise in weakening reliance on spurious features. However, explicit strategies for mitigating spurious correlations in knowledge transfer scenarios remain comparatively underexplored. Our work extends this line of research by explicitly focusing on color-induced bias reduction within the context of knowledge distillation.

Color Bias: CNNs often exploit low-level statistical regularities instead of learning stable semantic features (1; 2; 11). Color bias is a well-documented case, where models learn to associate certain color patterns with class labels, deteriorating out-of-distribution (OOD) performance. Existing approaches have introduced adversarial color perturbations (6) or stronger augmentations to weaken the reliance on color. However, these approaches typically do not integrate seamlessly with KD objectives. They treat debiasing and compression as separate concerns. Furthermore, some works propose color-invariant methods by training models on stylized datasets (1), but these methods often require significant data preprocessing. Our work leverages color transformations adaptively based on color reliance, thereby aligning debiasing efforts with the distillation process itself.

Knowledge Distillation and Robustness: While KD is widely used to compress models (7), less attention has been given to ensuring that the distilled student is robust and unbiased. Attempts to improve KD robustness often focus on preserving semantic feature alignment (8) or using contrastive objectives (9), but not specifically addressing spurious color correlations. Recent research points out that distillation can amplify teacher biases when the teacher is already over-reliant on non-generalizable features (2; 1). Consequently, naive KD may propagate undesirable shortcuts into the student model, particularly if color or texture cues dominate teacher predictions. Our work aims to fill this gap by introducing and comparing targeted bias-mitigation mechanisms within the distillation framework. By explicitly measuring color reliance (B_{MI}), color sensitivity (CSS), and feature independence (FIS), we offer a more holistic view of how KD can be adapted to address spurious correlations and strengthen OOD robustness.

3 Dataset

For each of our experiments, we make use of the Waterbirds dataset introduced by Sagawa et al. (2). It is a benchmark dataset specifically designed to study and address spurious correlations in image classification tasks. The dataset comprises of approximately 28,000 224x224 RGB images of birds categorized into two primary classes: Waterbirds and Landbirds. Each bird class is predominantly associated with specific background environments (land or water), introducing a clear spurious correlation between bird type and background color/texture.

3.1 Dataset Splits

- **Training Set ($\mathcal{D}_{\text{train}}$):** The training split has a biased distribution, with Waterbirds predominantly appearing against water backgrounds and Landbirds against land backgrounds. We use this split to train the teacher model f_T and, subsequently, to distill knowledge into the student model f_S.
- **In-Distribution Test Set ($\mathcal{D}_{\text{test\_ID}}$):** The distribution is similar to the training set, maintaining the spurious correlation between bird type and background. We use this to evaluate the

model's performance on data that aligns with the training distribution, ensuring that models retain high accuracy on familiar data.

- **Out-of-Distribution Test Set ($\mathcal{D}_{\text{test\_OOD}}$):** The distribution of this split contradicts the training set correlations by pairing Waterbirds with land backgrounds and Landbirds with water backgrounds. We leverage this split to assess the model's generalization capabilities and robustness when faced with data that breaks the learned spurious correlations.
- **Validation Set ($\mathcal{D}_{\text{val}}$):** Distribution mirrors the training set, maintaining the spurious correlations. Used to monitor model performance and compute bias metrics (e.g., B_{MI}) during training, especially crucial for adaptive methods like ACKD.

4 Evaluation Metrics

In each experiment, we aim to assess the ID and OOD performance on the dataset (classification accuracy on original and shifted test sets) to ensure we do not overly degrade accuracy on the original color-biased distribution and achieve improved accuracy on backgrounds that break the color correlation. Moreover, we discuss some additional evaluation metrics in this section that will be used in our experiments for qualitatively and quantitatively assessing color reliance in our models.

