S^2-FracMix: Label-Preserving Self-Saliency Mixup Augmentation

Khawar Islam[1]*, Arif Mahmood[2], Xin Jin[3], and Naveed Akhtar[1]

[1] The University of Melbourne, Australia
[2] Information Technology University, Pakistan
[3] Westlake University, China

Abstract. Data augmentation is known to improve generalization of deep visual models. Recent methods favor mixup strategies that generate interpolated samples to improve model performance. However, these techniques not only incur significant computational overhead, they also lead to semantic disruption of augmentation data due to cross-sample mixing. We first propose Self-Saliency (S^2) Mixup, which constructs challenging yet label-consistent samples by extracting multi-scale salient patches and reinserting them into non-salient regions of the same image. This promotes scale-invariant feature learning while avoiding cross-sample interference. To further enhance model robustness, we introduce FracMix, a mixing scheme that injects self-similarity patterns into salient regions using adaptive ratios. Collectively, our unified framework, S^2-FracMix, enables simultaneous learning from fractal and non-fractal structures within a single image, yielding a targeted and structurally coherent augmentation strategy. We theoretically analyze the advantage of our technique, and empirically establish its superiority over the existing methods by achieving state-of-the-art performance in extensive evaluation with seven benchmarks across classification (coarse and fine-grained), robustness, calibration, object detection, and transfer learning tasks. Project page is available at fracmix-data-augmentation.github.io

Keywords: Data Augmentation · Generalization · Robustness

1 Introduction

The rapid growth in the scale and representational capacity of Deep Neural Networks (DNNs) has enabled modern models to effectively memorize training data [4,5,59,65]. While this capacity contributes to strong empirical performance, it also exacerbates overfitting, thereby widening the generalization gap. To address this challenge, data augmentation has emerged as a fundamental research direction [30–32, 45, 47]. Data augmentation strategies have been widely adopted across diverse vision tasks, including image classification [7, 50], object detection [67], and segmentation [13, 28]. By enriching training distributions, these

* Corresponding author: islam.k@unimelb.edu.au

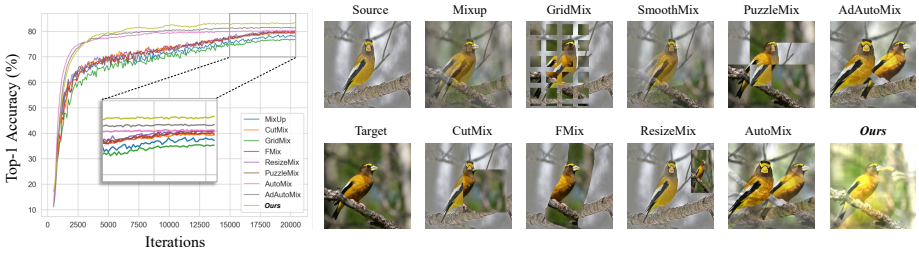

Fig. 1: (Left) Performance of ResNet-18 trained with various augmentation methods for 200 epochs on CIFAR100. (Right) Representative augmentation samples created by different methods. The samples get constructed with source and target images.

methods improve robustness to unseen data, mitigate model collapse [30, 58, 61], and enhance resilience to distribution shifts [29, 45, 48].

A central objective of modern augmentation techniques is to increase sample diversity and robustness while preserving the structural and semantic integrity of the underlying data [15, 19, 24, 28, 29, 45, 46]. Equally important is their practical applicability, which requires a careful balance between performance gains and computational overhead [25, 27, 31, 32]. In this work, we introduce S^2-FracMix, a computationally efficient augmentation framework that promotes greater diversity and structural complexity in augmented samples, thereby improving generalization without incurring substantial overhead (see Fig. 1 & 2).

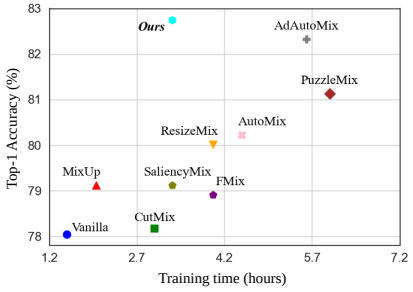

Fig. 2: Comparisons of total training time *vs.* top-1 accuracy of `ResNet-18` on CIFAR-100 dataset with RTX 3090.

Within the broader task-agnostic *mixup* paradigm, samples are typically generated using random pairs of training instances [65]. Representative methods such as CutMix [64], Manifold Mixup [55], AlignMixup [54], and ResizeMix [51] adopt different mixing strategies to construct synthetic training examples. Since these techniques combine random pairs of images, they fall short on preserving semantically salient image regions, potentially leading to suboptimal structural consistency. To address this limitation, saliency-guided methods - including SaliencyMix [53], PuzzleMix [31], Co-Mixup [32], and GuidedMixup [30] - were introduced to prioritize informative regions during mixing. While effective, these approaches typically incur substantial computational overhead, increasing training time and demanding high-end computational resources [31,32]. Other methods such as AutoMix [42] and AdAutoMix [49] attempt to automatically learn optimal mixing strategies and label assignments. However, they remain less effective for Transformer-based architectures, besides their heavy compute overhead. Similarly, fractal-based tech-

niques [20, 24, 26] end up disrupting essential image content while inducing undesirable distribution shifts, which compromises model robustness.

Addressing these challenges, we propose S^2-FracMix, a unified augmentation framework comprising two complementary components: Self-Saliency (S^2) and *FracMix*. The S^2 component extracts multi-scale, saliency-guided patches from an input image, applies diverse transformations to these patches, and reintegrates them into the same image through a controlled mixing operation. By operating within a single image, S^2 avoids cross-sample semantic interference while promoting structural diversity and scale-invariant representation learning. Building upon this foundation, *FracMix* introduces targeted fractal-based augmentation. In contrast to methods such as DiffuseMix [26], which diffuses fractal textures across the entire image, *FracMix* confines self-similar fractal injection to the salient regions identified by S^2. This targeted design increases structural complexity while preserving the clean contextual background. Consequently, each augmented sample simultaneously contains fractal and non-fractal structures, enhancing diversity without inducing undesirable distribution shifts.

Rather than relying on a fixed mixing heuristic, S^2-FracMix incorporates multiple mixing modes that are randomly selected for each training instance. This high-level stochastic mixing strategy further diversifies the training distribution, enabling the model to learn more robust object representations and generalize effectively to unseen data. Our extensive evaluations establish that S^2-FracMix consistently outperforms state-of-the-art augmentation baselines across seven datasets and nine competitive methods. Our approach achieves superior performance in clean accuracy, adversarial robustness, recognition under occlusion, and a broad range of downstream scenarios. The main contributions of this work are summarized as follows:

- We introduce Self-Saliency (S^2) mixing, a multi-scale, saliency-guided augmentation strategy that extracts informative patches, applies diverse transformations, and reintegrates them into the same image.
- We propose *FracMix*, a targeted fractal-based mixing mechanism that injects self-similar structures exclusively within salient regions, increasing structural complexity while preserving contextual integrity and data fidelity.
- We develop a unified framework with S^2 and *FracMix*, termed S^2-FracMix, and extend it with a proposed concept of multi-mode mixing. We also theoretically analyze the key strength of our framework.
- Through comprehensive evaluations on seven datasets and comparisons with nine state-of-the-art methods, we demonstrate consistent improvements across general and fine-grained classification, object detection, transfer learning, self-supervised learning, calibration, and few-shot learning.

