IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 28 August 2024, accepted 19 September 2024, date of publication 30 September 2024, date of current version 19 December 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3470122

SURVEY

Image Data Augmentation Approaches: A Comprehensive Survey and Future Directions

TEERATH KUMAR[1], ROB BRENNAN[2], (Senior Member, IEEE), ALESSANDRA MILEO[3], AND MALIKA BENDECHACHE[4]

[1]CRT-AI and ADAPT Research Centre, School of Computing, Dublin City University, Dublin 9, D09 V209 Ireland
[2]ADAPT Research Centre, School of Computer Science, University College Dublin, Dublin 4, D04 V1W8 Ireland
[3]INSIGHT and I-Form Research Centre, School of Computing, Dublin City University, Dublin 9, D09 V209 Ireland
[4]ADAPT and Lero Research Centre, School of Computer Science, University of Galway, Galway, H91 TK33 Ireland

Corresponding author: Teerath Kumar (teerath.menghwar2@mail.dcu.ie)

This work was supported by the Science Foundation Ireland through the Science foundation of Ireland (SFI) Centre for Research Training in Artificial intelligence under Grant 18/CRT/6223, through the Insight SFI Research Centre for Data Analytics under Grant SFI/12/RC/2289/*P*_2, through the Lero SFI Centre for Software under Grant 13/RC/2094/*P*_2, and through the ADAPT SFI Research Centre for AI-Driven Digital Content Technology under Grant 13/RC/2106/*P*_2.

ABSTRACT Deep learning algorithms have exhibited impressive performance across various computer vision tasks; however, the challenge of overfitting persists, especially when dealing with limited labeled data. This survey explores the mitigation of the overfitting issue through a comprehensive examination of image data augmentation techniques, which aim to enhance dataset size and diversity by introducing varied samples. The survey exclusively focuses on these techniques, presenting an insightful overview and introducing a novel taxonomy. The discussion encompasses the strengths and limitations of these techniques. Additionally, the paper provides extensive results evaluating the impact of these techniques on prevalent computer vision tasks: image classification, object detection, and semantic segmentation. The survey concludes with an examination of challenges, limitations, and potential future research directions.

INDEX TERMS Computer vision, data augmentation, deep learning, image classification, object detection, segmentation.

I. INTRODUCTION

The proliferation of deep learning models has propelled significant advancements in computer vision (CV) tasks, encompassing domains such as image classification [1], [2], [3], [4], object detection [5], [6], and image segmentation [7], [8], [9], [10]. The effectiveness of these models can be attributed to well-designed deep neural network architectures, substantial computational resources, and the widespread availability of data [11]. Among a wide range of neural networks, Convolutional Neural Networks (CNNs) have demonstrated unparalleled performance in CV tasks. Operating by applying the convolution operation with the input image and kernel, CNNs learn various features of an image. The initial layers focus on low-level features, such as edges and lines, while deeper layers discern more structured and complex features. In recent years, Vision Transformers (ViTs) have also become a mainstream backbone network for CV tasks. ViTs leverage self-attention mechanisms, which enable the model to capture long-range dependencies within images, providing an alternative to CNN-based approaches [30].

Despite their success, deep neural networks are inherently data-intensive and susceptible to the challenge of overfitting [12]. Overfitting arises when a model excels on training data but falters on test data (i.e., unseen data). This issue is exacerbated when task-specific datasets are limited, a scenario often dictated by privacy concerns or the resource-intensive nature of human labeling [11], [13]. Even with extensive datasets like ImageNet [14], overfitting persists as the standard training process tends to overlook less crucial features necessary for generalization [15]. Moreover,

The associate editor coordinating the review of this manuscript and approving it for publication was Jenny Mahoney.

© 2024 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

the accuracy of differnt models is threatened by adversarial attacks [16], [17], [18], [19], [22], [23], where imperceptible perturbations in the input image can mislead the network, causing it to misinterpret critical features.

To tackle these challenges, data augmentation has emerged as a widely adopted strategy, not only in CV tasks but also in diverse data domains, including audio [20], [21]. Ko et al. [21] discuss audio augmentation techniques for improving speech recognition systems, while Nanni et al. [20] present data augmentation approaches for enhancing animal audio classification. Similarly, in the text domain, data augmentation plays a crucial role [24], [25], [26], [27], [28]. Feng et al. [24] introduce Genaug, a data augmentation method for fine-tuning text generators, and Liu et al. [25] conduct a survey on various text data augmentation techniques. Additionally, Shorten et al. [26] provide insights into text data augmentation for deep learning, and Bayer et al. [27] discuss data augmentation techniques specifically for text classification tasks. Lastly, Feng et al. [28] offer a comprehensive survey of data augmentation approaches for natural language processing (NLP). Another aspect of data augmentation, point cloud augmentation techniques [29] have gained attention in computer vision tasks such as 3D object recognition and reconstruction, contributing to the enhancement of model robustness and performance.

In this study, we review the recent works for the challenges mitigation of limited data and overfitting by employing image data augmentation for both CNN and ViT. By presenting the model with diverse views of an image, this technique enhances model generalization and facilitates the extraction of more information from the original dataset. Additionally, in the context of labeling, augmenting samples preserves the original label and extends it to the augmented samples.

Several researchers have worked on review of data augmentation. For instance, Wang et al. [32] explored and compared traditional data augmentation techniques [31], which include foundational techniques such as rotation, flipping, scaling, and cropping, widely used in various computer vision tasks. However, their focus was limited to image classification. In a separate study, Wang et al. [32] reviewed data augmentation approaches in the context of face recognition. Khosla et al. [33] briefly discussed warping and oversampling-based data augmentation without providing a comprehensive taxonomy or thorough evaluation. Shorten et al. [11] presented a detailed survey on image data augmentation but limited the scope to image classification tasks and lacked the inclusion of the latest state-of-the-art (SOTA) augmentation methods. Recently, Yang et al. [34] conducted a survey on data augmentation in computer vision tasks, covering only a few augmentation methods. Another study by Xu et al. [35] proposed a novel taxonomy but did not evaluate the discussed techniques. Additionally, a recent survey [36] briefly discussed current data augmentation, compiled results for classification tasks only. Our paper presents an expanded taxonomy for data augmentation and reviews state-of-the-art techniques. Importantly, none of these surveys demonstrate the effect of data augmentation for ViTs; Most importantly, not a single of these surveys show what the effect of data augmentation is to the ViTs, while we also offer results about the effect of certain data augmentation methods for both CNNs and ViTs.

We exclude Generative Adversarial Networks (GANs) for data augmentation as it is a broad topic that encompasses various techniques and applications. However, it is important to recognize their vital role in this area. GANs and Variational Autoencoders (VAEs) are potential supplements to increase data diversity by generating synthetic data for model generalization. Augmentation methods using GANs provide higher quality and diverse image representations, which are useful for training robust CNNs [37], [38]. Additionally, VAEs offer a probabilistic model for generating new data samples by learning the latent representation of the given data. They are particularly valuable in scenarios with missing labeled information. But, in this survey, we cover state-of-the-art diffusion-based data augmentation techniques, which is one type of generative technique. Diffusion-based methods offer a different approach to generating synthetic data and have shown promising results in enhancing model generalization. Readers interested in exploring the use of GAN-based data augmentation further may refer to Su et al. [37] and Yue et al. [38] for up-to-date reviews on this research area.

Our contributions and their associated benefits are outlined as follows:

- We present a comprehensive taxonomy of image data augmentation.
- An extensive survey of state-of-the-art data augmentation techniques, including visual examples, is provided, streamlining researchers' exploration within the field.
- We collect and compare the performance of state-of-the-art data augmentation techniques across major computer vision tasks, aiding in informed decision-making for researchers.
- Additionally, we highlight challenges and identify future research directions, fostering interest and advancements in the research community.

The rest of the work is organized as follows: first, we provide an in-depth discussion of the taxonomy and background of each method in our proposed taxonomy in Section II. Next, we present the results of our experiments on the impact of data augmentation on three popular computer vision tasks in Section III. In Section IV, we discuss the current trends in data augmentation and highlight potential gaps in the existing research. Finally, we summarize our findings and draw conclusions in Section V.

II. TAXONOMY AND BACKGROUND

Before delving into the taxonomy of image data augmentation techniques, it is essential to first define what constitutes ''image data'' within the context of this paper. Image data typically consists of three channels: Red (R), Green (G), and

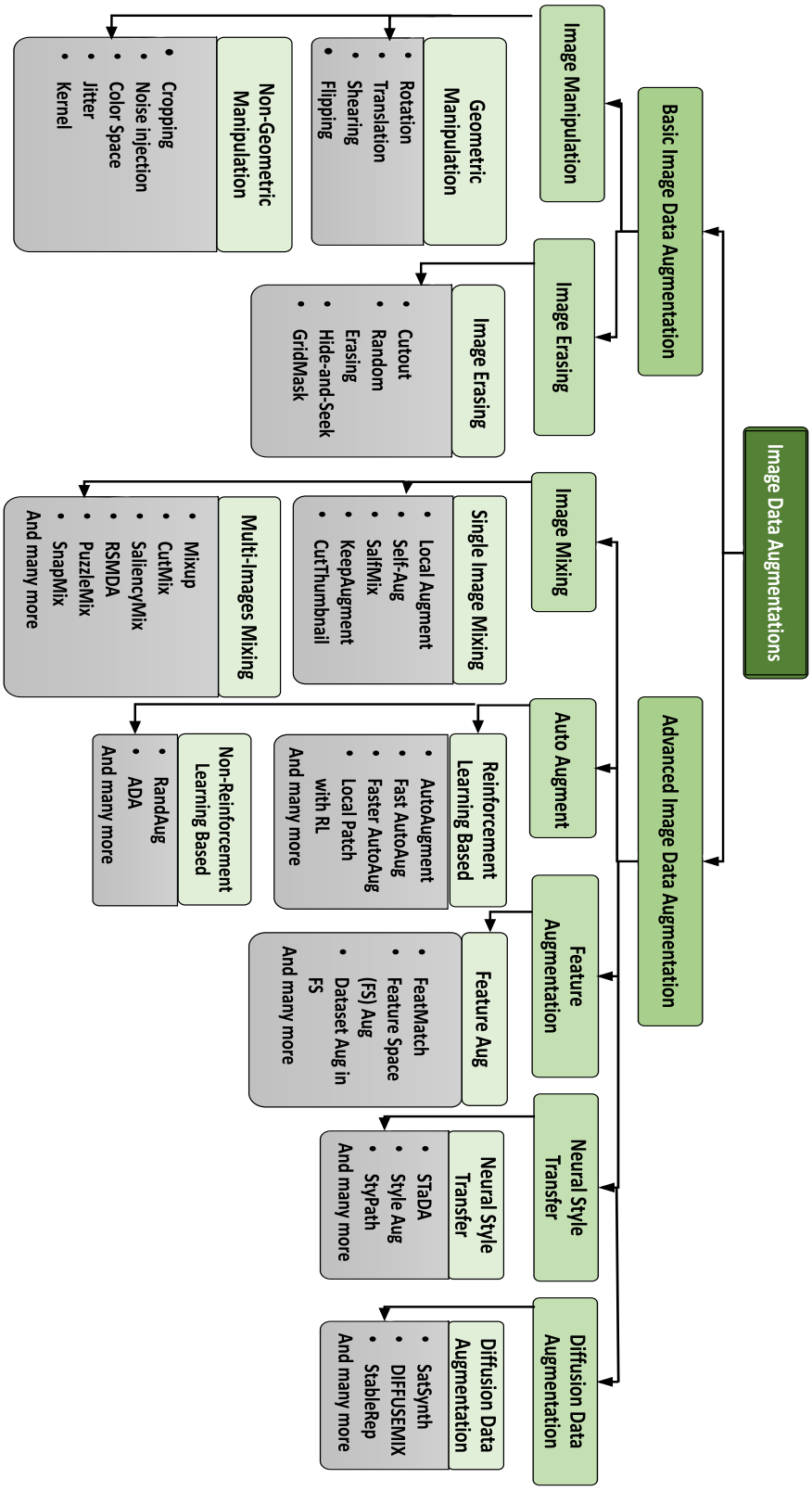

FIGURE 1. Image data augmentation taxonomy. Note: Due to space constraints, not all image data augmentation techniques are included in this taxonomy. However, all relevant and remaining image data augmentation approaches are discussed in the text, including additional sub-types of categories.

Blue (B), and is represented with dimensions of Height (H) x Width (W) x Channels (C).

In alignment with the importance of data augmentation in enhancing model performance, this section presents a novel taxonomy of data augmentation techniques aimed at providing a structured framework for understanding and categorizing the diverse methodologies employed in the field. Some of the data augmentation techniques are not included in taxonomy Fig. 1, due to space constraint, like Multi-images mixing: FMix, MixMo, StyleMix, RandomMix etc. But each of data augmentation is explained in the next subsections. Inspired by previous article classifications and motivated by the need for clarity and organization, our taxonomy delineates data augmentation into two primary branches: basic image data augmentation and advanced image data augmentation. This classification scheme draws upon insights from seminal works such as Shorten and Khoshgoftaar's survey on Image Data Augmentation [11], Zhu et al.'s [29] examination of advancements in Point Cloud Data Augmentation, and Garcea et al.'s [39] systematic literature review on Data Augmentation for Medical Imaging. By leveraging these insights, our taxonomy not only establishes a foundation for discussion but also fosters coherence and understanding across various augmentation techniques.

Basic image data augmentation encompasses fundamental techniques for image data augmentation, while the advanced image data augmentation encompasses more complex techniques. The specifics of each image data augmentation method are thoroughly discussed in subsequent sections.

A. BASIC IMAGE DATA AUGMENTATION

This section describes basic image data augmentation and their classification as shown in Fig. 1. They are classified as below:

- **Image Manipulation**
  - *Geometric Manipulation*
  - *Non-Geometric Manipulation*
- **Image Erasing**
  - *Erasing*

1) IMAGE MANIPULATION

Image manipulation refers to the changes made in an image with respect to its position or color. Positional manipulation is made by adjusting the position of the pixels while color manipulations are made by altering the pixel values of the image. Image manipulation is further divided into two main categories. Each of them is discussed below:

Geometric Data Augmentation: It encompasses modifications to the geometric attributes of an image, including its position, orientation, and aspect ratio. This technique involves transforming the arrangement of pixels within an image through a variety of techniques such as rotation, translation, shearing, and flipping. These methods are widely used in the domain of computer vision to diversify the training data and improve the resilience of models to diverse transformations. The utilization of geometric data augmentation has become a critical component in the development of robust computer vision algorithms. Each of the geometric data augmentation approaches is discussed below:

(a) **Rotation:** Rotation data augmentation involves rotating an image by a specified angle within the range of 0 to 360 degrees, as shown in Fig. 2. The precise degree of rotation is a hyperparameter that requires careful consideration based on the nature and characteristics of the dataset. For instance, in the MNIST [40] dataset, rotating all digits by 180 degrees, right-rotation of 6 results into a 9, would not be a meaningful transformation. Therefore, a thorough understanding of the dataset is necessary to determine the optimal degree of rotation and achieve the best results.