4.1 Mutual Information (MI)-based Color Reliance B_{MI}

We treat the model predictions $\hat{Y}$ and color channel statistics $\mathcal{C}$ (e.g., average RGB intensities in an image region) as random variables. This helps us measure how knowing one variable, color, reduces uncertainty about another variable, the predicted label. The Mutual Information is defined as follows:

$$B_{\text{MI}} = I(\hat{Y}; \mathcal{C}) = \sum_{\hat{y},c} P(\hat{y}, c) \log \frac{P(\hat{y}, c)}{P(\hat{y})P(c)},$$

This formula tells us how dependent our model prediction is on color. High B_{MI} would mean the model's predictions are highly influenced by color, for instance, if certain backgrounds or color channels strongly correlate with the predicted bird class. Therefore, our goal is to minimize this metric. We will estimate $P(\hat{y}, c)$ from validation data using discrete bins (10).

4.2 Color Sensitivity Score (CSS)

CSS measures how sensitive the model is to a small color shift δ_c:

$$\text{CSS} = \frac{\mathbb{E}_X \left[\| f_S(X) - f_S(X + \delta_c) \|_2 \right]}{\mathbb{E}_X \left[\| X - (X + \delta_c) \|_2 \right]}.$$

A high CSS value would mean a small color shift leads to a big change in the model's outputs. Conversely, a low CSS implies the model is fairly robust to color changes. Hence, our goal is to lower this metric to ensure predictions remain stable under color shifts, reducing color dependence.

4.3 Feature Independence Score (FIS)

Let $F \in \mathbb{R}^{N \times d}$ be a matrix of extracted features from an intermediate layer and $\mathcal{C} \in \mathbb{R}^{N \times 3}$ the color features (e.g., mean RGB). We compute the cross-covariance $\Sigma_{F,\mathcal{C}}$ and find the maximum absolute correlation $\rho_{\max}$ between any feature dimension and color channel. This will help us check how correlated each internal feature dimension is with the color channels.

$$\text{FIS} = 1 - \rho_{\max}.$$

High FIS means feature representations are less tied to color. Minimizing correlation encourages more general, color-invariant features. Hence, we need to increase FIS from the baseline.

4.4 Gradient-weighted Class Activation Mapping (Grad-CAM)

Grad-CAM provides visual insights into the model's focus areas in an input image, helping to determine whether predictions rely on color regions or meaningful structures. Given an input image X and a target class k, Grad-CAM generates a localization map $L^k_{\text{Grad-CAM}}$. The heatmaps highlight image regions contributing most to the prediction for class k, helping us assess whether the model focuses on color-dominated regions or object structures. Through mitigation experiments outlined later in this document, we hope to highlight structural features in the Grad-CAM visualizations, reflecting reduced color bias.

5 Experiments

5.1 Configuration

For each experiment that we conduct, we use the ResNet-50 model as our teacher and ResNet-18 as our student model. Learning rate is 0.001. We will fine-tune the teacher and train the student for 20 epochs each. KD temperature is set to 4 and KD alpha value is set to 0.5.

Now, we detail the experiments designed to evaluate both baseline and proposed methods. We first describe three baseline experiments to establish reference performance levels and to highlight the challenge of mitigating color-induced spurious correlations. We then introduce three proposed approaches that aim to systematically reduce reliance on color cues while preserving model performance.

5.2 Baseline Experiments

5.2.1 Simple Color Jitter Augmentations Only

In our first baseline experiment, we wanted to observe the effect of changing the background colours of the images on the accuracies of the models. The model we used in our approach was a simple ResNet-18 model, which would be used in our main experiments as well later on. Using a batch size of 32, a learning rate of 0.001, cross-entropy loss, and 20 epochs, we fine-tuned our ResNet-18 model on the unaltered Waterbirds dataset and reported the accuracies after validation. This was a fairly simple fine-tuning task meant to reveal the current situation and the effect colour bias has on the accuracies (results discussed in the result section).