2 Related Work

2.1 Mixup Augmentation

Mixup-based augmentation generates diverse training samples through interpolation strategies [21, 35, 63]. Manifold Mixup [55] extends linear interpola-

tion to hidden representations, encouraging smoother decision boundaries in latent space. CutMix [64] replaces rectangular regions between images to promote occlusion-aware learning. Subsequent variants such as FMix [16] and GridMix [2] adopt structured masking strategies, while ResizeMix [51] resizes and overlays patches to introduce scale-aware transformations. SnapMix [23] improves fine-grained recognition by proportionally mixing semantically relevant regions. More recently, Decoupled Mixup [41] proposed an efficient objective with a decoupled regularizer that leverages hard mixed samples to mine discriminative features. Collectively, these methods enhance generalization by synthesizing new samples, yet they largely rely on inter-image mixing without explicitly preserving semantic saliency.

2.2 Automated Mixup Augmentation

Automated mixup approaches balance mixing strategy design with optimization complexity, addressing the disconnect between the mixing heuristics and downstream training objectives. AutoMix [42] jointly optimizes sample generation and classification to produce task-relevant mixed examples. Related directions include adversarial data augmentation [66] and GAN-based methods [1], which seek to automate augmentation through learned generators. Adversarial MixUp [49, 60] further tackles domain shift by synthesizing mixed samples for adaptation. While effective, these approaches often introduce additional training complexity and computational overhead. Their generalizability across network architectures also remains restricted.

2.3 Saliency-driven Mixup Augmentation.

Saliency-guided methods prioritize discriminative regions during mixing to preserve semantic integrity. SaliencyMix [53] and Attentive-CutMix [57] combine the most informative regions from source and target images. PuzzleMix [31] redistributes patches based on saliency and local statistics, and Co-Mixup [32] extends this formulation to multi-image mixing with supermodular diversity constraints. SAMix [36] decomposes mixup objectives into locally emphasized and globally constrained terms to enable adaptive mixing. SalfMix [10] generates self-mixed samples by transferring salient regions into less salient areas within the same image. GuidedMixup [30] focuses on critical local features using spectral residual-based saliency. Despite these advances, most saliency-driven methods rely on transferring salient regions across different images, which introduces semantic inconsistencies.

3 Proposed Method

3.1 Overview

The motivation behind the proposed S^2-FracMix is to directly encode self-contained multi-scale saliency-guided augmentation. We preserve object saliency

while promoting structural diversity. Unlike prior works, we take a direct approach for the detection of salient regions via spectral residual method [22], which is then used to guide patch extractions at various scales.

These patches are transformed using rotation and blurring and mixed with the original image at non-salient random positions. Thus, S^2-FracMix explicitly encodes scale-invariant representation learning while preserving semantic integrity. As shown in Algorithm 1 and illustrated in Fig. 3, these multiscale patches are mixed with the original image, ensuring that important information highlighted through saliency is retained, while background or less discriminative areas are altered, to strengthen robust feature learning.

3.2 Self Saliency (S^2) Mixing

Let $\mathcal{D} = \{(I_i, y_i)\}_{i=1}^{N}$ represent the training dataset, where $I_i \in \mathbb{R}^{c\times h\times w}$ is an input image with c channels, height h, and width w, and y_i is its corresponding one-hot encoded label. Self Saliency mixup generates an augmented image $\tilde{I}_i$ (with $\tilde{y}_i = y_i$) as follows. Saliency maps $S_i \in \mathbb{R}^{1\times h\times w}$ are computed to highlight regions critical to the predictions of model as:

$$S_i = f(I_i), \tag{1}$$

where $f(\cdot)$ is the saliency detection method. The saliency maps guide the selection and transformation of patches. Patches are extracted from the input image I_i at 2 patch scales $\mathcal{P} = \{(h/2, w/2), (h/4, w/4)\}$. For a patch $P_k \in \mathcal{P}$, the top left position (x_k, y_k) is sampled uniformly $x_k \sim \text{Uniform}(0, w-w_k)$, and $y_k \sim \text{Uniform}(0, h-h_k)$. The patch is then extracted as: $P_k = I_i[x_k : x_k+w_k, y_k : y_k+h_k]$. Let S_k represent the saliency mask for the corresponding patch, defined as: $S_k = S_i[x_k : x_k + w_k, y_k : y_k + h_k]$. S_k is normalized to $[0, 1]$ and a random threshold $t \sim \text{Uniform}(0.5, 1.0)$ is applied to obtain the binarized mask $\tilde{S}_k$. The patch is accepted only if the salient area ratio meets $\sum \tilde{S}_k/(h_k w_k) \geq (1 - t)$; otherwise it is rejected. Each accepted patch is first blended with a randomly selected fractal image F as described in Section 3.3 to obtain P_k^f. The transformation $T_k(P_k^f, \tilde{S}_k)$ is applied as

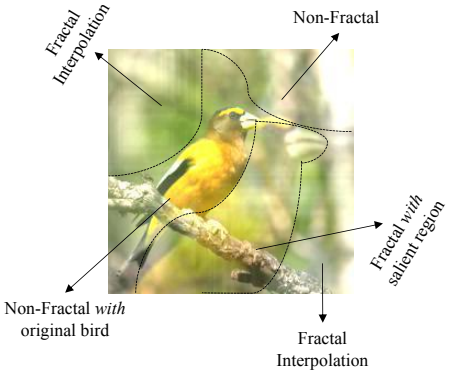

Fig. 3: Augmentation samples of S^2-FracMix are intricate composition of fractal and non-fractal patterns interpolated at multiple scales, all while retaining original image semantics.

$$T_k = R(P_k^f, \theta) \cdot \tilde{S}_k + B(P_k^f) \cdot (1 - \tilde{S}_k), \tag{2}$$

where $R(P_k^f, \theta)$ applies a random rotation $\theta \sim \text{Uniform}(-30°, 30°)$ and $B(\cdot)$ applies Gaussian blurring. The transformed patch is resized back to the original image dimensions.