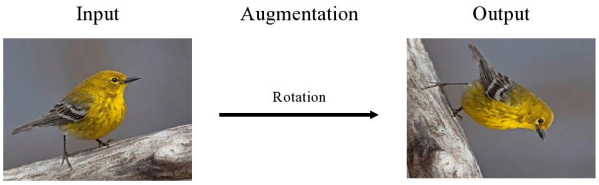

FIGURE 2. Rotation.

(b) **Translation:** It involves shifting an image in any of the upward, downward, right, or left directions, as illustrated in Fig. 3, in order to provide a more diverse representation of the data. The magnitude of this type of augmentation must be selected with caution, as an excessive shift can result in a substantial change in the appearance of the image. For example, translating a digit 8 to the left by half the width of the image could result in an augmented image that resembles the digit 3. Hence, it is imperative to consider the nature of the dataset when determining the magnitude of the translation augmentation to ensure its efficacy. T

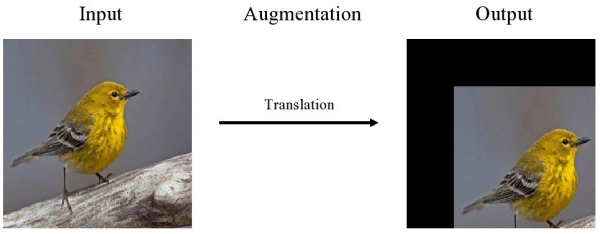

FIGURE 3. Translation.

(c) **Shearing:** It involves shifting one part of an image in one direction, while the other part in opposite direction , as shown in Fig. 4. This technique can provide a new and diverse perspective on the data, thereby improving the robustness of a model. However, excessive shearing

can cause significant deformation of the image, making it difficult for the model to accurately recognize the objects within it. It is therefore important to consider the amount of shearing applied to the data carefully in order to avoid over-augmenting the images and introducing unwanted noise. In this way, shearing can be a powerful tool for enhancing the generalization ability of computer vision models, while avoiding the potential drawbacks of over-augmentation. For example, applying excessive shearing on cat image during data augmentation may result in a distorted, stretched appearance, hindering the ability of a model to correctly classify the image as a cat. It is crucial to find a balance between the amount of shearing applied and the desired level of diversity, as excessive shearing can introduce significant noise.

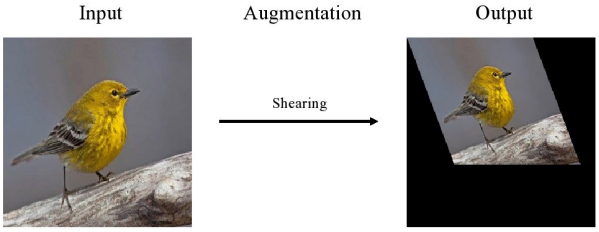

FIGURE 4. Shearing.

(d) **Flipping:** It is a type of image data augmentation technique that involves flipping an image either horizontally or vertically, as shown in Fig. 5. The efficacy of this method has been demonstrated on various widely-used datasets, including CIFAR10 and CIFAR100 [91]. However, care must be taken when applying this technique, as the outcome may depend on the nature of the dataset. For instance, the horizontal flipping of the digit 2 in the Urdu digits dataset [162] may result in the appearance of the digit "6". As such, the choice of flipping must be made carefully to ensure that the desired level of augmentation is achieved without introducing significant noise into the data.

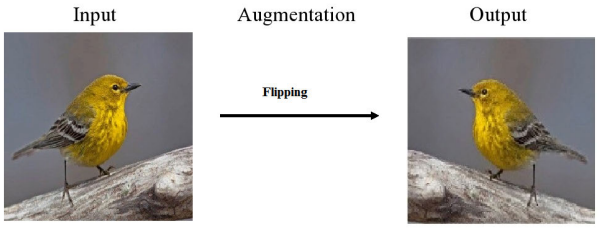

FIGURE 5. Flipping.

Non-Geometric Data Augmentation: This category focuses on modifications to the visual characteristics of an image, as opposed to its geometric shape. This includes techniques such as noise injection, flipping, cropping, resizing, and color space manipulation. These techniques can help improve the generalization performance of a model by exposing it to a wider variety of image variations during training. However, it is important to consider the trade-off between augmenting the data and preserving the integrity of the underlying information in the image. The following section outlines several classical non-geometric data augmentation approaches.

(a) **Cropping and resizing:** It is a common pre-processing data augmentation technique that can be applied randomly or to the center of the image, as demonstrated in Fig. 6. This technique involves trimming the image and then resizing it back to its original size, preserving the original label of the image. However, caution must be exercised when using cropping as a data augmentation method, as it may result in misleading information for the model, such as cropping the upper or lower part of the digit 8 and making it appear as the digit 0.

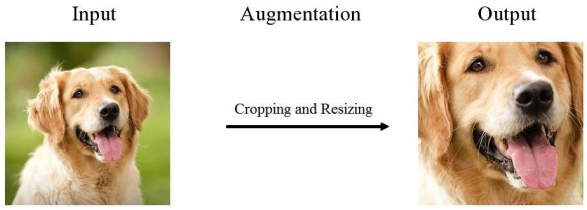

FIGURE 6. Cropping and resizing.

(b) **Noise Injection:** It is a data augmentation technique that has been demonstrated to enhance the robustness of neural networks in learning features and defending against adversarial attacks. As shown in Fig. 7 and demonstrated in the survey of nine datasets from the UCI repository [11], the use of noise injection has resulted in impressive performance improvements.

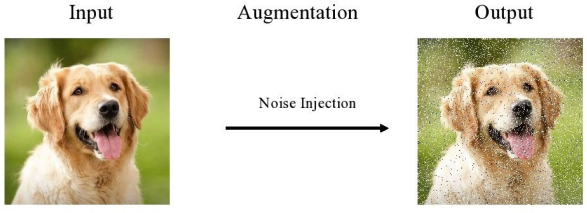

FIGURE 7. Noise injection.

(c) **Color Space:** By altering the values of each channel separately, this technique can prevent a model from becoming biased towards specific lighting conditions. The most straightforward approach to perform color space augmentation involves replacing a single channel within the image with a randomly generated channel of the same size, or with a channel filled with either 0 or 255, as shown in Fig. 8. The utilization of color space manipulation is commonly observed

in photo editing applications, where it is used to adjust the brightness or darkness of the image [11]. However, one potential downside is that it can introduce unrealistic color variations, which may confuse the model and reduce its ability to generalize to real-world images, such as incorrectly altering the colors of specific objects (e.g., flags) that have distinct color patterns [11].

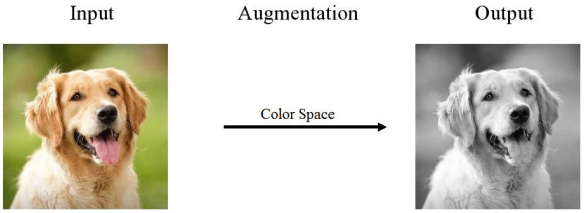

FIGURE 8. Color space.

(d) **Jitter:** This technique involves randomly altering the brightness, contrast, saturation, and hue of an image. The four hyperparameters, i.e., brightness, contrast, saturation, and hue, can be adjusted by specifying their minimum and maximum range, as demonstrated in Fig. 9. However, it is important to carefully select these ranges as improper adjustments can negatively impact the image's content. For example, increasing the brightness of X-Ray images used for lung disease detection can result in the whitening and blending of the lungs in the X-Ray, hindering the diagnosis of the disease [11].

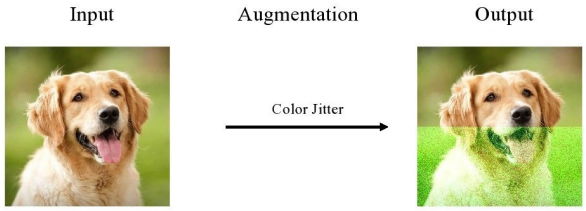

FIGURE 9. Jitter.

(e) **Kernel Filter:** It is a form of data augmentation that enhances or softens the image. This is achieved by applying a window, with a specified size of n x n, containing a Gaussian-blur or an edge filter to the image. The Gaussian-blur filter serves to soften the image, while the edge filter sharpens its edges either horizontally or vertically, as demonstrated in Fig. 10. While this technique can be beneficial, it also has potential downsides. For example, excessive blur can remove distinguishing details from the image, making it difficult for the model to accurately identify important features. Additionally, over-sharpening can introduce noise and artifacts, which might lead to incorrect interpretations by the model.

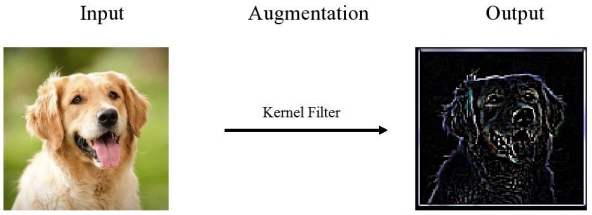

FIGURE 10. Kernel filter.

2) IMAGE ERASING

Image erasing sub-category involves the process of removing specific parts of an image and replacing them with 0, 255, or the mean of the entire dataset. It can affect the overall appearance and content of the image without necessarily altering its geometric or visual characteristics uniformly (non-geometric). That is why, Image erasing is classified into separate category. This type of data augmentation includes various methods such as cutout, random erasing, hide-and-seek, and grid mask, each with their unique implementation and purpose.

(a) **Cutout:** The Cutout data augmentation method involves the random removal of a sub-region within an image, which is then filled with a constant value such as 0 or 255, during the training phase. This approach has been shown to result in improved performance on widely used datasets [41]. An illustration of the Cutout data augmentation process is provided in Fig. 23.

(b) **Random erasing (RE):** RE [42] randomly erases the sub-region in the image similar to cutout. But the main difference is, it randomly determines whether to mask out region or not and also determines the aspect ratio and size of the masked region. RE demonstration for different tasks is shown in Fig. 11.

FIGURE 11. Random erasing examples [42].

(c) **Hide-and-Seek:** The process of hide-and-seek data augmentation [43] involves dividing an image into uniform squares of random size and then randomly

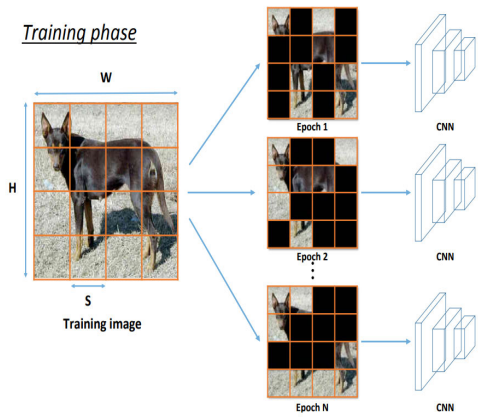

FIGURE 12. Hide-and-Seek augmentation [43].

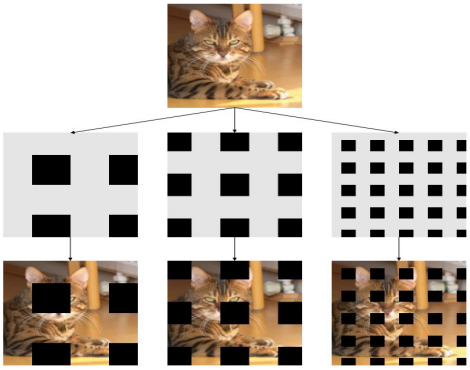

FIGURE 13. GridMask augmentation [44].

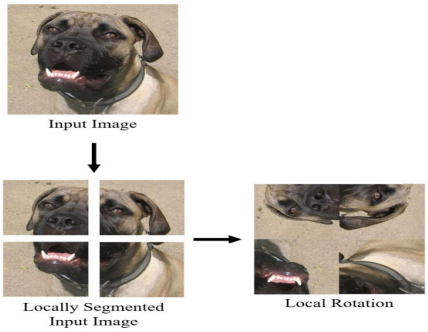

FIGURE 14. Local rotation image [45].

removing a specified number of these squares. This technique aims to force neural networks to learn relevant features by hiding important information. On every epoch, different augmented image is passed to the neural network during the training process as depicted in Fig. 12. It is important to note that while this technique has been found to be particularly effective in scenarios such as object detection and image classification, where it enhances model robustness to occlusions by forcing the model to recognize objects from incomplete information. For example, in autonomous driving, it helps the model identify partially obscured pedestrians or vehicles. However, this technique may negatively impact performance in fine-grained classification tasks or sensitive applications like medical diagnosis, where masking parts of the image could lead to the loss of crucial details necessary for accurate analysis.

(d) **GridMask Data (GM) Augmentation** GM data augmentation technique [44] aims to tackle the challenges associated with randomly removing regions from images. This process, which can completely erase objects or strip away context information, requires a trade-off between the two. To resolve this, GridMask creates a uniform masking pattern and applies it to images as demonstrated in Fig. 13.

B. ADVANCED IMAGE DATA AUGMENTATION

The wide range of innovative methods for augmenting image data, such as mixing images in novel ways, have been developed using reinforcement learning, feature-based augmentation, and style-based augmentation. To better understand these advancements, advanced data augmentation techniques have been classified into different major categories. These categories provide a useful framework for surveying the current state of the field and identifying areas for further research and development.

1) IMAGE MIXING

This technique involves blending one or more images, including the same image, resulting in improved deep neural network model accuracy. We categorize image mixing data augmentation into two sub-categories: single image mixing and multi-images mixing. We compare the effectiveness of these sub-categories on benchmark datasets (such as CIFAR10, CIFAR100, ImageNet etc), as shown in Table 3, 4, 9, 10 and 11.

Single Image Mixing Data Augmentation: A single-image mixing technique uses only one image and mixes a single image from different mixing points of view. Recently, there has been a lot of work done on single-image augmentation, such as LocalAugment, SelfAugmentation, SalfMix, and many more. The description of each SOTA single image mixing data augmentation has been discussed below.