The next part of the experiment was identical to the first part, with only one change. This time we added jitter to each image in the dataset. The jitter was induced in the transforms part of the code with the following values: brightness=0.4, contrast=0.4, saturation=0.4, hue=0.1. Once again, all parameters were the same as before. After the training of this jittered model was complete, we evaluated it on the test set containing OOD data as well as on the IID data to record the accuracies. Moreover, we used Grad-CAM to visualize the features that the jittered and un-jittered models are using. Our aim was to maximize the OOD accuracy and make the model predict the class using actual features of birds rather than the features in the background.

5.2.2 Naive Direct Knowledge Distillation (KD)

In this next baseline experiment, we aim to see if we can reduce color-induced spurious correlations using Knowledge Distillation (KD). In this setup, we implement simple knowledge distillation without any color bias reduction techniques, and we want to see if a teacher model transfers over its potential color-based bias to its student model. The purpose of conducting this is to determine how much the student model inherits color bias from its teacher model.

The equation for the loss function used to implement knowledge distillation is as follows:

$$L_{\text{KD}} = (1-\alpha)L_{\text{CE}}(f_S(X), Y) + \alpha T^2 \text{KL}(p_T(X), p_S(X))$$

The hyperparameter T in the above equation is the distillation temperature. α controls the knowledge contribution from the teacher and true labels by the student. The teacher model was trained using a simple cross-entropy loss function. $p_T(X)$ represents the teacher's predicted probabilities of each class after being softmaxed with temperature, showing relationships between classes, while $p_S(X)$

represents the student's predicted probabilities derived using the same method to align the student and the teacher's output.

Once trained, the teacher model transferred its knowledge to the student using a combination of cross-entropy loss and KL-divergence to form the knowledge distillation loss.

5.2.3 KD with Color Jitter

In our final baseline experiment, we combine our naive KD with color jitter augmentations. The goal of this experiment is to determine whether adding a simple color bias reduction technique such as color jitter can help remove color bias from the student model after the knowledge distillation process.

The loss function used to implement knowledge distillation with color jitter is as follows:

$$L_{\text{KD-jitter}} = (1-\alpha)L_{\text{CE}}(f_S(X_{\text{aug}}), Y) + \alpha T^2 \text{KL}(p_T(X_{\text{aug}}), p_S(X_{\text{aug}}))$$

This equation is similar to the one used in naive KD, but the key difference is that we pass color-jittered images (X_{aug}) as input to both the softened probability distributions of the teacher and the student in the KL-divergence term, as well as to the student model.

5.3 Proposed Approaches

To overcome the limitations of the baseline methods, we propose three strategies. Each approach is grounded in theoretical principles and aims to reduce the mutual information (MI) between model predictions and color cues, the color sensitivity of predictions, and the correlation between features and color channels.

5.3.1 Adaptive Color Knowledge Distillation (ACKD)

In this approach, we aim to integrate real-time feedback about color bias into the training process. While our naive KD implementations typically involve the teacher and student seeing the same input images, ACKD modifies this by dynamically color-augmenting the inputs to the student after each epoch, which differs from the exact input which the teacher was trained on. In our loss function, we modify the combination of cross-entropy on the original data and KL-divergence term between the teacher and student logits to incorporate the adaptive color transform on the student's input, defined as following:

$$L_{\text{ACKD}} = (1-\alpha)L_{\text{CE}}\big(f_S(T_\lambda(X)), Y\big) + \alpha T^2 KL\big(p_T(X), p_S(T_\lambda(X))\big),$$

where $T_\lambda(X)$ denotes the adaptive color transformation. This transformation is chosen according to $\lambda(B_{MI})$, where λ is a value between 0 and 1 (inclusive) that is set dynamically at the start of each epoch based on the latest value of B_{MI} computed on the validation set. We determine lambda using the following formula:

$$\lambda = \min\big(\max(0,\, \text{scale} \times B_{\text{MI}}), 1\big).$$

If BMI is high, we intensify color augmentation, and vice versa. This adaptive real-time feedback mechanism helps us apply just enough color shift to gradually reduce color bias without excessively harming overall performance. Although it is unusual for our mutual information value to be negative, we use $\max(0,\, \text{scale} \times B_{\text{MI}})$ instead of only using scale $\times\, B_{\text{MI}}$ in the first argument of the min function to ensure a strict lower bound of 0, preventing any negative values that may be caused by binning edge cases.