$$R_k = \text{Resize}(T_k, (h, w)) \tag{3}$$

Algorithm 1: S^2-FracMix Algorithm

Require: $\mathcal{I}_b = \{I_i, \mathbf{y}_i\}_{i=1}^{b}$, $\mathcal{P}$, $\mathcal{F}$, λ
Ensure: $\widetilde{\mathcal{I}}_b = \{\tilde{I}_i, \mathbf{y}_i\}_{i=1}^{b}$: Augmented batch with labels
foreach *image* $(I_i, y_i) \in \mathcal{I}_b$ **do**
  $S_i \leftarrow \texttt{SaliencyMap}(I_i)$, $P_m \leftarrow \texttt{zeros}(w, h)$, $n_k \leftarrow 0$
  foreach *patch scale* $\mathbf{p}_k(h_k, w_k) \in \mathcal{P}$ **do**
    $x_k \leftarrow \texttt{uniform}(1, h - h_k)$, $y_k \leftarrow \texttt{uniform}(1, w - w_k)$
    Image patch: $P_k \leftarrow I_i[x_k : x_k + h_k,\ y_k : y_k + w_k]$
    Saliency patch: $S_k \leftarrow S_i[x_k : x_k + h_k,\ y_k : y_k + w_k]$
    $S_k \leftarrow (S_k - \texttt{min}(S_k))/(\texttt{max}(S_k) - \texttt{min}(S_k))$
    $t \leftarrow \texttt{uniform}(0.50, 1.0)$, $\tilde{S}_k \leftarrow \texttt{threshold}(S_k, t)$
    if *Sum* $(\tilde{S}_k)/(w_k h_k) < (1 - t)$ **then**
      `reject` P_k
    end
    else
      $F \leftarrow \texttt{uniform}(\mathcal{F})$, $\theta \leftarrow \texttt{uniform}(-30, 30)$
      Fractal blending: $P_k^f = \lambda F + (1 - \lambda) P_k$
      Transform: $T_k \leftarrow \mathrm{R}(P_k^f, \theta) \cdot \tilde{S}_k + \mathrm{B}(P_k^f) \cdot (1 - \tilde{S}_k)$
      Resize patch: $R_k \leftarrow \texttt{Resize}(T_k, (h, w))$
      Mixed patch: $P_m \leftarrow P_m + R_k$
      $n_k \leftarrow n_k + 1$
    end
  end
  $\alpha \leftarrow \texttt{uniform}(0, 1)$
  Augment image: $\tilde{I}_i \leftarrow \alpha I_i + \frac{1-\alpha}{n_k} P_m$
  $\widetilde{\mathcal{I}}_b \leftarrow \widetilde{\mathcal{I}}_b \cup \{(\tilde{I}_i, \mathbf{y}_i)\}$
end
return $\widetilde{\mathcal{I}}_b$

These resized patches are mixed into the original image using a weighted sum

$$\tilde{I}_i = \alpha I_i + (1 - \alpha) \sum_{k=1}^{n_p} \lambda_k R_k, \tag{4}$$

where $\lambda_k = 1/n_p$ are mixing weights with $\sum_{k=1}^{n_p} \lambda_k = 1$, n_p is the number of accepted patches, and $0 \leq \alpha \leq 1$ is a uniform random variable. Thus, S^2 drives learning models to handle a range of spatial transformations without complex mask optimization procedures as used in previous methods such as SaliencyMix [53], PuzzleMix [31], and Co-Mixup [32]. As a result, our method remains computationally efficient yet highly diverse, synthesizing effective mixing modes that preserve semantic cues.

3.3 FracMix & High-level Mixing Modes

In traditional methods, fractals are blended with the whole input image. In contrast, we propose fractal blending only inside the salient patches selected by the Self-Saliency ($\mathbf{S^2}$) procedure. We maintain a fixed library of $\mathcal{F}$ of fractal patterns constructed before training for each accepted patch P_k, we sample $F \leftarrow \texttt{uniform}(\mathcal{F})$ and blended it with P_k using factor λ as

$$P_k^f = \lambda F + (1 - \lambda) P_k, \tag{5}$$

In practice, we select λ empirically via a small sweep on a held-out validation split (see Sec. 6, Fig. 5), and use λ =0.20 as the default setting in all experiments. The resulting fractal-enriched patch P_k^f then undergoes the selective transformation T_k (Eq. 2) followed by resizing to full resolution (Eq. 3) to obtain R_k, and mixing in Eq. (4).

It is notable that existing approaches generally employ a single mixing strategy throughout training, such as Mixup [14], CutMix [64], ResizeMix [51], PuzzleMix [31], and GuidedMixup [30]. We empirically observed that restricting the model to only one mode of low-level mixing limits the diversity of supervisory signals, compromising model performance. Therefore, we propose to incorporate *multiple low-level modes of mixing* within the training pipeline. Specifically, we allow mixing computationally efficient mechanisms of Mixup [14], CutMix [64], and ResizeMix [51] jointly at high-level with our S^2-FracMix technique to further the performance gain. The process behind this mode mixing randomly selects the mechanisms to encourage complementary regularization effects and expose the model to a richer variety of mixed inputs, ultimately improving model generalization. We thoroughly analyze this in Sec. 6.

4 Theoretical Analysis

Our self-saliency mixing forces high-saliency feature reconstruction in low-saliency context. This can be viewed as effectively adding a structured auxiliary loss that encourages the minimal sufficient statistic to be distributed across the entire image rather than concentrated in a few dominant regions, which improves robustness to local perturbations. Below, we provide a theoretical analysis of this property of our method.

Let $(x, y) \sim \mathcal{D}$ denote a training sample drawn from the data distribution, where $x \in [0, 1]^{C \times H \times W}$ is an input image with C channels and spatial resolution $H \times W$, and y is its ground-truth label. Let h_ω denote the predictor, and let $\ell(\cdot, \cdot)$ denote the training loss. For notational convenience, we define

$$g(z) := \ell(h_\omega(z), y), \tag{6}$$

i.e., g is the loss viewed as a function of the input z while keeping the label y fixed. We denote by $\theta \sim \mathit{\Pi}$ the random augmentation parameters, where θ collects all randomness in the augmentation process (sampled patch scale and

location, saliency mask, rotation angle, fractal sample, and the mixing coefficient α). We define the augmentation operator by $A_\theta(\cdot)$, and augmented input by

$$x' := A_\theta(x). \tag{7}$$

To analyze its effect, we decompose the augmentation into a structured transformation term and an additive localized perturbation term:

$$A_\theta(x) = T_\theta(x) + \Delta_\theta(x). \tag{8}$$

Here, $T_\theta(x) \in [0,1]^{C\times H\times W}$ denotes the structured transform component, and $\Delta_\theta(x) \in \mathbb{R}^{C\times H\times W}$ denotes the additive perturbation component. In our setting, the structured transform is saliency-guided and corresponds to the selective rotation/blur step (Eq. 2) aggregated over the accepted patches (Algorithm 1). Let $S(x) \in [0,1]^{1\times H\times W}$ be a per-pixel saliency map, and let $M_\theta \in \{0,1\}^{1\times H\times W}$ be an effective binary mask induced by the accepted patch-level saliency masks after resizing to full resolution. Let $R(x,\varphi)$ denote a rotation of x by angle φ, where φ is sampled as part of θ, and let $B(\cdot)$ denote Gaussian blurring. With the same mixing coefficient $\alpha \sim \text{Uniform}(0,1)$ as in Eq. (4), we define

$$T_\theta(x) = \alpha x + (1-\alpha)\Big(R(x,\varphi)\odot M_\theta + B(x)\odot(1-M_\theta)\Big), \tag{9}$$

where $\odot$ denotes element-wise multiplication. We further model the localized fractal perturbation consistent with Sec. 3.3 fractals are injected only through the selected region, using the same blending factor λ as Eq. (5):

$$\Delta_\theta(x) = (1-\alpha)\lambda\,(M_\theta \odot F), \tag{10}$$

where $F \in \mathbb{R}^{C\times H\times W}$ denotes the (full-resolution) fractal pattern corresponding to the sampled fractal(s) after resizing/aggregation. Then, we analyze the augmentation under the vicinal risk minimization (VRM) [6] objective:

$$\text{VRM}(h_\omega) := \mathbb{E}_{(x,y)\sim\mathcal{D}}\,\mathbb{E}_{\theta\sim\Pi}\big[\ell(h_\omega(A_\theta(x)),y)\big] = \mathbb{E}_{x,y}\mathbb{E}_\theta\big[g(T_\theta(x)+\Delta_\theta(x))\big]. \tag{11}$$