(a) **Local Augment:** Kim et al. [45] proposed a technique called LocalAugment, which involves dividing an image into smaller patches and applying different types of data augmentation to each patch. Local Augment involves randomly modifying individual patches of an image, whereas global augment pertains to altering the entire image. Both types of augmentations are visually demonstrated in Fig. 14. The purpose of this technique is to increase diversity in local features, which could help reduce bias and improve generalization performance of

neural networks. While this approach does not preserve the global structure of an image, it provides a rich set of local features that can benefit neural network training. Fig.14 and Fig.15 provide visual representations of the LocalAugment technique.

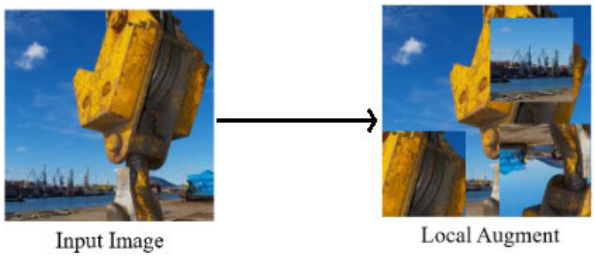

FIGURE 15. LocalAugment, example is taken from [45].

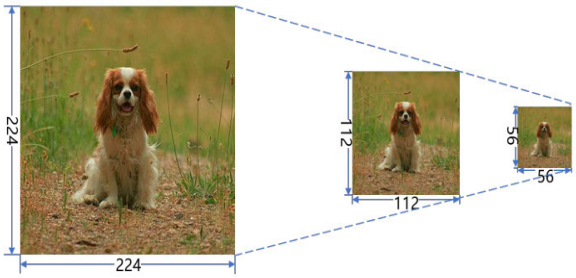

FIGURE 16. This image shows an example of reduced images that are called thumbnails. After reducing the image to a certain size of 112×112 or 56×56 [50].

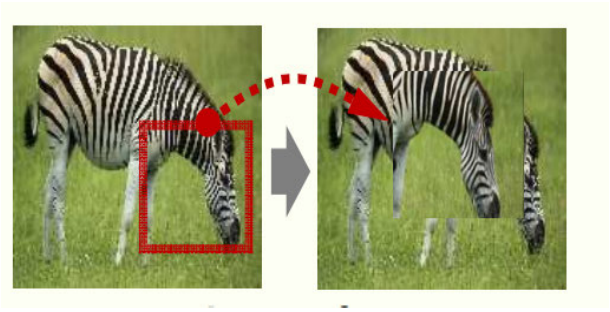

FIGURE 17. Self augmentation [46].

(b) **Self-Augmentation:** Self-Augmentation [46] crops random region of an image and pastes randomly in the image, improving the generalization capability in few-shot learning. Moreover, the self-augmentation combines regional dropout and knowledge distillation - knowledge from the trained large network is transferred to a small network. This augmentation process is demonstrated in the Fig. 17.

(c) **SalfMix:** A work by Choi et al. [47] focuses on whether it is possible to generalize neural networks based on single-image mixed augmentation. For that purpose, it proposes SalfMix, the first salient part of the image is found to decide which part should be removed and which portion should be duplicated. Most salient regions are cropped and placed into non-salient regions. This process is defined and compared with other techniques in Fig. 18.

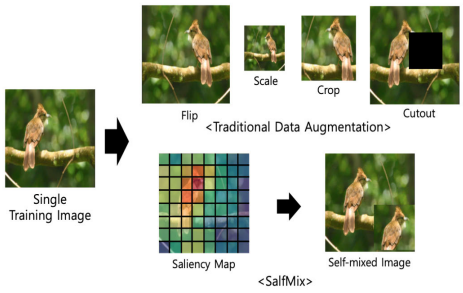

FIGURE 18. Conceptual comparison between SalfMix method and other single image-based data augmentation methods, the example is taken from [47].

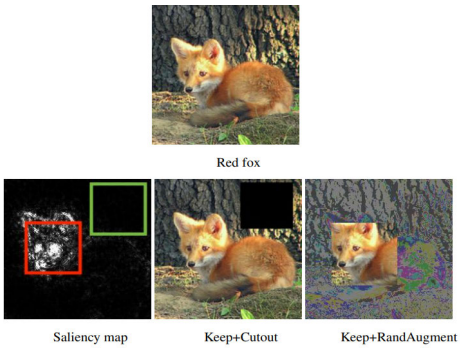

FIGURE 19. This image shows the example of KeepAugment with other augmentation methods, courtesy [48].

(d) **KeepAugment** KeepAugment [48] is introduced to prevent distribution shift which degrades the performance of neural networks. The idea of KeepAugment is to increase fidelity by preserving the salient features of the image and augmenting the non-salient region. Preserved features help to increase diversity without shifting the distribution. KeepAugment is demonstrated in Fig. 19.

(e) **You Only Cut Once:** You Only Cut Once (YOCO) [49] is introduced with the aim of recognizing objects from partial information and improving the diversity of augmentation that encourage neural networks to perform better. YOCO makes two pieces of image and augmentation is applied on each piece, then each piece is concatenated for an image and YOCO shows impressive performance and compared with SOTA augmentation methods, sometimes it outperforms them. It is easy to implement, has no parameters, and is easy to use. The YOCO augmentation process is shown in Fig. 20.

(f) **Cut-Thumbnail:** Cut-Thumbnail [50] is a novel data augmentation, that resizes the image to a certain small size and then randomly replaces the random region of

FIGURE 20. YOCO augmentation, image is taken from [49].

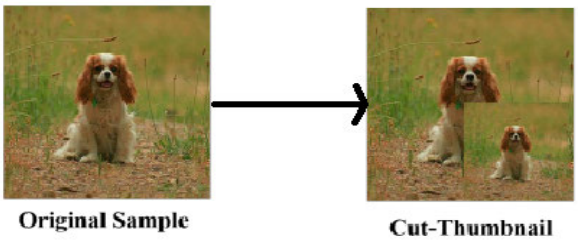

FIGURE 21. Cut-Thumbnail [50].

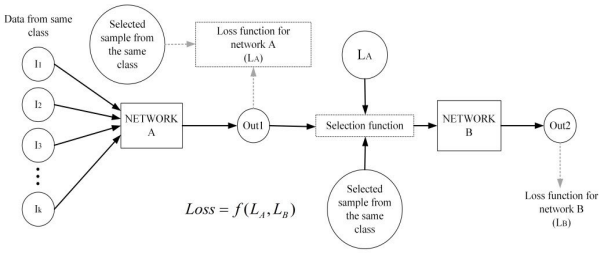

FIGURE 22. Illustration of smart data augmentation, courtesy [50].

FIGURE 23. Overview of the Mixup, Cutout, and CutMix [15].

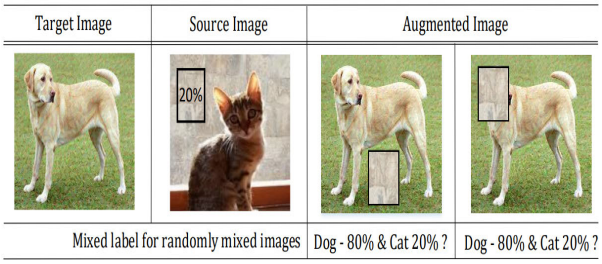

FIGURE 24. Problem in cutout solved by SaliencyMix, image is taken from [53].

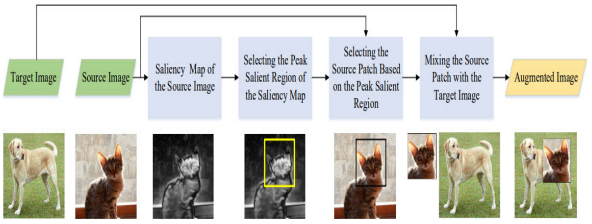

FIGURE 25. SaliencyMix data augmentation procedure, courtesy [53].

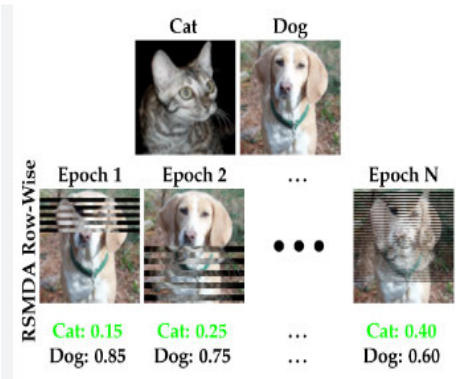

FIGURE 26. RSMDA demonstration. Image is taken from [54].

the image with the resized image, aiming to alleviate the shape bias of the network. The advantage of Cut-thumbnail is, that it not only preserves the original image but also keeps it global in the small resized image. On ImageNet, it shows impressive performance using resnet50 as a backbone. Overall, the cut-thumbnail process and its comparison are shown in Fig. 16 and Fig. 21, respectively.

Multi-Images Mixing: Multi-Images Mixing data augmentation uses more than one image and applies different mixing strategies. Recently, many researchers have explored a lot of Multi-Images Mixing strategies and still, it is a very attentive topic for many researchers. Recent work has included Mixup, CutMix, SaliencyMix, and many more. Each of the relevant multi-images image mixing data augmentation techniques is discussed below.

(a) **Smart Augmentation:** Smart data augmentation [51] starts by learning data augmentation using generative network for the target network in a way to minimize the loss. More detail is shown in Fig. 22.

(b) **Mixup:** Mixup blends two images based on the blending factor (alpha) and the corresponding labels of these images are also mixed in the same way. Mixup data augmentation [52] consistently improved the performance not only in terms of accuracy but also in terms of robustness. Experiments on ImageNet-2012 [12], CIFAR-10, CIFAR-100, Google commands [1] and UCI datasets [2] showed impressive results on SOTA methods. Further demonstration and comparison are shown in the Fig. 23.

[1]https://research.googleblog.com/2017/08/launching-speech-commands-dataset.html

[2]http://archive.ics.uci.edu/ml/index.php

(c) **CutMix:** CutMix tackles the issues of information loss and region dropout [15]. It is inspired by cutout [41], where any random region is filled with 0 or 255, while in cutmix instead of filling the random region with 0 or 255, the region is filled with a patch from another image. Correspondingly, their labels are also mixed proportionally to the number of pixels mixed. It is compared with other methods and shown in Fig. 23.

(d) **SaliencyMix:** This technique [53] tackles the problem of cutmix and argues that filling a random region of the image with a patch from another will not guarantee that patch has rich information and thereby mixing labels of unguaranteed patches leads the model to learn unnecessary information about the patch [53]. To deal with that issue, saliencyMix first selects the salient part of the image and pastes it to a random region or salient or non-salient of another image. It is shown in Fig. 24 and Fig. 25.

(e) **Random Slices Mixing Data Augmentation (RSMDA)** RSMDA [54] deals issues of feature losing in single image erasing data augmentation. RSMDA gets the slices of one image and mixes them with another image alternatively and the corresponding labels are also mixed accordingly. RSMDA further investigates three different strategies of RSMDA; row-wise slice mixing, column-wise slice mixing and randomness of both. Row-wise slice mixing has shown superior performance. A demonstration of the row slices mixing strategy is in Fig. 26.

(f) **Puzzle Mix:** Puzzle Mix data augmentation [55] focuses on using explicitly salient information and basic statistics of image wisely with the aim of breaking the misleading supervision of neural networks over existing data augmentation approaches. Furthermore, the demonstration is shown and compared with relevant methods in Fig. 27.

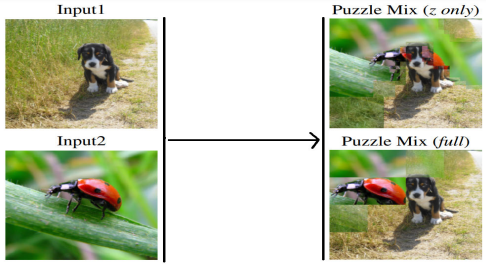

FIGURE 27. A visual representation of PuzzleMix. It ensures to contain sufficient target class information while preserving the local statistics of each [55].

(g) **Semantically Proportional Mixing (SnapMix):** SnapMix [56] utilises class activation map (CAM) to reduce the label noise level. SnapMix creates the target label considering the actual salient pixel taking part in the augmented image, which ensures semantic correspondence between the augmented image and mixed labels. The overall process is demonstrated and compared with closely matching augmentation approaches in Fig. 28.

(h) **FMix:** FMix [57], a kind of mixed sample data augmentation (MSDA), utilises the random binary masks.

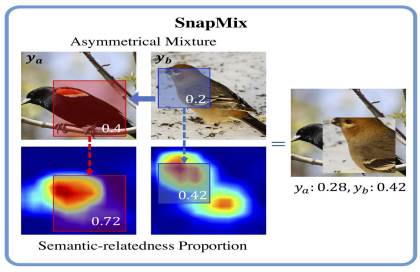

FIGURE 28. A visual representation of SnapMix. Label generated by SnapMix is visually more consistent with the mixed image semantic structure compared to CutMix and Mixup, courtesy [56].

FIGURE 29. Example masks and mixed images from CIFAR-10 for FMix, example is from [57].

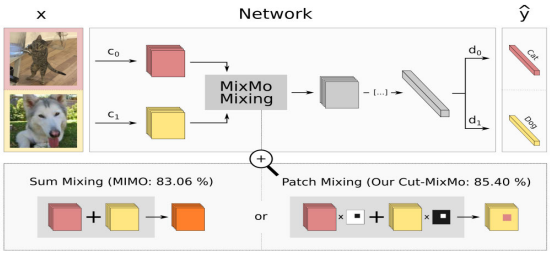

FIGURE 30. This image shows the overview of MixMo augmentation, the image is taken from [58].

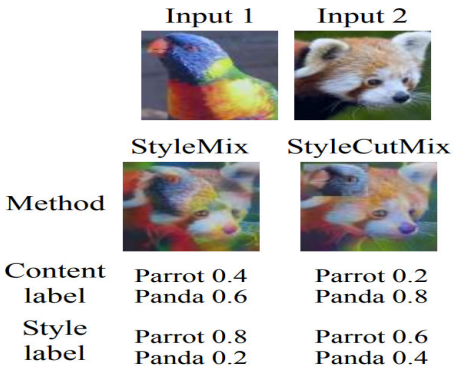

FIGURE 31. A visual representation of StyleMix [59], example is from [59].