The "scale" value is a knob that decides how quickly we ramp up color transformations if the model is discovered to be color-biased. We find setting this value to 0.1 to be a sweet spot that prevents overly drastic color changes in the input image to our student. The rest of our KD pipeline remains the same, but including this extra feedback-loop encourages the student to align with the teacher's logits without relying heavily on color cues.

5.3.2 Adversarial Color Training

Building upon the baseline experiments involving knowledge distillation, this experiment used the same setup as baseline 2. The batch size was 32, learning rate was 0.001, epochs were 20, KD temperature was 4.0, and KD alpha was 0.5. Moreover, the classifier head was modified to perform binary classification for both teacher (ResNet-50) and student (ResNet-18). The loss functions present in the code are simple KD loss and adversarial loss. The adversarial loss is the one used in the training process of the student. The purpose of the adversarial loss is to combine the standard loss, the KD loss, and the cross-entropy loss. This combination is called the adversarial loss.

$$L_{adv} = (1-\alpha)L_{CE}(f_S(X+\delta_c), Y) + \alpha T^2 KL(p_T(X+\delta_c), p_S(X+\delta_c))$$

The cross-entropy loss between the student model's predictions on the adversarially perturbed input and the true labels is $L_{CE}(f_S(X+\delta_c), Y)$. The Kullback-Leibler (KL) divergence between teacher and student models' softened probability distributions on adversarially perturbed input is $KL(p_T(X+\delta_c), p_S(X+\delta_c))$. T is the temperature used in knowledge distillation to control the softness of the predicted probability distribution.

We used a function to generate adversarial colour perturbations in the colour channels of the images. This function works by using the teacher model, the batch of images, the true labels, and an epsilon value, which is then used to control the magnitude of the perturbations. The value of epsilon that we chose was 0.1. Using this approach, we aimed to make our model learn the features itself rather than using the colours to make predictions. The goal of the experiment was to minimize the colour sensitivity score. Lastly, the student model was trained by the teacher using simple knowledge distillation, and the adversarial images generated via the teacher model were used in the loss function.

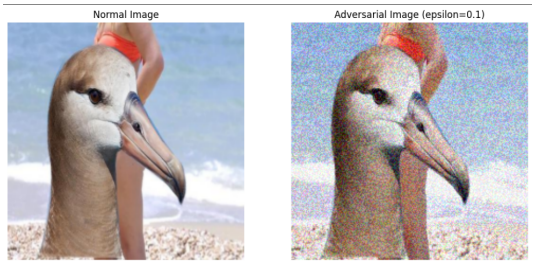

Figure 1: Adversarial Color Perturbations

5.3.3 Feature Decorrelation

This approach explores our final technique of mitigating color-induced spurious correlation using knowledge distillation. This approach is called Feature Decorrelation. In this approach, we add a penalty in the Knowledge Distillation loss function to penalize spurious features such as colors and instead focus on class-relevant features.

We first extract feature embeddings from the student model's fourth layer and retrieve color features, which are mean RGB values for each input image. Once we get these, we compute the Cross-Covariance matrix to capture the correlation between feature embeddings (F) and color features (C), as defined by the following equation:

$$\Sigma_{F,C} = \frac{1}{N-1}(F-\mu_F)^\top (C-\mu_C)$$

Here:

- N is the number of samples.
- μ_F is the mean of the feature embeddings.
- μ_C is the mean of the color features.

The purpose of measuring cross-covariance is to quantify the degree to which each feature embedding aligns with each of the RGB color channels.