For brevity, let

$$x_T := T_\theta(x), \qquad \Delta_\theta := \Delta_\theta(x), \tag{12}$$

so that $x' = x_T + \Delta_\theta$. We make two assumptions. The fractal perturbation is zero-mean with covariance Σ_f, i.e., $\mathbb{E}[F] = 0$ and $\text{Cov}(F) = \Sigma_f$. Since $\Delta_\theta = (1-\alpha)\lambda(M_\theta \odot F)$ and α is sampled independently, this implies $\mathbb{E}_\theta[\Delta_\theta] = 0$. Label consistency under the structured transform T_θ: applying T_θ does not alter the semantic label y. This assumption is used only to justify evaluating the same target y after the transformation. We now perform a second-order Taylor expansion of g around x_T. Since $x' = x_T + \Delta_\theta$, we have

$$g(x_T+\Delta_\theta) \approx g(x_T) + \nabla g(x_T)^\top \Delta_\theta + \frac{1}{2}\Delta_\theta^\top H_g(x_T)\Delta_\theta, \tag{13}$$

where $\nabla g(x_T)$ and $H_g(x_T)$ denote the gradient and Hessian of g with respect to the input, evaluated at x_T. Taking expectation over θ, the first-order term vanishes due to the zero-mean perturbation assumption:

$$\mathbb{E}_\theta[\nabla g(x_T)^\top \Delta_\theta] = \nabla g(x_T)^\top \mathbb{E}_\theta[\Delta_\theta] = 0. \tag{14}$$

Therefore, the expected loss under augmentation is approximately

$$\mathbb{E}_\theta[g(x_T + \Delta_\theta)] \approx \mathbb{E}_\theta[g(x_T)] + \frac{1}{2}\mathbb{E}_\theta\left[\Delta_\theta^\top H_g(x_T)\Delta_\theta\right]. \tag{15}$$

Taking expectation over $(x, y) \sim \mathcal{D}$ yields

$$\mathrm{VRM}(h_\omega) \approx \mathbb{E}_{x,y}\mathbb{E}_\theta[g(x_T)] + \frac{1}{2}\mathbb{E}_{x,y}\mathbb{E}_\theta\left[\Delta_\theta^\top H_g(x_T)\Delta_\theta\right]. \tag{16}$$

To interpret the second term, we rewrite the quadratic form in trace form using the identity $\Delta^\top H \Delta = \mathrm{tr}(H\Delta\Delta^\top)$. This gives

$$\mathbb{E}_\theta\left[\Delta_\theta^\top H_g(x_T)\Delta_\theta\right] = \mathrm{tr}\big(H_g(x_T)\,\mathbb{E}_\theta[\Delta_\theta\Delta_\theta^\top]\big). \tag{17}$$

We now compute second moment of the perturbation. Since $\Delta_\theta = (1-\alpha)\lambda(M_\theta \odot F)$ and $\alpha \sim \mathrm{Uniform}(0,1)$ is independent of (M_θ, F), we have $\mathbb{E}[(1-\alpha)^2] = \frac{1}{3}$ and thus

$$\mathbb{E}_\theta[\Delta_\theta\Delta_\theta^\top] = \frac{\lambda^2}{3}\,\mathbb{E}_\theta\left[M_\theta\,\Sigma_f\,M_\theta^\top\right]. \tag{18}$$

We define mask-dependent covariance as localized perturbation covariance:

$$\Sigma_{\mathrm{loc}}(x) := \mathbb{E}_\theta\left[M_\theta\,\Sigma_f\,M_\theta^\top\right]. \tag{19}$$

Substituting Eq. (18) and Eq. (19) into Eq. (17) yields

$$\mathbb{E}_\theta\left[\Delta_\theta^\top H_g(x_T)\Delta_\theta\right] = \frac{\lambda^2}{3}\,\mathrm{tr}(H_g(x_T)\Sigma_{\mathrm{loc}}(x)). \tag{20}$$

Hence, the vicinal risk admits the following second-order approximation:

$$\mathrm{VRM}(h_\omega) \approx \underbrace{\mathbb{E}_{x,y}\mathbb{E}_\theta\left[\ell(h_\omega(T_\theta(x)), y)\right]}_{\text{invariance term}} + \underbrace{\frac{\lambda^2}{6}\,\mathbb{E}_x[\mathrm{tr}(H_g(T_\theta(x))\Sigma_{\mathrm{loc}}(x))]}_{\text{saliency-local stability penalty}}. \tag{21}$$

The first term enforces invariance to the structured saliency-guided transform T_θ, while the second penalizes curvature-weighted sensitivity to fractal perturbations restricted by $\Sigma_{\mathrm{loc}}(x)$, concentrating the regularization on accepted salient regions (Algorithm 1). This characterization yields two predictions borne out by our experiments. The penalty grows quadratically in λ, so accuracy should first improve and then degrade as λ increases, matching Figure 5(b) where performance peaks at λ=0.20 and drops at λ=0.50. Moreover, since the penalty suppresses sensitivity to localized input perturbations, the largest gains should appear under corruption and adversarial noise rather than on clean data, matching Table 4 where our corruption gain (**2.40%**) and FGSM error reduction (**3.14%**) over AdAutoMix both exceed the clean gain (**1.19%**). The choice between local and global injection corresponds to different $\Sigma_{\mathrm{loc}}(x)$, and we resolve that comparison empirically in Table 6.

5 Experiments and Results

We benchmark our proposed S^2-FracMix against several recent competitive mixup approaches, including Mixup [65], CutMix [64], ManifoldMix [55], FMix [16], ResizeMix [51], SaliencyMix [53], PuzzleMix [31], AutoMix [42], and AdAutoMix [49]. Additionally, we report the computational overhead of our method and compare it with timings reported in AdAutoMix [30]. To demonstrate the generalizability of our method, we conduct experiments from small-scale to large-scale backbones including `ResNet18` [17], `ResNet34` [17], `ResNet50` [17], `ResNeXt50` [62], transformer-based architectures including `Swin Transformer` [39] and `ConvNeXt` [40], and contrastive method `MoCo v2` [8] and `SimSiam` [9]. All experiments are implemented using open-source OpenMixup.

Note that, we follow the standard evaluation practices and protocols mentioned in AdAutoMix [49] for fair comparison. Hyperparameter configurations and brief *implementation guidelines*, with *dataset statistics* are also provided in Appendix A and *detailed settings* provided in Appendix B. We show that our proposed S^2-FracMix not only improves classification performance across both general- and fine-grained tasks, but also enhances robustness to distributional shifts, such as background corruption [19], data scarcity, transfer learning, calibration, contrastive learning methods, object detection while maintaining minimal computational overhead.