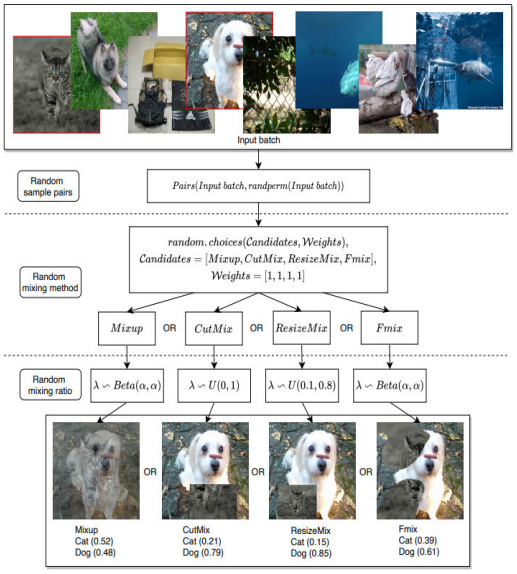

FIGURE 32. An illustrative example of RandomMix, image is taken from [60].

These random binary masks are acquired by applying a threshold to low-frequency images that are obtained from the Fourier space. Once the mask is obtained, one color region is applied to input one and another color region is applied to another input. The overall process is shown in Fig. 29.

(i) **MixMo:** A work by Rame [58] focuses on the learning of multi-input multi-output via sub-network, which refers to a technique where multiple inputs are fed into the network, and each input is processed by a separate sub-network or branch within the overall architecture. Each sub-network generates its own output, which could represent different aspects or features of the input data. The main motivation of the paper is to replace direct hidden summing operations with more solid mechanisms. For that purpose, it proposes MixMo, which embeds M inputs into the shared space, mixes and passes these to a further layer for classification. Moreover, the overall process is demonstrated in Fig. 30:

(j) **StyleMix:** Recent article [59] targets previous approaches problems that are unable to differentiate between content and style features, approaches such as mixup based data augmentation. To remedy this problem, it proposes two approaches styleMix and StyleCutMix, this is the first work that separately deals with content and style features of images very carefully and it showed impressive performance on popular benchmark datasets. The overall process is defined and compared with SOTA approaches in Fig. 31.

(k) **RandomMix:** RandomMix [60] randomly selects augmentation from a set of image mixing augmentation approaches and applies it to images, enabling the model to look at diverse samples. This method showed impressive results over SOTA image mixing methods. The overall demonstration is shown in Fig. 32.

(l) **MixMatch:** MixMatch [61] data augmentation technique is very useful in semi-supervised learning. It augments single image K times and passes all K number of images to a classifier, averages their prediction and finally, their predictions are sharpened by adjusting their distribution temperature term. It is demonstrated in Fig. 33.

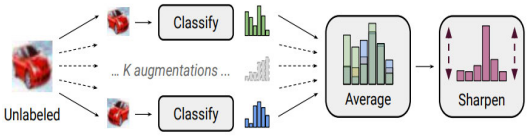

FIGURE 33. Diagram of the label guessing process used in MixMatch, courtesy [61].

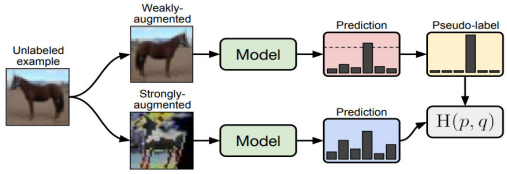

FIGURE 34. The procedure of FixMatch [63].

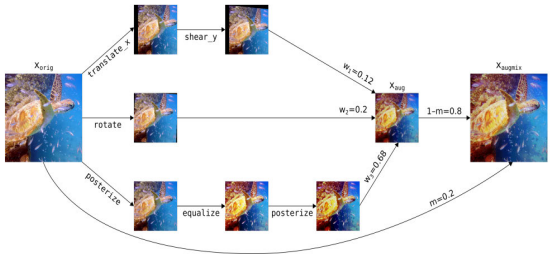

FIGURE 35. AugMix augmentation [64].

(m) **ReMixMatch:** ReMixMatch [62], an extension of MixMatch [61] is proposed to make prior work efficient by introducing distribution alignment and augmentation anchoring. The distribution alignment task aims to minimize the gap between the marginal distribution of predictions on unlabeled data and the marginal distribution of ground truth labels. On the other hand, augmentation anchoring feeds multiple strongly augmented versions of the input into the model and encourages each output to be close to the prediction for a weakly-augmented version of the same input. The process is illustrated in Fig. 36.

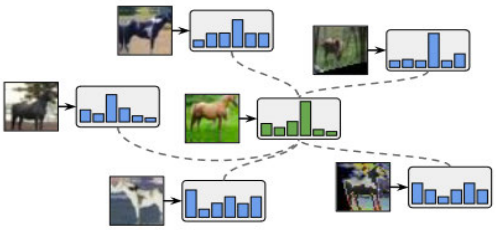

FIGURE 36. Anchoring augmentation. It makes predictions on strong augmentation of the same image (blue) using the forecast for a weakly enhanced image (green, centre), courtesy [62].

(n) **FixMatch:** Fixmatch [63] is a method for improving the performance of semi-supervised learning (SSL). It first assigns pseudo-labels to unlabeled images that have a predicted probability above a certain threshold, and then trains the model to match these labels using cross-entropy loss on a strongly augmented version of the image. The process is illustrated in Fig. 34.

(o) **AugMix:** The approach proposed in [64] presents Augmix, a data augmentation technique that aims to reduce the distribution gap between training and test data. Augmix applies M random augmentation to an input image, each with a random strength, and merges the resulting images to produce a new image that spans a wider area of the input space. The process is illustrated in Fig. 35, where three branches perform separate augmentation and additional operations are added to increase diversity. The resulting images are then mixed to produce a final augmented image, which is effective in improving model robustness.

(p) **Simple Copy-Paste:** Copy-paste [65] involves copying and pasting instances from one image to another to create an augmented image. This simple technique has shown promising results and is easy to implement. Fig. 37 illustrates the process, where instances from two images are pasted onto each other at different scales.

(q) **Improved Mixed-Example Data Augmentation:** Recently, label non-preserving data augmentation techniques based on linear combinations of two examples have demonstrated promising results. Summers et al. [66], the authors investigate two research questions: (i) the reasons behind the success of these methods and (ii) the significance of linearity in data augmentation. Fig. 38 illustrates the overall process.

(r) **Random image cropping and patching (RICAP):** RICAP [67] is a new data augmentation technique that cuts and mixes four images rather than two images. The key idea behind RICAP is to crop patch from each of the four images and then mixes these patch to create augmented image. The labels of the images are also mixed in proportion to the area of the patcehs. This technique showed impressive performance on popular datasets i.e., CIFAR10, CIFAR100, and imageNet. RICAP demonstration is shown in Fig. 39.

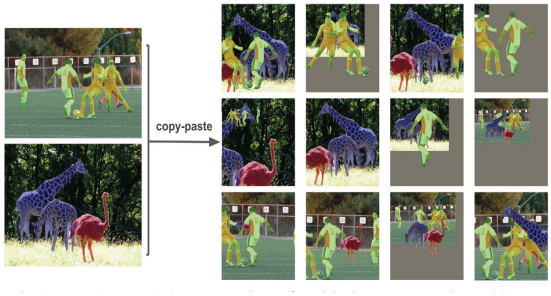

FIGURE 37. Simple Copy-Paste method, image courtesy [65].

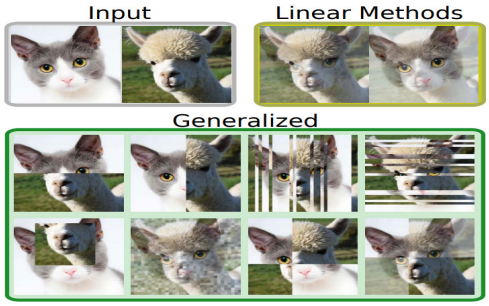

FIGURE 38. A visual comparison of linear methods and generalized augmentation performed by Improved Mixed-Example [66].

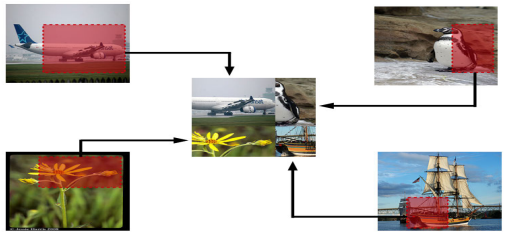

FIGURE 39. A conceptual explanation of the RICAP data augmentation [67].

(s) **Cutblur:** Cutblur [68] explores and analyses existing data augmentation techniques for super-resolution and proposes another data augmentation technique for super-resolution, named cutblur that cuts high-resolution image patches and pastes to corresponding low-resolution images and vice-versa. Cutblur shows impressive performance on several super-resolution benchmark datasets. Furthermore, the process is illustrated in Fig. 41 and Fig. 42.

(t) **ResizeMix:** ResizeMix [69] method directly cuts and pastes the source data in four different ways to target the image. These four different ways include salient part, non-part, random part or resized source image to patch, as shown in the Fig. 40. It answers two questions:

- How to obtain a patch from the source image?
- Where to paste the patch from the source image in the target image?

Furthermore, it was found that saliency information is not important to promote mixing data augmentation. ResizeMix is shown in the Fig. 40.

(u) **ClassMix:** ClassMix [70] is novel data augmentation for semi-supervised learning for semantic segmentation task. It showed that traditional data augmentation approaches are not effective for image semantic segmentation as they are for image classification. The proposed data augmentation named ClassMix, augments the training sample by mixing unlabeled samples, by exploiting network prediction while taking into account object boundaries. It showed a massive performance gain on two common semantic segmentation datasets for semi-supervised learning. The overall process is shown in the Fig. 43.

(v) **Context Decoupling Augmentation(CDA):** CDA [71] deals the problem of traditional data augmentation techniques for Weakly Supervised Semantic Segmentation (WSSS), increasing the same contextual data semantic samples does not add much value in object differentiation, i.e., in image classification, ''cat'' recognition is due to the cat itself and its surrounding context, these both contexts discourages model to focus only on the cat. CDA increases diversity of the specific object and it guides the network to break the dependencies between object and contextual information. In this way, it also provides augmentation and the network focuses on object(s) only rather than object(s) and its contextual information. A comparison of traditional data augmentation and CDA is shown below in the Fig. 44.

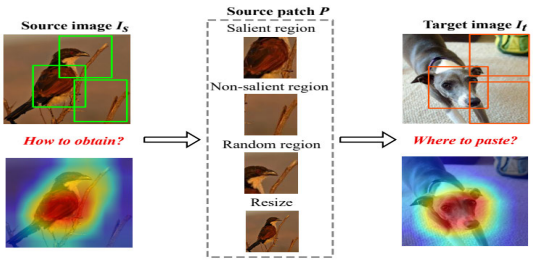

FIGURE 40. A visual representation of different cropping manners from the source image and different pasting manners to the target image, image is taken from [69].

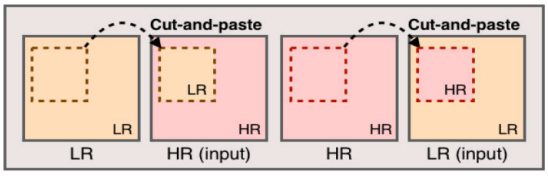

FIGURE 41. Illustration of CutBlur operation [68].

FIGURE 42. A visual comparison between High resolution, low resolution and CutBlur, courtesy [68].

(w) **ObjectAug:** ObjectAug [72] tackles the problem of mixing image-level data augmentation strategies, which failed to work for segmentation as object and background are coupled and boundaries of objects are not augmented due to their fixed semantic bond with the background. To mitigate this problem, Zhang et al. [72] proposes a novel approach named ObjectAug, object-level augmentation for semantic segmentation. First, it separates object(s) and backgrounds from an image with the help of semantic labels then each object is augmented using popular data augmentation techniques such as flipping and rotating. Pixels change due to these data augmentation are restored using image inpainting. Finally, the object(s) and background are coupled to create an augmented image. Experimental results suggest that ObjectAug has shown performance improvement for segmentation tasks. Furthermore, ObjectAug is shown in Fig. 45.

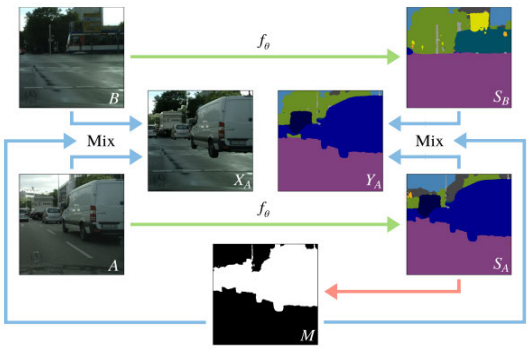

FIGURE 43. ClassMix augmentation, two images are sampled then based on the predictions of each image a binary mask is created. The mask is then used to mix the images and their predictions [70].

2) AUTOAUGMENT

The goal of this technique is to find the data augmentation policies from training data. It solves the problem of finding the best augmentation policy as a discrete search problem. It consists of a search algorithm and a search space. Furthermore, these techniques are classified into two sub-categories based on reinforcement learning and non-reinforcement learning.

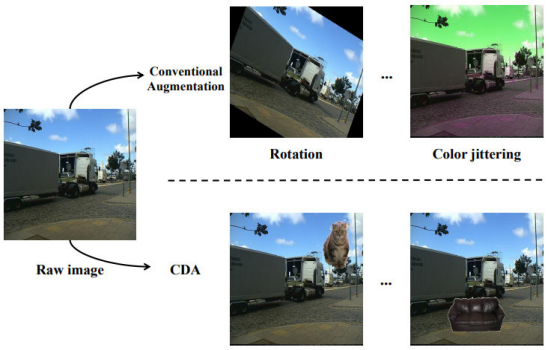

FIGURE 44. Difference between the conventional augmentation approach and context decoupling augmentation (CDA) [71].

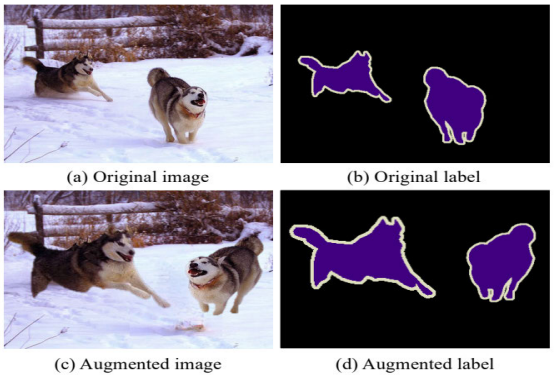

FIGURE 45. ObjectAug can perform various augmentation methods for each object to boost the performance of semantic segmentation. The left husky is scaled and shifted, while the right one is flipped and shifted. Thus, the boundaries between objects are extensively augmented to boost their performance [72].

Reinforcement Learning data augmentation: Reinforcement learning data augmentation techniques learn data augmentation policies from given data augmentation policies and magnitude.