Next, we apply the decorrelation penalty, which is defined by the squared Frobenius norm of the cross-covariance matrix. By minimizing this penalty, we encourage the model to learn features that are less dependent on color:

$$L_{\text{decor}} = \|\Sigma_{F,C}\|_F^2$$

Finally, we incorporate this penalty into the Knowledge Distillation loss function. By doing so, the student model is penalized when its feature embeddings align with color features. The modified Knowledge Distillation loss function is given as:

$$L_{\text{KD-decor}} = L_{\text{KD}} + \beta L_{\text{decor}}$$

This encourages the student model to learn class-relevant features that are independent of color.

6 Results

In this section, we outline and discuss the results from each of our experiments. In addition to reporting the evaluation metrics, we also evaluate on two test sets from the Waterbirds benchmark: the In-Distribution (ID) test set, which maintains the original color-class correlations, and the OOD test set, which presents waterbirds in non-water backgrounds and landbirds in non-land backgrounds. By comparing model accuracy on both sets, we assess whether improvements in OOD generalization are achieved without sacrificing ID performance.

6.1 Simple Color Jitter Augmentations

The results of baseline experiment 1 were as expected. The accuracy of the simple Waterbirds ResNet-18 model was 79.86% on the IID data, whereas it dropped to 28.32% on the OOD data. If we look at the accuracies of the model trained on the jittered dataset, it showed a lower accuracy on IID but a higher accuracy on the OOD data, achieving 77.45% and 36.35%, respectively. Reasons will be discussed in the discussion section.

Figure 2: un-jittered model (up) vs jittered model (down)

This experiment provided us with valuable insight regarding the behavior of our models. The minor drop in IID accuracy of the jittered versus un-jittered model showed us that, for the same number of epochs, it becomes harder for the model to learn shortcuts by looking at the backgrounds. So it has to focus more on the features of the birds rather than looking at other unwanted correlations. Moreover, the increase in accuracy of the OOD data revealed that due to the jitter, the model actually did learn the features of the birds better than before. This was further confirmed by Grad-CAM images, as the signature shifted from the background to the neck of the bird, as displayed below.

Metric	Value
ID Accuracy (Majority Groups)	92.41%
OOD Accuracy (Minority Groups)	30.34%
Mutual Information B_{MI}	0.3284
Color Sensitivity Score (CSS)	0.3523
Feature Independence Score (FIS)	0.3667

6.2 Naive Direct Knowledge Distillation (KD)

Figure 3: Grad-CAM Visualization: Shows that the student model is more focusing on the background rather than the bird.

Two key observations can be drawn from our results: the in-distribution (ID) accuracy is strong at 92.41%, showing that our model performs well in classifying ID samples. However, the out-of-distribution (OOD) accuracy is only 30.34%, which is abysmally low. The poor OOD accuracy can be attributed to the lack of color bias mitigation techniques in this baseline.

The high color sensitivity score (CSS) of 0.3523 further confirms that the model inherits the teacher's color bias. Grad-CAM visualizations show that the model focuses on the background colors rather than the bird features for the predictions, indicating a significant color bias. This baseline serves as a control to demonstrate that knowledge distillation transfers color bias to the student model, which will be compared to future experiments that incorporate bias reduction techniques.

6.3 KD with Color Jitter

Metric	Value
ID Accuracy (Majority Groups)	91.58%
OOD Accuracy (Minority Groups)	33.34%
Mutual Information B_{MI}	0.3641
Color Sensitivity Score (CSS)	0.3897
Feature Independence Score (FIS)	0.4552

Figure 4: Grad-CAM Visualization: Shows that the student model is now more focusing is somewhat improved and focusing on the features of the bird and somewhat on the background.

The results show a clear improvement in OOD accuracy, which increased from 30.34% in the naive KD baseline to 33.34% in this experiment. This suggests that applying color jitter augmentation helped mitigate some of the color-induced biases transferred from the teacher to the student model.

The feature independence score (FIS) also improved, rising to 0.4552, indicating that the model is less dependent on color cues for decision-making. However, the bias mitigation index (B_MI) and color sensitivity score (CSS) increased slightly to 0.3641 and 0.3897, respectively, showing that some residual color dependency remains.