5.1 General Classification

We compare the performance of S^2-FracMix in Tab. 1, our approach achieves SOTA performance on CNNs and ViTs, consistently outperforming existing augmentation strategies such as AdAutoMix [49], AutoMix, ResizeMix, and PuzzleMix. Notably, S^2-FracMix surpasses AdAutoMix, the existing best performing method [49], by approximately **0.42%** and **0.69%** in Top-1 accuracy on CIFAR-100. The trend is similar across different backbones from small-scale to large-scale backbones. In terms of Tiny-ImageNet and ImageNet-1K, the improvement gap is even more pronounced, underlining S^2-FracMix capacity to

Table 1: Top-1 performance (%)↑ of mixup methods on CIFAR-100, Tiny-ImageNet and ImageNet-1K. The results of previous mixup SOTA methods are taken from AdAutoMix [49]. Res18, ResXt50 CNext-T and Res34 refers to ResNet18, ResNeXt50, ConvNeXt-T and ResNet34. Also, ViT-B results are taken from [3].

	CIFAR-100		CIFAR-100		Tiny-ImageNet		ImageNet-1K			
Method	**Res18**	**ResXt50**	**Swin-T**	**CNeXt-T**	**Res18**	**ResXt50**	**Res18**	**Res34**	**Res50**	**ViT-B**
Vanilla	78.04	81.09	78.41	78.70	61.68	65.04	70.04	73.85	76.83	76.7
MixUp	79.12	82.10	76.78	81.13	63.86	66.36	69.98	73.97	77.12	80.8
CutMix	78.17	81.67	80.64	82.46	65.53	66.47	68.95	73.58	77.17	79.9
SaliencyMix	79.12	81.53	80.40	82.82	64.60	66.55	69.16	73.56	77.14	–
FMix	79.69	81.90	80.72	81.79	63.47	65.08	69.96	74.08	77.19	–
PuzzleMix	81.13	82.85	80.33	82.29	65.81	67.83	70.12	74.26	77.54	–
ResizeMix	80.01	81.82	80.16	82.53	63.74	65.87	69.50	73.88	77.42	–
AutoMix	82.04	83.64	82.67	83.30	67.33	70.72	70.50	74.52	77.91	–
AdAutoMix	82.32	84.22	84.33	83.54	69.19	72.89	70.86	74.82	78.04	–
S^2-FracMix	**82.74**	**84.91**	**85.35**	**84.41**	**70.38**	**74.27**	**71.37**	**75.34**	**78.54**	**81.2**

Table 2: Accuracy (%) ↑ of mixup methods on Caltech Birds-200, FGVC-Aircrafts and Stanford-Cars.

Method	Caltech Birds-200		FGVC-Aircrafts		Stanford-Cars	
	ResNet18	ResNet50	ResNet18	ResNeXt50	ResNet18	ResNeXt50
Vanilla	77.68	82.38	80.23	85.10	86.32	90.15
MixUp	78.39	82.98	79.52	85.18	86.27	90.81
CutMix	78.40	83.17	78.84	84.55	87.48	91.22
ManifoldMix	79.76	83.76	80.68	86.60	85.88	90.20
SaliencyMix	77.95	81.71	80.02	84.31	86.48	90.60
FMix	77.28	83.34	79.36	86.23	87.55	90.90
PuzzleMix	78.63	83.83	80.76	86.23	87.78	91.29
ResizeMix	78.50	83.41	78.10	84.08	88.17	91.36
AutoMix	79.87	83.88	81.37	86.72	88.89	91.38
AdAutoMix	80.88	84.57	81.73	87.16	89.19	91.59
S^2-FracMix	**81.84**	**85.73**	**82.81**	**88.34**	**90.56**	**92.86**

Table 3: Top-1 accuracy (%) of ResNet-50 (R50) and Vision Transformer (ViT) backbones on CUB-200 and Stanford Cars datasets.

Backbone	Dataset	Vanilla	MixUp	CutMix	PuzzleMix	AutoMix	AdAutoMix	S^2-FracMix
R50	CUB-200	81.76	82.79	81.67	82.59	82.93	83.36	**84.42**
	Stanford Cars	88.88	89.45	88.99	89.37	88.71	89.65	**90.85**
ViT-B	CUB-200	88.0	88.75	87.76	88.23	88.91	88.76	**89.84**
	Stanford Cars	91.31	91.36	91.53	91.83	92.51	91.38	**92.86**

capture rich discriminative features. These results demonstrate that S^2-FracMix not only addresses the challenges inherent in CIFAR-100, Tiny-ImageNet, and ImageNet-1K but also establishes a new SOTA among mixup based augmentation methods for enhancing generalization performance.

5.2 Fine-Grained Visual Classification

In fine-grained classification, we follow the same training protocols established in AdAutoMix [49] and also previous results are taken from same paper. As reported in Tab. 2, S^2-FracMix consistently achieves the highest Top-1 accuracy across different architectures and datasets. On Caltech Birds-200, S^2-FracMix improves over the AdAutoMix by **+0.96%** on `ResNet-18` and **+1.16%** on `ResNet-50`. On FGVC-Aircrafts, it achieves gains of **+1.08%** and **+1.18%** on `ResNet-18` and `ResNeXt-50`, respectively. On Stanford-Cars, improvements of **+1.37%** and **+1.27%** are observed with `ResNet-18` and `ResNeXt-50`. These results establish the effectiveness of S^2-FracMix across fine-grained categorization tasks.

5.3 Transfer Learning

We further evaluate the transferability of features learned by S^2-FracMix on downstream transfer learning classification tasks, as presented in Table 3. We utilized two pre-trained deep models including `ResNet-50` and `ViT-B`. Both models are pretrained on ImageNet-1K and fine-tuned on Caltech Birds-200 and Stanford-Cars for classification using S^2-FracMix. Compared to AdAutoMix [49], the strongest existing method, S^2-FracMix achieves consistent gains. In Table 3 on Caltech Birds-200, S^2-FracMix reaches a Top-1 performances of

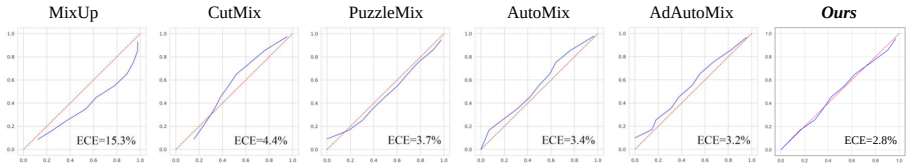

Fig. 4: Calibration plots of S^2-FracMix on CIFAR-100 using `ResNet18`.

84.42%, 89.84%, outperforming the AdAutoMix by **1.06%, 1.08%**. On the Stanford-Cars, it achieves **90.85%, 92.86%**, exceeding the baseline by **1.20%, 1.48%**. These results demonstrate that S^2-FracMix improves fine-tuning performance over baseline and recent SOTA methods across different datasets.

5.4 Robustness

Following the same protocols as used by AdAutoMix [49], we carried out robustness evaluation experiments under common corruptions on CIFAR100-C [18] dataset as shown in Tab. 4. We compared our S^2-FracMix with widely used mixup approaches, including CutMix, FMix, PuzzleMix, AutoMix, and AdAutoMix [49]. As shown in Tab. 4, S^2-FracMix demonstrated the best performance on both clean and corrupted samples, achieving relative gains of **1.19%** and **2.4%** in classification accuracy over AdAutoMix. The robustness improvement of **3.14%** is achieved compared to AdAutoMix.

Table 4: Top-1 accuracy and FGSM error of `ResNet-18` with other methods.