(a) **AutoAugment:** AutoAugment [73] automatically finds the best data augmentation rather than manual data augmentation. To tackle the limitations of manual search-based data augmentation, Cubuk et al. proposes AutoAugment, where search space is designed and has policies consisting of many sub-policies. Each sub-policy has two parameters one is the image processing function and the second one is the probability with magnitude. These sub-policies are found using reinforcement learning as a searching algorithm. The overall process is demonstrated in Fig. 46.

(b) **Fast AutoAugment:** Fast AutoAugment [92] deals the problem of AutoAugment technique, which takes a lot of time to find the optimal data augmentation strategy. To reduce the searching time, fast auto augment finds more optimal data augmentation using an efficient search strategy based on density matching. It reduces the higher order of training time compared to AutoAugment. The overall procedure is shown in Fig. 47.

(c) **Faster AutoAugment:** Taster AutoAugment [74] intends to find effective data augmentation policies very efficiently. Faster AutoAugment is based on a differentiable augmentation searching policy and additionally, it not only estimates gradients for many transformation operations having discrete parameters but also provides a mechanism for choosing operations efficiently. Moreover, it introduces a training objective function with aim of minimising the distance between original and augmented distribution, which is also differentiable. Parameters of augmentation are updated during backpropagation. The Overall process is defined in Fig. 49:

(d) **Reinforcement Learning with Augmented Data (RAD):** RAD [75] enhances the performance of

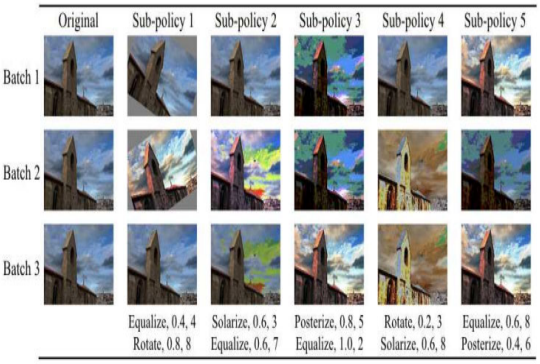

FIGURE 46. Overview of the sub-policies from ImageNet using AutoAugment [73].

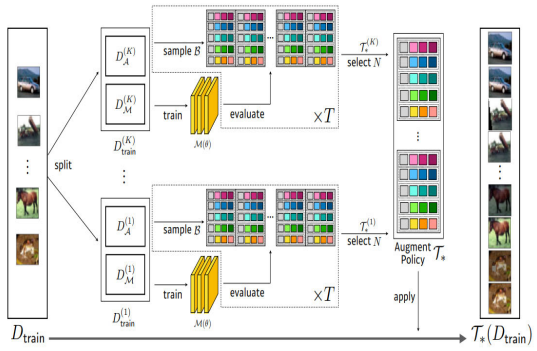

FIGURE 47. Overview of augmentation search by Fast AutoAugment algorithm, courtesy [92].

RL algorithms by targeting two issues i) learning data efficiency and ii) generalisation capability for new environments. Furthermore, it shows traditional data augmentation techniques enable RL algorithms to outperform complex SOTA tasks for pixel-based control

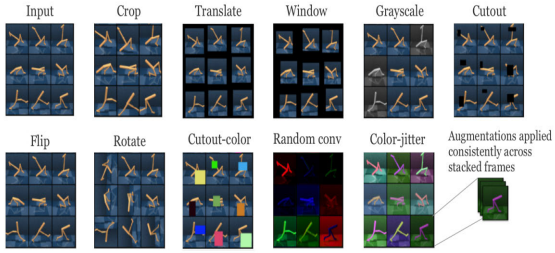

FIGURE 48. Overview of different augmentation investigated in RAD, the example [75].

and state-based control. Overall process is demonstrated in Fig. 48:

(e) **Local Patch AutoAugment(LPA):** LPA is the first work [76] to finds data augmentation policy for patch level using reinforcement learning, named multi-agent reinforcement learning (MARL). MARL starts by dividing images into patches and jointly finds the optimal data augmentation policy for each patch. It shows competitive results on SOTA benchmarks. Overall process is defined in Fig. 50:

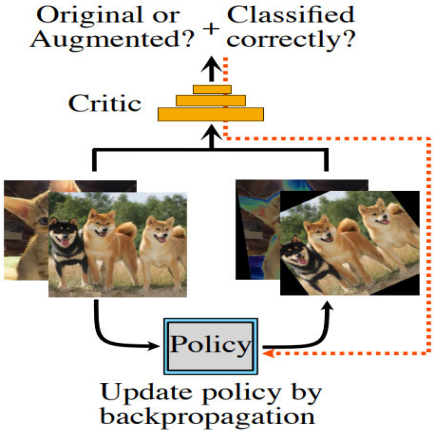

FIGURE 49. Overview of the Faster AutoAugment augmentation [74].

(f) **Learning Data Augmentation Strategies for Object Detection:** Zoph [78] proposes to use AutoAugment that learns the best policies for object detection. It finds the best operation and optimal value. Moreover, it deals two key issues of augmentation for object detection,

a) Classification learned policies can not directly be applied for detection tasks, and it adds more complexity to deal with bounding boxes if geometric augmentation methods are applied.

b) Most researchers think it adds much less value compared to designing new network architecture so gets less attention but augmentation for object detection should be selected carefully.

Some sub-policies for this data augmentation are shown below:

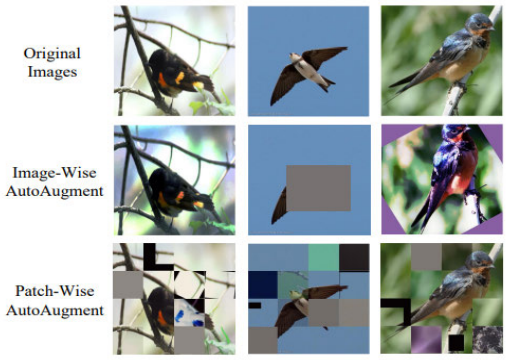

FIGURE 50. An illustration of different automated augmentation policies, courtesy [76].

Sub-policy 1. (Color, 0.2, 8), (Rotate, 0.8, 10)
Sub-policy 2. (BBox_Only_ShearY, 0.8, 5)
Sub-policy 3. (SolarizeAdd, 0.6, 8), (Brightness, 0.8, 10)
Sub-policy 4. (ShearY, 0.6, 10), (BBox_Only_Equalize, 0.6,8)
Sub-policy 5. (Equalize, 0.6, 10), (TranslateX, 0.2, 2)

(g) **Scale-aware Automatic Augmentation for Object Detection:** The article [79] proposes a new data augmentation for object detection named scale aware autoAug, first, it defines a search space where image level and box level data augmentation are prepared for scale invariance, secondly, it also proposes a new search metric named Pareto scale balance for search augmentation effectively and efficiently. Some examples of data augmentation are shown in Fig. 53.

Non-Reinforcement Learning data augmentation: In AutoAugment category, there are some approaches that do not require any reinforcement learning algorithm to find the best data augmentation, we refer to them as non-reinforcement learning data augmentation. We categorise a few of them as discussed below.

(a) **RandAugment:** Previous optimal augmentation finding uses reinforcement or some complex learning strategy that takes a lot of time to find. RandAugment augmentation [77] removes obstacles of a separate searching phase, which makes training more complex and consequently adds computational cost overhead. To break this, randaugment applies randomly N number of data augmentation methods with M magnitude of all augmentation methods. Some visualisation is demonstrated in Fig. 52:

(b) **RangeAugment:** RangeAugment [80] is a data augmentation technique that aims to improve upon the shortcomings of existing approaches like AutoAugment and RandAugment. These methods use manually-defined ranges of magnitudes for each type of data augmentation, which can result in sub-optimal policies.

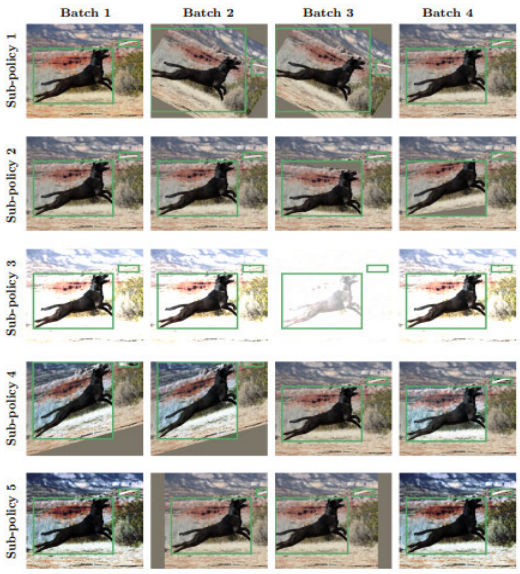

FIGURE 51. Different data augmentation sub-policies explored [78].

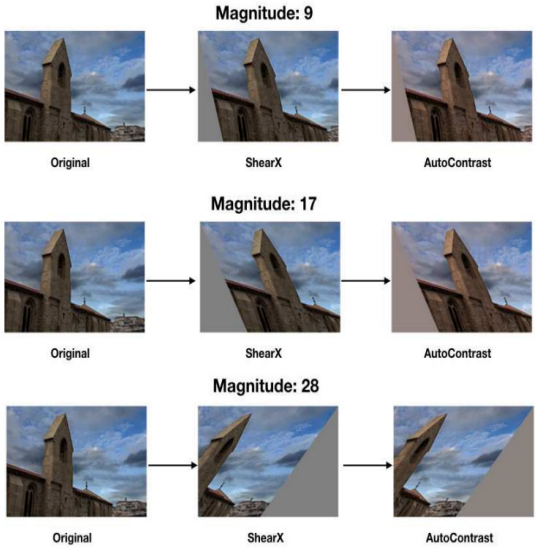

FIGURE 52. Example images augmented by RandAugment [77].

In contrast, RangeAugment learns efficient ranges of magnitudes for each augmentation and composite data augmentation by introducing an auxiliary loss based on image similarity. This loss is designed to control the magnitude ranges, resulting in more effective and optimal policies. The process of RangeAugment is illustrated in Fig. 54.

(c) **Adversarial Data Augmentation for Object Detection:** Data augmentation improves performance but it is difficult to understand whether these augmentation methods are optimal or not. Behpour et al. [81] provides a systematic way to find optimal adversarial perturbation

FIGURE 53. Example of scale-aware search space which includes image level and box-level augmentation [79].

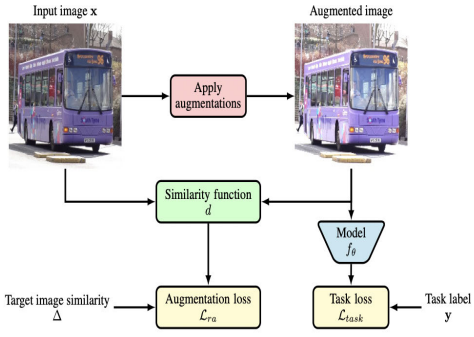

FIGURE 54. RangeAugment with neural network training [80].

of data augmentation for an object detection, that is based on game-theoretic interpretation aka Nash equilibrium of data. Nash equilibrium provides the optimal bounding box predictor and optimal design for data augmentation. Optimal adversarial perturbation refers to the worst perturbation of ground truth, that forces the box predictor to learn from the most difficult distribution of samples. An example is shown in Fig. 55.

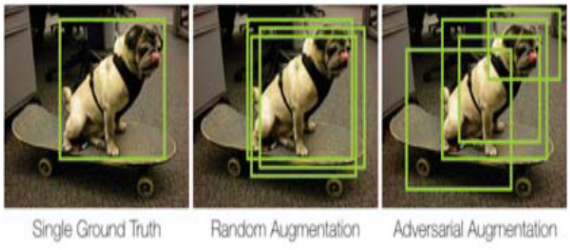

FIGURE 55. Annotation distribution types. Adversarial augmentation chooses bounding boxes that are as distinct from the truth as possible while yet containing crucial object characteristics. The example is taken from [81].

(d) **Deep CNN Ensemble with Data Augmentation for Object Detection:** It [82] is a new variant of the regions with convolutional neural network (R-CNN) model with

two core modifications in training and evaluation. First, it uses several different CNN models as ensembler in R-CNN, secondly, it smartly augments PASCAL VOC training examples with Microsoft COCO data by selecting a subset from Microsoft COCO datasets that are consistent with PASCAL VOC. Consequently, it increases the dataset size and improves the performance.

(e) **Robust and Accurate Object Detection via Adversarial Learning:** It [83] first shows classifier performance gain from different data augmentation approaches when it is fine-tuned to object detection tasks and suggests that the performance in terms of accuracy or robustness is not improving. The article provides a unique way of exploring adversarial samples that helps to improve performance. To do so, it augments the example during the fine-tuning stage for object detectors by exploring adversarial samples, which is considered as model-dependent data augmentation. First, it picks the stronger adversarial sample from detector classification and localization layers and ensures the augmentation policy remains consistent. It showed significant performance gain in terms of accuracy and robustness on different object detection tasks. Furthermore, the robustness and accuracy of the proposed method are shown in Fig. 56.

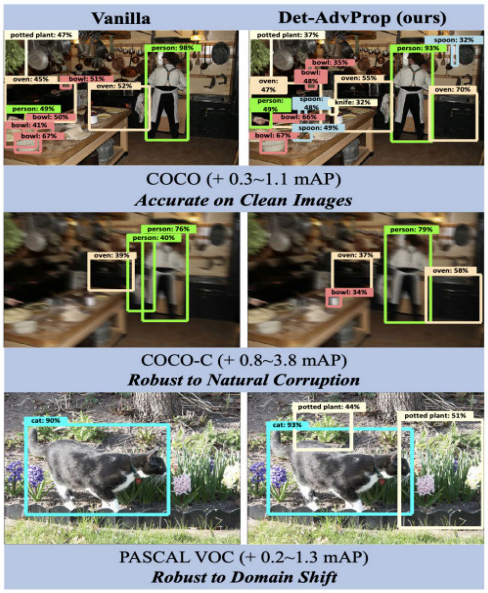

FIGURE 56. Overview of Robust and Accurate Object detection via adversarial learning. In the top image, it improves object detector accuracy on clean images. In the middle, it improves the detector's robustness against natural corruption, and at the bottom, it improves the robustness against cross-dataset domain shift. The image is taken from [79].

(f) **Perspective Transformation Data Augmentation for Object Detection:** It [84] is a new data augmentation for objection detection named perspective transformation that generates new images captured at different angles.