Grad-CAM visualizations reveal that the student model focuses more on the bird's features (e.g., wings) compared to the naive KD setup, where predictions were heavily influenced by background colors. Overall, this experiment demonstrates that combining color jitter augmentation with knowledge distillation can reduce, though not completely eliminate, color bias in student models.

6.4 Adaptive Color Knowledge Distillation (ACKD)

Metric	**Value**
ID Accuracy	90.06%
OOD Accuracy	39.01%
Color Sensitivity Score (CSS)	0.0145
Feature Independence Score (FIS)	0.2372
Mutual Information B_{MI}	0.2092

In our ACKD experiment, we observe an improvement in out-of-distribution (OOD) performance, rising to an accuracy of 39.01% (vs. 30.34% under naive KD and 33.34% with color jitter) on the OOD test split. Although the in-distribution (ID) accuracy of 90.06% is slightly lower than the baselines, the significantly higher OOD accuracy suggests more robust generalization across mismatched backgrounds. Furthermore, our mutual information score $B_{\text{MI}} = 0.2092$ is lower than those in the baseline experiments, indicating that we successfully reduced color reliance. This conclusion is also reinforced by the very low color sensitivity score (CSS) of 0.0145, which shows that the model is comparatively stable under color perturbations. However, the feature independence score (FIS) of 0.2372 indicates that some latent correlation with color channels still remains. Qualitative evidence from Grad-CAM (Figure 5) shows that, compared to the baseline visualizations, the model's focus has shifted more toward the bird's head and beak regions rather than relying strongly on the background.

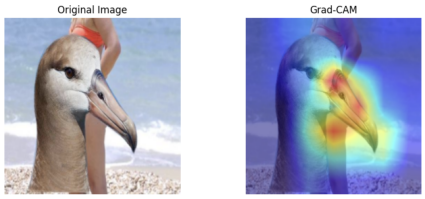

Figure 5: Grad-CAM Visualization for ACKD: The model focuses more on the bird itself compared to the color-dominated attention seen in baseline methods.

6.5 Adversarial Colour Training

Once the student model was trained, we evaluated it on the IID and OOD data once again. The results were as follows: IID accuracy was 87.54% and the OOD accuracy was 29.62%. In addition, the color sensitivity score (CSS) was 0.1678. The mutual information score turned out to be 0.4657, whereas the feature independence score turned out to be 0.3587. Using Grad-CAM images, the focus of our model successfully shifted from background features to bird features.

If we look at the results of this experiment, we can compare it with the simple knowledge distillation baseline task. The mutual information score increased to 0.4657 from 0.3284, indicating that the student model is learning from the teacher more effectively. The IID and OOD accuracies are very similar to those obtained in baseline task 2, which involved simple knowledge distillation. However, if we dig deeper into the results and analyze them, we can see that CSS dropped to 0.1678 from 0.3523. This indicates that even though the accuracies are almost the same as those of simple KD, the reliance of the student model on the background colors has decreased significantly by almost half. It is now more dependent on the actual features of the birds rather than the background colors. This can further be verified by the Grad-CAM image provided below. We can clearly see that the model is now identifying the beak, eyes, and neck with minimal focus on the background. Lastly, the feature independence score (FIS) was almost the same as before, dropping by only 0.01. This is an area where we can work, as FIS should be maximized.

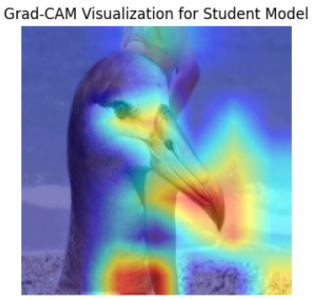

Figure 6: Grad-CAM image illustrating the focus of the model.