Method	Clean Acc(%)↑	Corruption Acc(%)↑	FGSM Error(%)↓
CutMix	79.45	46.66	88.24
FMix	78.91	50.58	88.35
PuzzleMix	79.96	51.04	80.52
AutoMix	80.02	50.75	82.67
AdAutoMix	81.55	51.44	75.66
S^2-FracMix	**82.74**	**53.84**	**72.52**

5.5 Calibration

Deep Neural Networks often exhibit overconfidence in their predictions during image classification tasks, which can lead to poor calibration. To quantitatively assess calibration performance, we measure the Expected Calibration Error (ECE) across different mixup methods on the CIFAR-100 dataset. Previous results are taken from AdAutoMix [49]. As visbible in Fig. 4, our S^2-FracMix attains the lowest ECE **2.8%** surpassing recent SOTA methods and second best method is AdAutoMix [49].

6 Ablation Study and Analysis

As noted in Sec. 3, our technique allows multiple modes. Here, we analyze the effects of those modes on performance. More results are mentioned in Appendix C.

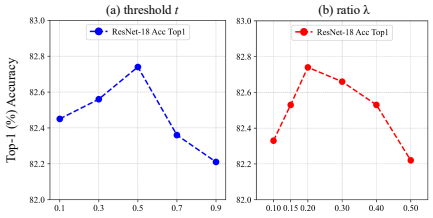

Fig. 5: Ablation of hyperparameters t threshold and λ for fractal mixing of S^2-FracMix on CIFAR100.

Table 5: Ablation study of different high-level mixing strategies on CIFAR-100.

CIFAR-100				Accuracy (%)	
S^2-FracMix	M_m	M_c	M_r	Res18	ResXt50
–	–	–	–	78.04	81.09
✓	–	–	–	81.73	82.22
–	✓	–	–	79.12	82.10
–	–	✓	–	78.17	81.67
–	–	–	✓	80.01	81.82
✓	✓	✓	–	82.24	82.32
✓	✓	–	✓	82.46	82.89
✓	–	✓	✓	82.58	83.52
✓	✓	✓	✓	**82.74**	**84.91**
M_f	✓	✓	✓	80.24	82.27

Inclusion of Simple Modes. Tab. 5 presents `Res18` and `ResXt50` results on CIFAR-100. In the table, M_m denotes Mixup [14], M_c represents CutMix [64], M_r presents ResizeMix [51] and M_f illustrates FMix [16]. We start with our S^2-FracMix, which offers two main improvements: i) it generates multi-scale features. ii) saliency-driven patch transformations in a more principled and diverse manner. The introduction of the S^2-FracMix leads to a significant gain of 3.69% and 1.13% in terms of performance over the base model, highlighting the impact of self-mixing compared to the individual performance of other modes M_m, M_c and M_r. To enable comparison to FMix [16], while retaining the same training and implementation details, we use the mixing mode "M_f+M_m+M_c+M_r". Clearly, this degrades the overall performance, which highlights the efficacy of S^2-FracMix in the mixed mode.

Exclusion of Simple Modes. In our proposed high-level mixing we do not select methods such as PuzzleMix [31], Co-Mixup [32], and GuidedMixup [30], which demonstrate good performance but incur high computational overhead. As mentioned in Tab. 5, the best combination is "S^2-FracMix+M_m+M_c+M_r" and the reason is that M_m complements S^2 in terms of global inter-image variations, while M_c introduces local inter-image diversity. M_r introduces down-scaled inter-image variations which are missing in the other modes. Thus, the selected set of modes complement each other well to generate a diverse set of augmentations.

Motivation behind High-level Mixing. Four crucial objectives of the current augmentation methods include: *scale-invariance*, *inter-image diversity*, *spatial variability*, and *resolution robustness*. Previous methods address these challenges in isolation. With high-level mixing, our objective is to jointly tackles all four objectives while maintaining low computational overhead. Replacing S^2-FracMix with M_f (as used by [38]) while keeping all remaining settings same in Tab. 5, leads to significant performance reduction. This verifies that our high-level mixing achieves the intended objective through correct composition of modes.

Global Fractal vs Local Fractal. In Tab. 6, we analyze design choices for fractal addition on Stanford-Cars. We compare S^2-FracMix with saliency-weighted

Table 6: Ablation study of key components of FracMix in terms of global and local fractals on Stanford-Cars. *sal* denotes saliency and *weight* denotes weighting.

Method	Top-1 (%)
Baseline	85.52
S^2-FracMix (*w/o sal weight*)	91.87
S^2-FracMix (*global fractal*)	92.27
S^2-FracMix (*local*)	**92.78**

Table 7: Top-1 (%) performance ablation of saliency vs. fractal mixing on CIFAR-100 with ResNet-18 and ResNeXt-50.

Method	Res18	ResXt50
Baseline	78.04	81.09
Saliency (A+B)	79.12	81.53
Saliency S^2 (A+A)	79.54	81.92
FracMix (ours)	81.73	82.22
S^2-FracMix	**82.74**	**84.91**

fractal addition applied only to salient regions (Eq. 5) against a global variant that adds fractals to the entire image, $I_i = \lambda F + (1 - \lambda)I_i$. We further study S^2-FracMix without saliency weighting by modifying (Eq. 2) to use an equal mixture of region and background transformations, $T_k = 0.50\, R(P_k^f, \theta) + 0.50\, B(P_k^f)$. These comparisons isolate the benefit of focusing fractal perturbations on salient content rather than applying them uniformly.

Effect of Saliency and Fractal Mixing. We also study how the mixing strategy influences CIFAR-100 accuracy across Res18 and ResXt-50 backbones. We compare a baseline with saliency-guided mixing, where Saliency (A+B) uses the saliency map from the source image A to guide mixing with a target image B, and Saliency S^2 (A+A) uses self-saliency from both inputs presented in Tab. 7. While saliency improves over the baseline on both backbones, our fractal-based *FracMix* yields a larger gain, and the full method S^2-FracMix achieves the best performance overall. We attribute this synergy to the fact that salient regions harbor the most discriminative semantic features; by injecting self-similar fractal patterns specifically into these regions, we force the model to learn robust features invariant to high-frequency perturbations where it matters most, rather than diluting regularization energy on non-discriminative background pixels.

Hyperparameters Ablation. In the S^2-FracMix, there are two main hyperparameters namely saliency threshold t and λ. In order to achieve good performance both parameters should be properly configured. Firstly, we train the `ResNet18` for 200 epochs via our S^2-FracMix. The performance of `ResNet18` with t=0.5 is shown in Fig. 5 (a). In addition, the fractal mixing gives best performance at λ = 0.20 in Fig. 5 (b). However, by increasing the λ to 0.50, we observe that the classification accuracy is slightly decreased to 82.22%, and when we set t=0.9 the performance degrades to almost the same level. This implies that the two parameters are equally capable of controlling the performance of S^2-FracMix.

Object Localization. For this analysis, we visually analyze the models trained with S^2-FracMix and SOTA methods. As evidenced in Fig. 6 (top row), our proposed S^2-FracMix produces a contiguous, high-intensity CAM region that consistently highlights the main region, indicating stronger object retention and clearer attention boundaries.