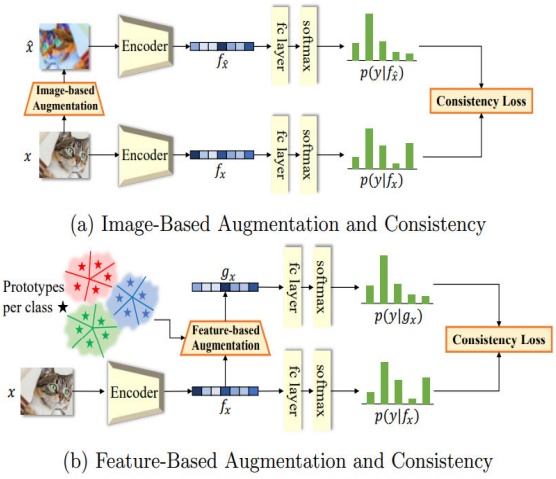

FIGURE 57. Overview of featMatch augmentation applied on images and features [95].

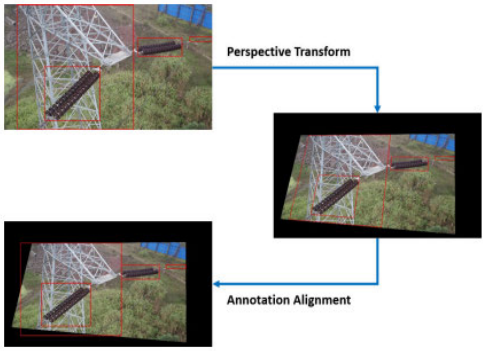

FIGURE 58. Perspective transformation data augmentation [84].

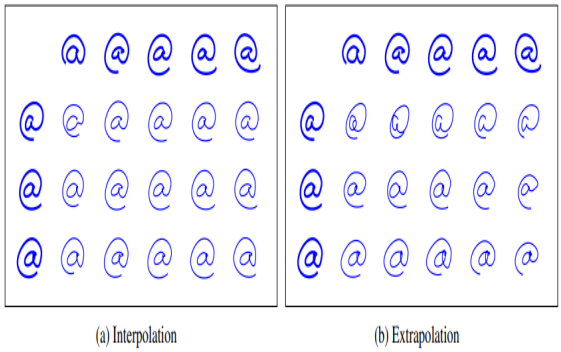

FIGURE 59. Overview of interpolation and extrapolation between handwritten characters. Original characters are shown in bold [94].

Thus, it mimics images as if they are taken at a certain angle where the camera can not capture those images. This method showed effectiveness on several object detection datasets. An example of the proposed data augmentation is shown in Fig. 58.

FIGURE 60. Overview of the original image and two stylized images by STaDA [98].

FIGURE 61. Overview of Style augmentation applied to an image. The shape is preserved but the style, including color, texture, and contrast is randomized [101].

(g) **Deep Adversarial Data Augmentation (DADA) for Extremely Low Data Regimes:** DADA [85] deals the challenge of working with extremely low data regimes, where there is very little labeled data and no unlabeled data available. To deal with that problem, DADA is proposed, where data augmentation is formulated as a problem of training class conditional and supervised GAN. Furthermore, it also introduces new discriminator loss with aim of fitting data augmentation where real and augmented samples are forced to participate equally and be consistent in finding decision boundaries.

3) FEATURE AUGMENTATION

Feature augmentation is another category of data augmentation, where images are transformed into embedding or representation then data augmentation is performed on the embedding of the image. This technique enhances the feature space used for model training by modifying these embeddings, rather than directly altering the raw image data. Recently a few works have been done in this area, we selectively highlight the work in a precise way.

(a) **FeatMatch:** FeatMatch [93] is a novel approach of data augmentation in features space for SSL inspired by an image-based SSL method that uses a combination of augmentation methods of the images and consistency regularization. Image-based SSL methods are restricted to only conventional data augmentation. To break this end, the feature-based SSL method produced diverse features from complex data augmentation methods. One key point is, these advanced data augmentation approaches exploit the information from both intra-class and inter-class representations extracted via clustering. The proposed method only showed significant performance gain on min-Imagenet such as an absolute 17.44% gain on miniImageNet, but also showed robustness on samples that are out-of-distribution. Moreover, the difference between image-level and feature-level augmentation and consistency is shown in Fig. 57.

(b) **Dataset Augmentation in Feature Space:** It [94] first uses encoder-decoder to learn representation, then on representation apply different transformations such as adding noise, interpolating, or extrapolating. The proposed method has shown performance improvement on both static and sequential data. Moreover, a demonstration of this augmentation is shown in Fig. 59.

(c) **Feature Space Augmentation for Long-Tailed Data:** This augmentation [95] is a novel data augmentation in feature space to mitigate the long-tailed issue and uplift the under-represented class samples. The proposed approach first separates class-specific features into generic and specific features with the help of class activation maps. Under-represented class samples are generated by injecting class-specific features of under-represented classes with class-generic features from other confusing classes. It enables diverse data and also deals with the problem of under-represented class samples. It has shown SOTA performance on different datasets. It is demonstrated in Fig. 62.

(d) **Adversarial Feature AugmentationAFA:** GANs showed promising results in unsupervised domain adaptation to learn target domain features indistinguishable from the source domain. AFA [96] extends GAN by contributing: i) it forces feature extractor to be domain-invariant ii) To train it via data augmentation in feature space, named feature augmentation.

(e) **Understanding data augmentation for classification: when to warp?:** It [97] investigates the data augmentation advantages on image space and feature space during training. It proposes two approaches: i) data warping which generates extra samples in image space using data augmentation methods and ii) synthetic over-sampling, which generates samples in feature space. It also suggests that it is possible to apply general data augmentation techniques in feature space if reasonable data augmentation methods for data are known.

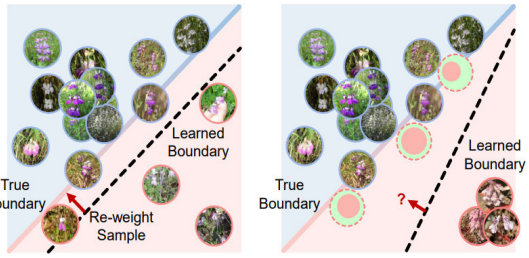

FIGURE 62. Left: limited but well-spread data. Right: Without sufficient data [95].

4) NEURAL STYLE TRANSFER

It is another category of data augmentation, which can transfer the artist style of one image to another without

FIGURE 63. Overview of the styled image by the neural algorithm [103].

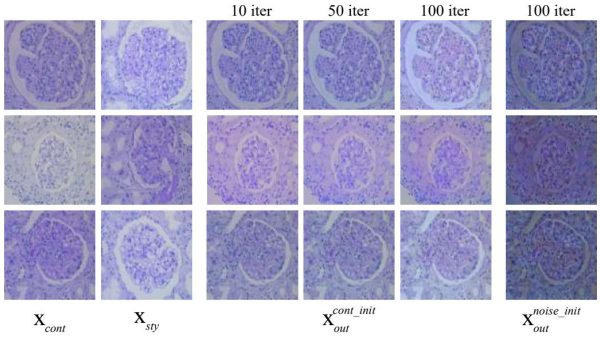

FIGURE 64. Comparison of content and random initialization. Authors observe that output images initialized as the noise appeared distorted and discolored and failed to retain the content fidelity [102].

changing semantics at a high level. It brings more variety to the training set. The main objective of this neural style transfer is to generate a third image from two images, where one image provides texture content and another provides high-level semantic content. We explore some of the SOTA augmentation methods for the sub-category.

(a) **Style Transfer as Data Augmentation (STaDA):** STaDA [98] is thoroughly evaluated different SOTA neural style transfer algorithms as data augmentation named STaDA for image classification tasks. It shows significant performance gain on Caltech 101 [99] and Caltech 256 [100] datasets. Furthermore, it also combines neural style transfer algorithms with conventional data augmentation methods. A sample of this augmentation is shown in Fig. 60.

(b) **Data Augmentation via Style Randomization (SA):** SA [101] is a novel data augmentation, which is based on style neural transfer. SA randomizes the color, contrast, and texture while maintaining the shape and semantic content during the training. This is done by picking an arbitrary style transfer network for randomizing the style and by getting the target style from multivariate normal distribution embedding. It improves performance in three different tasks: classification, regression, and domain adaptation. The style augmentation sample is shown in Fig. 61.

(c) **Style-Transfer Data Augmentation (STDA):** STDA [102] is a novel pipeline for Antibody Mediated Rejection (AMR) classification in kidneys based on StyPath data augmentation. StyPath is data augmentation that transfers style intending to reduce bias. The proposed augmentation is much faster than SOTA augmentation methods for AMR classification. Some samples are shown in Fig. 64.

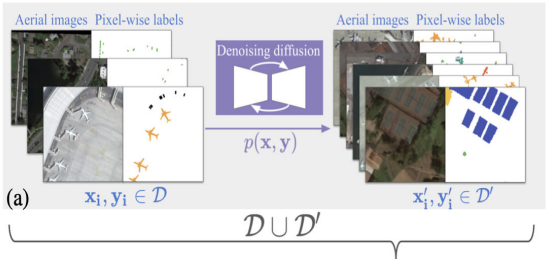

FIGURE 65. SatSynth flow mechanism, image courtesy [104].

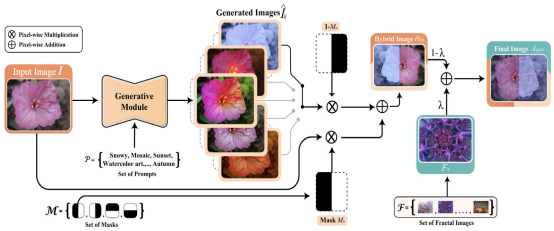

FIGURE 66. DiffuseMix working architecture, image source [105].

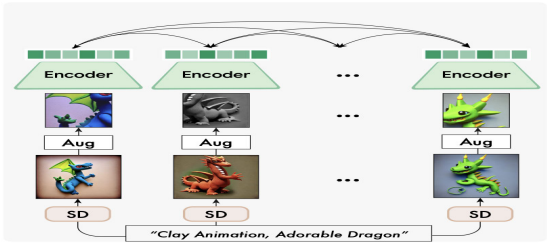

FIGURE 67. StableRep flow mechanism, image courtesy [106].

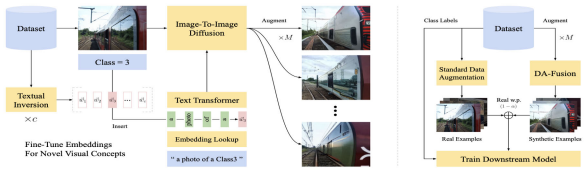

FIGURE 68. Effective data augmentation, image source [107].

(d) **A Neural Algorithm of Artistic Style:** It [103] introduces an artificial system (AS) based on a deep neural network that generates artistic images of high perceptual quality. AS creates neural embedding then it uses the embedding to separate the style and content of the image and then recombines the content and style of target images to generate the artistic image. The sample is shown in Fig. 63

TABLE 1. Effectiveness levels of different augmentation types.

Augmentation Type	Effectiveness Level for Classification	Effectiveness Level for Segmentation	Effectiveness Level for Detection
Adversarial Data Augmentation for Object Detection	High	High	High
Adversarial Feature Augmentation	High	High	High
AugMix	High	High	High
AutoAugment	High	High	High
ClassMix	High	High	High
Context Decoupling Augmentation	High	Medium	High
Cropping and resizing	High	High	High
Cutblur	Medium	Medium	Medium
CutMix	High	High	High
Cutout	High	High	High
Deep Adversarial Data Augmentation for Extremely Low Data Regimes	High	High	High
Deep CNN Ensemble with Data Augmentation for Object Detection	High	High	High
DeepLab-v2 [70]	High	High	High
Deeplabv3+ [115]	High	High	High
ExFuse [116]	High	High	High
Fast AutoAugment	High	High	High
Faster AutoAugment	High	High	High
FeatMatch	High	High	High
Feature Space Augmentation for Long-Tailed Data	High	High	High
FixMatch	High	High	High
FMix	High	High	High
Flipping	High	High	High
GridMask	High	High	High
Hide-and-Seek	High	High	High
Image Erasing	Medium	Medium	Medium
Improved Mixed-Example Data Augmentation	High	High	High
Jitter	Medium	Medium	Medium
Kernel Filter	Low	Low	Low
Learning Data Augmentation Strategies for Object Detection	High	High	High
MixMatch	High	High	High
MixMo	High	Low	High
Mixup	High	Low	High
Noise Injection	Medium	Low	Medium
ObjectAug	High	High	High
Perspective Transformation Data Augmentation	Low	High	Low
Puzzle Mix	High	High	High
RandAugment	High	High	High
Random erasing	Medium	Low	Medium
Random image cropping and patching	Medium	Low	Medium
RangeAugment	High	High	High
Reinforcement Learning with Augmented Data	High	High	High
ResizeMix	Medium	Low	Medium
Rotation	High	Medium	High
RSMDA	Medium	Low	Medium
SaliencyMix	Medium	Medium	Medium
Scale-aware Automatic Augmentation for Object Detection	High	High	High
Shearing	Low	Low	Low
Simple Copy-Paste	Medium	Low	Medium
SnapMix	Medium	Medium	Medium
StyleMix	High	High	High
Translation	Medium	Low	Medium
Xception-65 [115]	High	High	High

5) DIFFUSION DATA AUGMENTATION

This technique generates new data samples by simulating the diffusion process, adding random perturbations followed by smoothing. It creates realistic data variations, enhancing the training set's diversity and improving model robustness and generalization.

(a) **SatSynth:** SatSynth [104] uses denoising diffusion probabilistic models to generate high-quality, diverse image-mask pairs for augmenting training data in aerial semantic segmentation. This approach reduces the need for manual annotations, enhancing model performance by providing more training data that captures the varied scales and frequencies of semantic classes in satellite imagery, as shown in Fig. 65.

(b) **DiffuseMix:** DiffuseMix [105] is a diffusion model-based data augmentation technique that preserves label integrity and avoids unrealistic image generation. By combining partial natural images with their generated counterparts and integrating fractal patterns, this method improves the generalization of deep neural networks and enhances resilience

TABLE 2. Dataset distribution summary.

Dataset Name	Training (Images)	Validation (Images)	Test (Images)
CIFAR-10	50,000	-	10,000
CIFAR-100	50,000	-	10,000
SVHN	73,257	-	26,032
MiniImagenet	50,000	-	10,000
ImageNet	1,200,000	50,000	150,000
PASCAL VOC 2007	4,981	4,981	4,982
PASCAL VOC 2012	11,530	5,379	5,204
COCO2017	118,000	5,000	-
Cityscapes	2,975	503	1,141

TABLE 3. Baseline performance comparison of various augmentation on CIFAR10 and CIFAR100 datasets, image classification. Note: + sign after dataset name shows that traditional data augmentation methods have been used.