6.6 Feature Decorrelation

Metric	Value
ID Accuracy	96.17%
OOD Accuracy	48.33%
Bias Mitigation Index (BMI)	0.1594
Feature Independence Score (FIS)	0.3471
Color Sensitivity Score (CSS)	0.0558

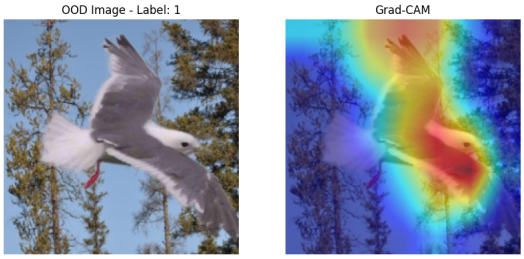

Figure 7: Grad-CAM Visualization: Shows that the student model work much better on focusing on class relevant features such as the wingspan of the bird..

From the GradCAM visualizations, it is evident that the student model now focuses on class-relevant features, such as the wings of the bird, to make predictions instead of relying on the background. This indicates that feature decorrelation effectively penalized the student model for learning features that align with colors, such as the background in OOD images. This improvement is reflected in the metrics: the OOD accuracy increased to 48.33%, and the ID accuracy improved to 96.17%. This is a nearly 18% improvement in the OOD accuracy which compared to the baseline task of knowledge distillation with color jitter. Furthermore, the BMI score decreased significantly, demonstrating a reduction in the disparity between ID and OOD performance. Further evidence of the reduced color reliance in the student model is demonstrated by the low Color Sensitivity Score (CSS) of 0.0558, which is a significant improvement compared to the baseline. However, the experiment fell somewhat short in the Feature Independence Score (FIS), which stood at 0.3471. This suggests that there is still room for optimization in minimizing the dependency of feature embeddings on color.

7 Conclusion

Through the three experiments conducted we achieved significant progress in mitigating color-induced spurious correlations in knowledge distillation. The Feature Decorrelation approach achieved the best OOD accuracy (48.33%) and ID accuracy (96.17%), while also having the lowest BMI value (0.1594) which we were aiming to reduce from our baseline values. The Adaptive Color Knowledge Distillation (ACKD) also showed promise with the best CSS score (0.0145) showcasing how good the approach was in reducing reliance on color perturbations. Finally, Adversarial Colour Training showed the best Feature Independence Score (0.3587) of the three experiments conducted. Further tuning the hyperparameters and experimenting with varying values of KD temperature, KD alpha, learning rate, and epochs can offer valuable insights.

References

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

[2] Sagawa, S., Koh, P. W., Hashimoto, T. B., and Liang, P. Distributionally Robust Neural Networks. In *ICLR*, 2020.

[3] Cubuk, E. D., Zoph, B., Mane, D., Vasudevan, V., and Le, Q. V. AutoAugment: Learning Augmentation Strategies from Data. In *CVPR*, 2019.

[4] Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. Invariant Risk Minimization. *arXiv:1907.02893*, 2019.

[5] Hendrycks, D., Mu, N., Cubuk, E. D., Zoph, B., Gilmer, J., and Lakshminarayanan, B. AugMix: A Simple Data Processing Method to Improve Robustness and Uncertainty. In *ICLR*, 2020.

[6] Li, Y., Liu, X., Wang, C., Zha, Z.J., and Zhang, Y. Shape-Texture Debiased Training for Image Classification. In *ICLR Workshop*, 2020.

[7] Hinton, G., Vinyals, O., and Dean, J. Distilling the Knowledge in a Neural Network. In *NIPS Deep Learning Workshop*, 2015.

[8] Cho, J. and Hariharan, B. On the Efficacy of Knowledge Distillation. In *ICCV*, 2019.

[9] Park, W., Kim, D., Lu, Y., and Cho, M. Contrastive Representation Distillation. In *ICCV*, 2021.

[10] Belghazi, M.I. et al. MINE: Mutual Information Neural Estimation. In *ICML*, 2018.

[11] Gulrajani, I. & Lopez-Paz, D. In Search of Lost Domain Generalization. In *ICLR*, 2021.