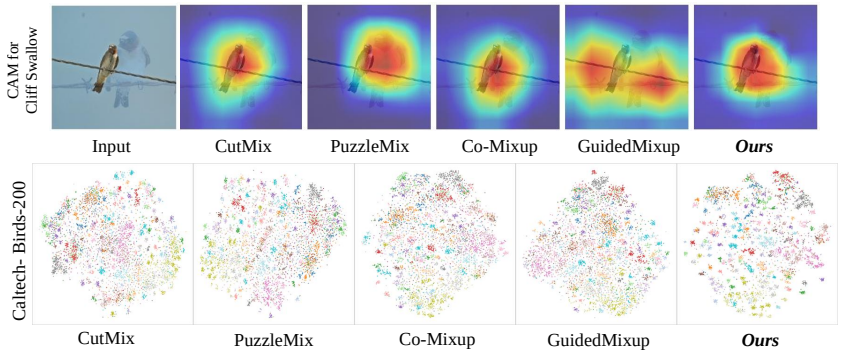

Fig. 6: (Top) Grad-CAM [52] visualization on augmented images and (bottom) t-SNE visualization of `ResNet18` trained from scratch.

Feature Representation. Finally, we compare the trained models by visualizing the feature representation of S^2-FracMix and SOTA methods in Fig. 6 (bottom row). Closely observing t-SNE [43], it can be seen that images of the same class cluster together for our method, representing better learning. Noticeably, S^2-FracMix exhibits distinct clusters with well-defined margins between classes, suggesting that the model consistently learns discriminative features specific to each class.

7 Conclusion

We introduce S^2-FracMix data augmentation technique to improve the performance of deep visual models. The method comprises two main components. In the S^2 mixing component, patches of varying sizes are extracted from an input image while utilizing saliency information. Different transformations are applied to these patches and they are seamlessly integrated back into the same image. In the *FracMix* component, self-similarity fractals are also blended into these salient patches. We also introduce the notion of high-level mixing of multiple low-level mixing modes to enhance diversity among the augmented samples. Experiments are performed on coarse and fine-grained classification, robustness against corruption, few-shot learning, and transfer learning. The proposed S^2-FracMix has demonstrated improved results compared to the existing state-of-the-art methods. The potential limitations of our approach are also discussed in Appendix D.

8 Acknowledgement

Naveed Akhtar is a recipient of the Australian Research Council Discovery Early Career Researcher Award (project # DE230101058) funded by the Australian Government. Khawar Islam is also supported by project # DE230101058. This research was also supported by The University of Melbourne's Research Computing Services and the Petascale Campus Initiative.

References

1. Antoniou, A., Storkey, A., Edwards, H.: Data augmentation generative adversarial networks. arXiv preprint arXiv:1711.04340 (2017)
2. Baek, K., Bang, D., Shim, H.: Gridmix: Strong regularization through local context mapping. Pattern Recognition **109**, 107594 (2021)
3. Bai, J., Yuan, L., Xia, S.T., Yan, S., Li, Z., Liu, W.: Improving vision transformers by revisiting high-frequency components. In: European Conference on Computer Vision. pp. 1–18. Springer (2022)
4. Cao, C., Zhou, F., Dai, Y., Wang, J., Zhang, K.: A survey of mix-based data augmentation: Taxonomy, methods, applications, and explainability. ACM Computing Surveys **57**(2), 1–38 (2024)
5. Carratino, L., Cissé, M., Jenatton, R., Vert, J.P.: On mixup regularization. Journal of Machine Learning Research **23**(325), 1–31 (2022)
6. Chapelle, O., Weston, J., Bottou, L., Vapnik, V.: Vicinal risk minimization. Advances in neural information processing systems **13** (2000)
7. Chen, J.N., Sun, S., He, J., Torr, P.H., Yuille, A., Bai, S.: Transmix: Attend to mix for vision transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12135–12144 (2022)
8. Chen, X., Fan, H., Girshick, R., He, K.: Improved baselines with momentum contrastive learning. arXiv preprint arXiv:2003.04297 (2020)
9. Chen, X., He, K.: Exploring simple siamese representation learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 15750–15758 (2021)
10. Choi, J., Lee, C., Lee, D., Jung, H.: Salfmix: a novel single image-based data augmentation technique using a saliency map. Sensors **21**(24), 8444 (2021)
11. Chrabaszcz, P., Loshchilov, I., Hutter, F.: A downsampled variant of imagenet as an alternative to the cifar datasets. arXiv preprint arXiv:1707.08819 (2017)
12. Deng, J., Dong, W., Socher, R., Li, L., Li, K., Li, F.: Imagenet: A large-scale hierarchical image database (2009)
13. Ghiasi, G., Cui, Y., Srinivas, A., Qian, R., Lin, T.Y., Cubuk, E.D., Le, Q.V., Zoph, B.: Simple copy-paste is a strong data augmentation method for instance segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2918–2928 (2021)
14. Guo, H., Mao, Y., Zhang, R.: Mixup as locally linear out-of-manifold regularization. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 3714–3722 (2019)
15. Han, J., Petersson, L., Li, H., Reid, I.: Cropmix: Sampling a rich input distribution via multi-scale cropping. arXiv preprint arXiv:2205.15955 (2022)
16. Harris, E., Marcu, A., Painter, M., Niranjan, M., Prügel-Bennett, A., Hare, J.: Fmix: Enhancing mixed sample data augmentation. arXiv preprint arXiv:2002.12047 (2020)
17. 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)
18. Hendrycks, D., Dietterich, T.: Benchmarking neural network robustness to common corruptions and perturbations. Proceedings of the International Conference on Learning Representations (2019)
19. Hendrycks, D., Mu, N., Cubuk, E.D., Zoph, B., Gilmer, J., Lakshminarayanan, B.: Augmix: A simple data processing method to improve robustness and uncertainty. In: International Conference on Learning Representations (ICLR) (2020)