	CIFAR-10		CIFAR-100		ImageNet	
Augmentation	Acc(%)	Model	Acc(%)	Model	Acc(%)	Model
Cutout [41]	97.04	WRN-28-10	81.59	WRN-28-10	77.1	ResNet-50
Random Erasing [42]	96.92	WRN-28-10	82.27	WRN-28-10	-	-
RSMDA (R) [54]	92.82	resnet20	69.82	resnet20	-	-
RSMDA (C) [54]	92.62	resnet20	69.72	resnet20	-	-
RSMDA (RC) [54]	92.52	resnet20	69.54	resnet20	-	-
RSMDA (R) [54]	93.68	resnet32	72.20	resnet32	-	-
RSMDA (C) [54]	93.94	resnet32	71.78	resnet32	-	-
RSMDA (RC) [54]	93.79	resnet32	71.58	resnet32	-	-
RSMDA (R) [54]	94.91	resnet44	75.51	resnet44	-	-
RSMDA (C) [54]	94.74	resnet44	74.79	resnet44	-	-
RSMDA (RC) [54]	94.49	resnet44	74.92	resnet44	-	-
RSMDA (R) [54]	94.98	resnet56	76.65	resnet56	-	-
RSMDA (C) [54]	94.72	resnet56	75.67	resnet56	-	-
RSMDA (RC) [54]	94.03	resnet56	75.09	resnet56	-	-
Hide-and-Seek [43]	95.53	ResNet-110	78.13	ResNet-110	77.20	ResNet-50
GridMask [44]	97.24	WRN-28-10	-	-	77.9	ResNet-50
LocalAugment [45]	-	-	95.92	WRN-22-10	76.87	ResNet-50
SalfMix [47]	96.62	PreActResNet-101	80.11	PreActResNet-101	-	-
KeepAugment [48]	97.8	ResNet-28-10	-	-	80.3	ResNet-101
Cut-Thumbnail [50]	97.8	ResNet-56	95.94	WRN-28-10	79.21	ResNet-50
MixUp [52]	97.3	WRN-28-10	82.5	WRN-28-10	77.9	ResNet-50
CutMix [15]	97.10	WRN-28-10	83.40	WRN-28-10	78.6	ResNet-50
SaliencyMix [53]	97.24	WRN-28-10	83.44	WRN-28-10	78.74	ResNet-50
PuzzleMix [55]	-	-	84.05	WRN-28-10	77.51	ResNet-50
FMix [57]	98.64	Pyramid	83.95	Dense	77.70	ResNet-101
MixMo [58]	96.38	WRN-28-10	82.40	WRN-28-10	-	-
StyleMix [59]	96.44	PyramidNet-200	85.83	PyramidNet-200	77.29	PyramidNet-200
RandomMix [60]	98.02	WRN-28-10	84.84	WRN-28-10	77.88	WRN-28-10
MixMatch [61]	95.05	WRN-28-10	74.12	WRN-28-10	-	-
ReMixMatch [62]	94.71	WRN-28-2	-	-	-	-
FixMatch [63]	95.69	WRN-28-2	77.04	WRN-28-2	-	-
AugMix [64]	-	-	-	-	77.6	ResNet-50
Improved Mixed-Example [66]	96.02	ResNet-18	80.3	ResNet-18	-	-
RICAP [67]	97.18	WRN-28-10	82.56	ResNet-28-10	78.62	WRN-50-2
ResizeMix [69]	97.60	WRN-28-10	84.31	WRN-28-10	79.00	ResNet-50
AutoAugment [?]	97.40	WRN-28-10	82.90	WRN-28-10	83.50	AmoebaNet-C
Fast AutoAugment [93]	98.00	SS(26 2×96d)	85.10	SS(26 2×96d)	80.60	ResNet-200
Faster AutoAugment [74]	98.00	SS(26 2 × 112d)	84.40	SS(26 2×96d)	75.90	ResNet-50
Local Patch AutoAugment [76]	98.10	SS(26 2 × 112d)	85.90	SS(26 2×96d)	81.00	ResNet-200
RandAugment [77]	98.50	PyramidNet	83.30	WRN-28-10	85.00	EfficientNet-B7

against adversarial attacks. Overall flow is shown in Fig. 66.

(c) **StableRep:** StableRep [106] explores using synthetic images generated by the text-to-image model Stable Diffusion to learn visual representations, as shwon in Fig. 67. By optimizing the generative model with proper classifier-free guidance and employing a multi-positive contrastive learning method, StableRep achieves representation learning performance that surpasses SimCLR and CLIP on large-scale datasets, solely using synthetic images. This approach demonstrates that synthetic images can effectively train self-supervised methods, matching or exceeding the performance of real image counterparts.

(d) **Effective data augmentation with diffusion models:** It [107] uses to enhance data augmentation by altering high-level semantic attributes in images, as shown in Fig. 68. This method goes beyond simple transformations like rotations and flips, introducing greater diversity by editing images to change their semantics,

TABLE 4. Performance comparison of the various image erasing and image mixing augmentation methods for image classification problems.

CIFAR-10		CIFAR-100		ImageNet	
Augmentation	**Acc(%)**	**Augmentation**	**Acc(%)**	**Augmentation**	**Acc(%)**
SimCLR [110]	84.8	SimCLR [110]	65.2	ReMixMatch [62]	75.90
CLIP [107], [114]	87.3	CLIP [107], [114]	69.5	Hide-and-Seek [43]	77.20
RSMDA (R) [54]	92.52	RSMDA (RC) [54]	69.54	GridMask [44]	77.9
RSMDA (C) [54]	92.62	RSMDA (C) [54]	69.72	MixUp [52]	77.9
RSMDA (RC) [54]	92.82	RSMDA (R) [54]	69.82	AugMix [64]	77.6
RSMDA (RC) [54]	93.79	RSMDA (RC) [54]	71.58	PuzzleMix [55]	77.51
RSMDA (C) [54]	93.94	RSMDA (C) [54]	71.78	FMix [57]	77.70
RSMDA (R) [54]	93.68	RSMDA (R) [54]	72.20	RandomMix [60]	77.88
RSMDA (R) [54]	94.91	MixMatch [61]	74.12	RICAP [67]	78.62
RSMDA (C) [54]	94.74	RSMDA (RC) [54]	74.92	SaliencyMix [53]	78.74
RSMDA (RC) [54]	94.49	RSMDA (C) [54]	74.79	CutMix [15]	78.6
RSMDA (R) [54]	94.98	RSMDA (R) [54]	75.51	DiffuseMix [106]	78.64
RSMDA (C) [54]	94.72	RSMDA (RC) [54]	75.09	KeepAugment [48]	80.3
RSMDA (RC) [54]	94.03	RSMDA (C) [54]	75.67	ResizeMix [69]	79.00
MixMatch [61]	95.05	RSMDA (R) [54]	76.65	Cut-Thumbnail [50]	79.21
FixMatch [63]	95.69	FixMatch [63]	77.04		
Hide-and-Seek [43]	95.53	Hide-and-Seek [43]	78.13		
Improved Mixed-Example [66]	96.02	Improved Mixed-Example [66]	80.3		
SalfMix [47]	96.62	SalfMix [47]	80.11		
ReMixMatch [62]	94.71	Cutout [41]	81.59		
MixMo [58]	96.38	SaliencyMix [53]	83.44		
StyleMix [59]	96.44	RICAP [67]	82.56		
Random Erasing [42]	96.92	Random Erasing [42]	82.27		
GridMask [44]	97.24	MixUp [52]	82.5		
RICAP [67]	97.18	CutMix [15]	83.40		
CutMix [15]	97.10	MixMo [58]	82.40		
Cutout [41]	97.04	PuzzleMix [55]	84.05		
AutoAugment [73]	97.40	RandomMix [60]	84.84		
Fast AutoAugment [93]	98.00	ResizeMix [69]	84.31		
Local Patch AutoAugment [76]	98.10	StyleMix [59]	85.83		
RandAugment [77]	98.50	AutoAugment [73]	82.90		
StableRep [107]	96.2	Fast AutoAugment [93]	85.10		
		Local Patch AutoAugment [76]	85.90		
		StableRep [107]	84.1		

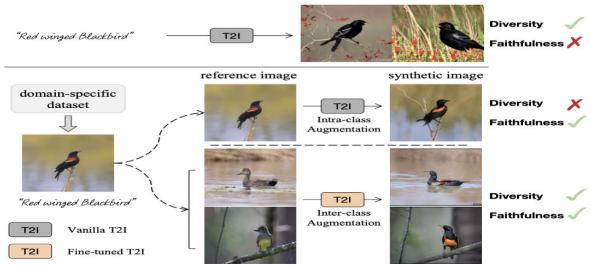

FIGURE 69. Diff-Mix data augmentation providing diversity and faithfulness as compared to other augmentation, image source [108].

such as altering the animal species present. This approach improves model robustness and generalization with minimal labeled examples.

(e) **Diff-Mix:** Diff-Mix [108] uses text-to-image diffusion models for inter-class data augmentation, enhancing image classification by translating images between classes. This method tackles the limitations of traditional augmentations, improving both foreground accuracy and background diversity, and boosting classification performance with diverse, context-rich synthetic images. Moreover overall, flow and comparison are shown in Fig. 69.

III. RESULTS

In this section, we provide the detailed result for various CV tasks such as image classification, object detection, and semantic segmentation. The main purpose is to show the effect of data augmentation on different CV tasks and to do so, we compile results from various SOTA data augmentation works. We have shown effect of the image data augmentation on abstract level for different tasks as shown in Table 1. The used datasets are summarized in Table 2.

A. IMAGE CLASSIFICATION

In this section, we discuss image classification results. Table 3 compares the baseline performance of various augmentation techniques applied to image classification tasks using the CIFAR10 and CIFAR100 datasets. Baseline models like ResNet-18, ResNet-50, and WideResNet-28-10, along with their variants augmented with techniques such as CutOut, Random Erasing, CutMix, and SaliencyMix, are evaluated. The accuracies achieved by each model variant on both datasets reveal insights into the effectiveness of different augmentation methods, with techniques like Random Erasing and SaliencyMix showing substantial improvements, particularly on the CIFAR100 dataset. This table highlights the

TABLE 5. Top-1 Accuracy (%) comparison when 1.2M generated images are used for data augmentation. Models trained only on generated images perform worse than those with real data, but augmenting real data with generated images from a fine-tuned diffusion model significantly boosts performance.

Model	Input Size	Params (M)	Real Only	Generated Only	Real + Generated	Performance Δ
ConvNets						
ResNet-50+DiffuseAugment [111]	224 × 224	36	76.39	69.24	78.17	+1.78
ResNet-101+DiffuseAugment [111]	224 × 224	45	78.15	71.31	79.74	+1.59
ResNet-152+DiffuseAugment [111]	224 × 224	64	78.59	72.38	80.15	+1.56
ResNet-RS-50+DiffuseAugment [111]	160 × 160	36	79.10	70.72	79.97	+0.87
ResNet-RS-101+DiffuseAugment [111]	160 × 160	64	80.11	72.73	80.89	+0.78
ResNet-RS-101+DiffuseAugment [111]	190 × 190	64	81.29	73.63	81.80	+0.51
ResNet-RS-152+DiffuseAugment [111]	224 × 224	87	82.81	74.46	83.10	+0.29
Transformers						
Swin-B+TL-Align [113]	224x224	88	83.5	-	83.7	+0.2
Swin-T+TL-Align [113]	224x224	29	81.2	-	81.4	+0.2
PVT-T+TokenMix [112]	224x224	13.2	75.1	-	75.5	+0.4
CaiT-XXS-24+TokenMix [112]	-	9.5	77.6	-	78.0	+0.4
Swin-S+TL-Align [113]	224x224	50	83.0	-	83.4	+0.4
PVT-S+TL-Align [113]	224x224	24.5	79.8	-	80.4	+0.6
DeiT-S+TL-Align [113]	224x224	22	79.8	-	80.6	+0.8
DeiT-B+DiffuseAugment [111]	384 × 384	87	83.16	75.45	83.75	+0.59
DeiT-L+DiffuseAugment [111]	224 × 224	307	82.22	74.60	83.05	+0.83
DeiT-T+TokenMix [112]	224x224	5.7	72.2	-	73.2	+1.0
DeiT-S+TokenMix [112]	-	22.1	79.8	-	80.8	+1.0
ViT-S/16+DiffuseAugment [111]	224 × 224	22	79.89	71.88	81.00	+1.11
DeiT-B+TokenMix [112]	-	86.6	81.8 (CutMix)	-	82.9	+1.1
DeiT-B+TL-Align [113]	224x224	86	81.8	-	82.3	+0.5
DeiT-B+DiffuseAugment [111]	224 × 224	87	81.79	74.55	82.84	+1.04
DeiT-S+DiffuseAugment [111]	224 × 224	22	78.97	72.26	80.49	+1.52

TABLE 6. Comparison on CIFAR-10 and SVHN. The number represents accuracy.

	CIFAR-10				SVHN			
Method	40 labels	250 labels	1,000 labels	4,000 labels	40 labels	250 labels	1,000 labels	4,000 labels
VAT [119]	-	36.03 ± 2.82	18.64 ± 0.40	11.05 ± 0.31	-	8.41 ± 1.01	5.98 ± 0.21	4.20 ± 0.15
Mean Teacher [120]	-	47.32 ± 4.71	17.32±4.00	10.36±0.25	-	6.45±2.43	3.75±.10	3.39±0.11
MixMatch [61]	47.54±11.50	11.08±.87	7.75±.32	6.24±.06	42.55±14.53	3.78±.26	3.27±.31	2.89±.06
ReMixMatch [62]	19.10±9.64	6.27±0.34	5.73±0.16	5.14±0.04	3.34±0.20	3.10±0.50	2.83±0.30	2.42±0.09
UDA	29.05±5.93	8.76± 0.90	5.87± 0.13	5.29± 0.25	52.63±20.51	2.76± 0.17	2.55± 0.09	2.47± 0.15
SSL with Memory [121]	-	-	-	11.9±0.22	-	8.83	4.21	-
Deep Co-Training [122]	-	-	-	8.35± 0.06	-	-	3.29 ±0.03	-
Weight Averaging [123]	-	-	15.58 ± 0.12	9.05± 0.21	-	-	-	-
ICT [124]	-	-	15.48 ± 0.78	7.29± 0.02	-	4.78 ± 0.68	3.89 ±0.04	-
Label Propagation [125]	-	-	16.93 ± 0.70	10.61 ± 0.28	-	-	-	-
SNTG [126]	-	-	18.41 ± 0.52	9.89 ±0.34	-	4.29± 0.23	3.86 ±0.27	-
PLCB [127]	-	-	6.85 ±0.15	5.97± 0.15	-	-	-	-
Π-model [128]	-	53.02 ±2.05	31.53 ± 0.98	17.41± 0.37	-	17.65 ±0.27	8.60± 0.18	5.57± 0.14
PseudoLabel [129]	-	49.98 ±1.17	30.91 ±1.73	16.21 ± 0.11	-	21.16± 0.88	10.19 ± 0.41	5.71± 0.07
Mixup [52]	-	47.43 ± 0.92	25.72 ± 0.66	13.15 ± 0.20	-	39.97 ± 1.89	16.79 ± 0.63	7.96 ±0.14
FeatMatch [94]	-	7.50 ±0.64	5.76 ±0.07	4.91± 0.18	-	3.34± 0.19	3.10± 0.06	2.62 ±0.08
FixMatch [63]	13.81±3.37	5.07±0.65	-	4.26±0.05	3.96±2.17	2.48±0.38	2.28±0.11	-
SelfMatch [130]	93.19±1.08	95.13±0.26	-	95.94±0.08	96.58±1.02	97.37±0.43	97.49±0.07	-

efficacy of various augmentation strategies in enhancing image classification performance.