20. Hendrycks, D., Zou, A., Mazeika, M., Tang, L., Li, B., Song, D., Steinhardt, J.: Pixmix: Dreamlike pictures comprehensively improve safety measures. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16783–16792 (2022)
21. Hong, M., Choi, J., Kim, G.: Stylemix: Separating content and style for enhanced data augmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 14862–14870 (2021)
22. Hou, X., Zhang, L.: Saliency detection: A spectral residual approach. In: 2007 IEEE Conference on computer vision and pattern recognition. pp. 1–8. Ieee (2007)
23. Huang, S., Wang, X., Tao, D.: Snapmix: Semantically proportional mixing for augmenting fine-grained data. In: Proceedings of the AAAI Conference on Artificial Intelligence (2021)
24. Huang, Z., Bao, X., Zhang, N., Zhang, Q., Tu, X., Wu, B., Yang, X.: Ipmix: Label-preserving data augmentation method for training robust classifiers. Advances in Neural Information Processing Systems **36**, 63660–63673 (2023)
25. Islam, K., AKHTAR, N.: Context-guided responsible data augmentation with diffusion models. In: ICLR 2025 Workshop on Navigating and Addressing Data Problems for Foundation Models
26. Islam, K., Zaheer, M.Z., Mahmood, A., Nandakumar, K.: Diffusemix: Label-preserving data augmentation with diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 27621–27630 (2024)
27. Islam, K., Zaheer, M.Z., Mahmood, A., Nandakumar, K., Akhtar, N.: Genmix: Effective data augmentation with generative diffusion model image editing. arXiv preprint arXiv:2412.02366 (2024)
28. Jin, X., Li, S., Jian, S., Yu, K., Wang, H.: Mergemix: A unified augmentation paradigm for visual and multi-modal understanding. arXiv preprint arXiv:2510.23479 (2025)
29. Jin, X., Zhu, H., Li, S., Wang, Z., Liu, Z., Yu, C., Qin, H., Li, S.Z.: A survey on mixup augmentations and beyond. arXiv preprint arXiv:2409.05202 (2024)
30. Kang, M., Kim, S.: Guidedmixup: an efficient mixup strategy guided by saliency maps. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37, pp. 1096–1104 (2023)
31. Kim, J.H., Choo, W., Song, H.O.: Puzzle mix: Exploiting saliency and local statistics for optimal mixup. In: International Conference on Machine Learning. pp. 5275–5285. PMLR (2020)
32. Kim, J.H., Park, J., Hwang, S.: Co-mixup: Saliency-guided joint mixup with supermodular diversity. In: Advances in Neural Information Processing Systems (NeurIPS) (2021)
33. Krause, J., Stark, M., Deng, J., Fei-Fei, L.: 3d object representations for fine-grained categorization. In: 4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13). Sydney, Australia (2013)
34. Krizhevsky, A.: Learning multiple layers of features from tiny images. Master's thesis, University of Tront (2009)
35. Lee, J.H., Zaheer, M.Z., Astrid, M., Lee, S.I.: Smoothmix: a simple yet effective data augmentation to train robust classifiers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops. pp. 756–757 (2020)
36. Li, S., Liu, Z., Wang, Z., Wu, D., Liu, Z., Li, S.Z.: Boosting discriminative visual representation learning with scenario-agnostic mixup. arXiv preprint arXiv:2111.15454 (2021)

37. Li, S., Wang, Z., Liu, Z., Wu, D., Tan, C., Jin, W., Li, S.Z.: Openmixup: A comprehensive mixup benchmark for visual classification (2022)
38. Liu, X., Shen, F., Zhao, J., Nie, C.: Randomix: A mixed sample data augmentation method with multiple mixed modes. Multimedia Tools and Applications **84**(8), 4343–4359 (2025)
39. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10012–10022 (2021)
40. Liu, Z., Mao, H., Wu, C.Y., Feichtenhofer, C., Darrell, T., Xie, S.: A convnet for the 2020s. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11976–11986 (2022)
41. Liu, Z., Li, S., Wang, G., Wu, L., Tan, C., Li, S.Z.: Harnessing hard mixed samples with decoupled regularizer. Advances in Neural Information Processing Systems **36** (2024)
42. Liu, Z., Li, S., Wu, D., Liu, Z., Chen, Z., Wu, L., Li, S.Z.: Automix: Unveiling the power of mixup for stronger classifiers. In: European Conference on Computer Vision. pp. 441–458. Springer (2022)
43. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(11) (2008)
44. Maji, S., Rahtu, E., Kannala, J., Blaschko, M., Vedaldi, A.: Fine-grained visual classification of aircraft. arXiv preprint arXiv:1306.5151 (2013)
45. Parast, A.Y., Azam, B., Akhtar, N.: Ddb: Diffusion driven balancing to address spurious correlations. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 17526–17535 (2025)
46. Parast, A.Y., Hosseini, P., Asadollahzadeh, H., Moakhar, A.S., Azam, B., Feizi, S., Akhtar, N.: Ghost: Hallucination-inducing image generation for multimodal llms. arXiv preprint arXiv:2509.25178 (2025)
47. Parast, A.Y., Islam, K., Won, S., Azam, B., Akhtar, N.: Hsfm: Hard-set-guided feature-space meta-learning for robust classification under spurious correlations. arXiv preprint arXiv:2603.29313 (2026)
48. Pinto, F., Yang, H., Lim, S.N., Torr, P., Dokania, P.: Using mixup as a regularizer can surprisingly improve accuracy & out-of-distribution robustness. Advances in Neural Information Processing Systems **35**, 14608–14622 (2022)
49. Qin, H., Jin, X., Jiang, Y., El-Yacoubi, M., Gao, X.: Adversarial automixup. In: The Twelfth International Conference on Learning Representations. pp. OpenReview, Spotlight (2024)
50. Qin, H., Jin, X., Zhu, H., Liao, H., El-Yacoubi, M.A., Gao, X.: Sumix: Mixup with semantic and uncertain information. In: European Conference on Computer Vision. pp. 70–88. Springer (2025)
51. Qin, J., Fang, J., Zhang, Q., Liu, W., Wang, X., Wang, X.: Resizemix: Mixing data with preserved object information and true labels. arXiv preprint arXiv:2012.11101 (2020)
52. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: visual explanations from deep networks via gradient-based localization. International journal of computer vision **128**, 336–359 (2020)
53. Uddin, A.S., Monira, M.S., Shin, W., Chung, T., Bae, S.H.: Saliencymix: A saliency guided data augmentation strategy for better regularization. In: International Conference on Learning Representations (2020)

54. Venkataramanan, S., Kijak, E., Amsaleg, L., Avrithis, Y.: Alignmixup: Improving representations by interpolating aligned features. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 19174–19183 (2022)
55. 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)
56. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The Caltech-UCSD Birds-200-2011 Dataset. Tech. Rep. CNS-TR-2011-001, California Institute of Technology (2011)
57. Walawalkar, D., Shen, Z., Liu, Z., Savvides, M.: Attentive cutmix: An enhanced data augmentation approach for deep learning based image classification. In: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 3642–3646. IEEE (2020)
58. Wang, Z., Wei, L., Wang, T., Chen, H., Hao, Y., Wang, X., He, X., Tian, Q.: Enhance image classification via inter-class image mixup with diffusion model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17223–17233 (2024)
59. Won, S., Bae, S.H., Kim, S.T.: Effects of mixed sample data augmentation on interpretability of neural networks. Neural Networks **190**, 107611 (2025)
60. Won, S., Kim, H.B., Ahn, Y.H., Lee, H.J., Kim, S.T.: Understanding adversarial robustness of deep neural networks via decision reliance. Image and Vision Computing p. 105743 (2025)
61. Xiao, H., Zheng, W., Zhu, Z., Zhou, J., Lu, J.: Token-label alignment for vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5495–5504 (2023)
62. Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1492–1500 (2017)
63. Yang, L., Li, X., Zhao, B., Song, R., Yang, J.: Recursivemix: Mixed learning with history. Advances in Neural Information Processing Systems **35**, 8427–8440 (2022)
64. Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y.: Cutmix: Regularization strategy to train strong classifiers with localizable features. In: International Conference on Computer Vision (ICCV) (2019)
65. Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: Beyond empirical risk minimization. In: International Conference on Learning Representations (ICLR) (2018)
66. Zhao, L., Liu, T., Peng, X., Metaxas, D.: Maximum-entropy adversarial data augmentation for improved generalization and robustness. Advances in Neural Information Processing Systems **33**, 14435–14447 (2020)
67. Zoph, B., Cubuk, E.D., Ghiasi, G., Lin, T.Y., Shlens, J., Le, Q.V.: Learning data augmentation strategies for object detection. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16. pp. 566–583. Springer (2020)