Table 4 offers a comprehensive comparison of image erasing and mixing augmentation methods for image classification tasks. The accuracies achieved by different augmentation techniques on datasets like CIFAR-10, CIFAR-100, and mini-ImageNet are reported, evaluating methods such as Cutout, Random Erasing, MixUp, SaliencyMix, FMix, MixMo, and StyleMix. By comparing these methods, insights into their relative effectiveness and suitability for different datasets emerge, guiding researchers and practitioners in selecting the most appropriate augmentation techniques.

Table 5 presents a comparison of different image data augmentations' effects on ViT architectures. Table 6 compares semi-supervised learning methods on CIFAR-10 and SVHN datasets, reporting accuracies under different label budgets (40 to 4,000 labels). Methods like VAT, Mean Teacher, MixMatch, and UDA are evaluated, revealing their performance across varying label budgets and datasets. Notably, some methods significantly improve over supervised learning baselines, especially when labeled data is scarce. This table underscores the importance of semi-supervised learning techniques in scenarios with limited labeled data.

TABLE 7. Comparison on CIFAR-100 and mini-ImageNet. The number represents error rates.

	CIFAR-100			mini-ImageNet	
Method	**400 labels**	**4,000 labels**	**10,000 labels**	**4,000 labels**	**10,000 labels**
Π-model [128]	-	-	39.19± 0.36	-	-
SNTG [126]	-	-	37.97± 0.29	-	-
SSL with Memory [121]	-	-	34.51± 0.61	-	-
Deep Co-Training [122]	-	-	34.63± 0.14	-	-
Weight Averaging [123]	-	-	33.62± 0.54	-	-
Mean Teacher [120]	-	45.36 ±0.49	36.08± 0.51	72.51± 0.22	57.55 ± 1.11
Label Propagation [125]	-	43.73 ±0.20	35.92 ±0.47	70.29± 0.81	57.58 ±1.47
PLCB [127]	-	37.55 ±1.09	32.15 ±0.50	56.49 ±0.51	46.08 ± 0.11
FeatMatch	-	31.06 ± 0.41	26.83 ± 0.04	39.05 ±0.06	34.79±0.22
MixMatch [61]	67.61±1.32	-	28.31±0.33	-	-
UDA	59.28±0.88	-	24.50±0.25	-	-
ReMixMatch [62]	44.28±2.06	-	23.03±0.56	-	-
FixMatch [63]	48.85±1.75	-	22.60±0.12	-	-

TABLE 8. Comparison of test error rates on CIFAR-10 & SVHN using WideResNet-28 and CNN-13.

Approach	**Method**	**CIFAR-10** ($N_l = 4000$)	**SVHN** ($N_l = 1000$)
WideResNet-28			
	Supervised	20.26 ± 0.38	12.83 ± 0.47
Pseudo Labeling	PL [129]	17.78 ± 0.57	7.62 ± 0.29
	PL-CB [127]	6.28 ± 0.3	-
	Π Model [132]	16.37 ± 0.63	7.19 ± 0.27
	Mean Teacher [120]	15.87 ± 0.28	5.65 ± 0.47
Consistency Regularization	VAT [119]	13.86 ± 0.27	5.63 ± 0.20
	VAT + EntMin [119]	13.13 ± 0.39	5.35 ± 0.19
	LGA + VAT [133]	12.06 ± 0.19	6.58 ± 0.36
	ICT [124]	7.66 ± 0.17	3.53 ± 0.07
	MixMatch [61]	6.24 ± 0.06	3.27 ± 0.31
	UDA	5.29 ± 0.25	2.46 ± 0.17
	ReMixMatch [62]	5.14 ± 0.04	2.42 ± 0.09
	FixMatch [63]	4.26 ± 0.05	2.28 ± 0.11
	CL	8.92 ± 0.03	5.65 ± 0.11
Pseudo Labeling	CL+FA [93]	5.51 ± 0.14	2.90 ± 0.19
	CL+FA [93] + Mixup [52]	5.09 ± 0.18	2.75 ± 0.15
	CL+RA+Mixup [52]	5.27 ± 0.16	2.80 ± 0.188
CNN-13			
Pseudo Labeling	TSSDL-MT	9.30 ± 0.55	3.35 ± 0.27
	LP-MT	10.61 ± 0.28	-
	Ladder net [134]	12.36 ± 0.31	-
	MeanTeacher [120]	12.31 ± 0.24	3.95 ± 0.19
	Temporal ensembling [132]	12.16 ± 0.24	4.42 ± 0.16
Consistency Regularization	VAT [119]	11.36 ± 0.34	5.42
	NATEntMin [119]	10.55 ± 0.05	3.86
	SNTG [126]	10.93 ± 0.14	3.86 ± 0.27
	ICT [124]	7.29 ± 0.02	2.89 ± 0.04
Pseudo Labeling	CL	9.81 ± 0.22	4.75 ± 0.28
	CL+RA	5.92 ± 0.07	3.96 ± 0.10

Table 7 and Table 8 compare semi-supervised learning methods on CIFAR-100 and mini-ImageNet datasets, focusing on error rates across different label budgets. Evaluating methods such as PseudoLabel, Label Propagation, Mean Teacher, and MixMatch, these tables provide insights into their effectiveness in leveraging unlabeled data to improve classification accuracy. The findings highlight the potential of semi-supervised learning approaches to alleviate data scarcity and enhance classification efficiency across diverse datasets and label budgets.

B. OBJECT DETECTION

In this section, we explore the effectiveness of various image data augmentation techniques on the COCO2017 [88], PASCAL VOC [89], VOC 2007 [86], and VOC 2012 [87] datasets, which are commonly used for object detection tasks. We compile results from several SOTA data augmentation methods and present them in three tables: Table 9, Table 10, and Table 11. Table 9 shows several classical and automatic data augmentation methods promising performance on the PASCAL VOC dataset. Table 10 illustrates that Faster

TABLE 9. Data augmentation effect on different object detection methods using PASCAL VOC dataset.

Method	Detector	BackBone	AP	AP_50	AP_75	AP_s	AP_m	AP_l
Hand-crafted:								
Dropblock [139]	RetinaNet	ResNet-50	38.4	56.4	41.2	—	—	—
AutoAugment+color Ops [78]	RetinaNet	ResNet-50	37.5	-	-	—	—	—
geometric Ops [78]	RetinaNet	ResNet-50	38.6	-	-	—	—	—
bbox-only Ops [78]	RetinaNet	ResNet-50	39.0	-	-	—	—	—
Mix-up [140]	Faster R-CNN	ResNet-101	41.1	-	-	-	-	-
PSIS* [141]	Faster R-CNN	ResNet-101	40.2	61.1	44.2	22.3	45.7	51.6
Stitcher [142]	Faster R-CNN	ResNet-101	42.1	-	-	26.9	45.5	54.1
GridMask [44]	Faster R-CNN	ResNeXt-101	42.6	65.0	46.5	-	-	-
InstaBoost* [143]	Mask R-CNN	ResNet-101	43.0	64.3	47.2	24.8	45.9	54.6
SNIP (MS test)* [144]	Faster R-CNN	ResNet-101-DCN-C4	44.4	66.2	49.9	27.3	47.4	56.9
SNIPER (MS test)* [145]	Faster R-CNN	ResNet-101-DCN-C4	46.1	67.0	51.6	29.6	48.9	58.1
Traditional Aug [34]	Faster R-CNN	ResNet-101	36.80	58.0	40.0	-	-	-
Traditional Aug* [136]	CenterNet	ResNet-101	41.15	58.01	45.30	-	-	-
Traditional Aug+ [44]	Faster-RCNN	50-FPN (2×)	37.4	58.7	40.5	-	-	-
Traditional Aug+ [44]	Faster-RCNN	50-FPN (2×)+GM(p=0.3)	38.2	60.0	41.4	-	-	-
Traditional Aug+ [44]	Faster-RCNN	50-FPN (2×)+GM(p=0.5)	38.1	60.1	41.2	-	-	-
Traditional Aug+ [44]	Faster-RCNN	50-FPN (2×)+GM(p=0.7)	38.3	60.4	41.7	-	-	-
Traditional Aug+ [44]	Faster-RCNN	50-FPN (2×)+GM(p=0.9)	38.0	60.1	41.2	-	-	-
Traditional Aug+ [44]	Faster-RCNN	50-FPN (4×)	35.7	56.0	38.3	-	-	-
Traditional Aug+ [44]	Faster-RCNN	50-FPN (4×)+GM(p=0.7)	39.2	60.8	42.2	-	-	-
Traditional Aug+ [44]	Faster-RCNN	X101-FPN (1×))	41.2	63.3	44.8	-	-	-
Traditional Aug+ [44]	Faster-RCNN	X101-FPN (2×))	40.4	62.2	43.8	-	-	-
Traditional Aug+ [44]	Faster-RCNN	X101-FPN(2×)+GM(p=0.7))	42.6	65.0	46.5	-	-	-
Traditional Aug+ [44]	Faster-RCNN	X101-FPN(2×)+GM(p=0.7))	42.6	65.0	46.5	-	-	-
KeepAugment: [48]	Faster R-CNN	ResNet50-C4	39.5	—	—	—	—	—
KeepAugment: [48]	Faster R-CNN	ResNet50-FPN	40.7	—	—	—	—	—
KeepAugment: [48]	RetinaNet	ResNet50-FPN	39.1	—	—	—	—	—
KeepAugment: [48]	Faster R-CNN	ResNet101-C4	42.2	—	—	—	—	—
KeepAugment: [48]	Faster R-CNN	ResNet101-FPN	42.9	—	—	—	—	—
KeepAugment: [48]	RetinaNet	ResNet101-FPN	41.2	—	—	—	—	—
DADAAugment: [150]	RetinaNet	ResNet-50	35.9	55.8	38.4	19.9	38.8	45.0
DADAAugment: [150]	RetinaNet	ResNet-50(DADA)	36.6	56.8	39.2	20.2	39.7	46.0
DADAAugment: [150]	Faster R-CNN	ResNet-50	36.6	58.8	39.6	21.6	39.8	45.0
DADAAugment: [150]	Faster R-CNN	ResNet-50 (DADA)	37.2	59.1	40.2	22.2	40.2	45.7
DADAAugment: [150]	Mask R-CNN	ResNet-50	37.4	59.3	40.7	22.2	40.6	46.3
DADAAugment: [150]	Mask R-CNN	ResNet-50(DADA)	37.8	59.6	41.1	22.4	40.9	46.6
AutoAugment: [83]	EfficientDet D0	EfficientNet B0	34.4	52.8	36.7	53.1	40.2	13.9
Det-AdvProp: [83]	EfficientDet D0	EfficientNet B0	34.7	52.9	37.2	54.1	40.6	13.9
AutoAugment: [83]	EfficientDet D1	EfficientNet B1	40.1	59.2	43.2	57.9	45.7	19.9
Det-AdvProp: [83]	EfficientDet D1	EfficientNet B1	40.5	59.2	43.3	58.8	46.2	20.6
AutoAugment: [83]	EfficientDet D2	EfficientNet B2	43.5	62.8	46.6	59.8	48.7	23.9
Det-AdvProp: [83]	EfficientDet D2	EfficientNet B2	43.8	62.6	47.3	61.0	49.6	25.6
AutoAugment: [83]	EfficientDet D3	EfficientNet B3	47.0	66.0	50.8	63.0	51.7	29.8
Det-AdvProp: [83]	EfficientDet D3	EfficientNet B3	47.6	66.3	51.4	64.0	52.2	30.2
AutoAugment: [83]	EfficientDet D4	EfficientNet B4	49.5	68.7	53.7	64.9	54.0	31.9
Det-AdvProp: [83]	EfficientDet D4	EfficientNet B4	49.8	68.6	54.2	65.2	54.2	32.4
AutoAugment: [83]	EfficientDet D5	EfficientNet B5	51.5	70.4	56.0	65.2	56.1	35.4
Det-AdvProp: [83]	EfficientDet D5	EfficientNet B5	51.8	70.7	56.3	66.1	56.2	36.2
Automatic:								
AutoAug-det [78]	RetinaNet	ResNet-50	39.0	-	-	-	-	-
AutoAug-det [78]	RetinaNet	ResNet-101	40.4	-	-	-	-	-
AutoAugment [73]	RetinaNet	ResNet-200	42.1	-	-	-	-	-
AutoAug-det' [78]	RetinaNet	ResNet-50	40.3	60.0	43.0	23.6	43.9	53.8
RandAugmnet* [77]	RetinaNet	ResNet-200	41.9	-	-	-	-	-
AutoAug-det [78]	RetinaNet	ResNet-101	41.8	61.5	44.8	24.4	45.9	55.9
RandAug [77]	RetinaNet	ResNet-101	40.1	-	-	-	-	-
RandAug? [10]	RetinaNet	ResNet-101	41.4	61.4	44.5	25.0	45.4	54.2
Scale-aware AutoAug [79]	RetinaNet	ResNet-50	41.3	61.0	441	25.2	44.5	54.6
Scale-aware AutoAug	RetinaNet	ResNet-101	43.1	62.8	46.0	26.2	46.8	56.7
Scale-aware AutoAug	Faster R-CNN	ResNet-101	44.2	65.6	48.6	29.4	47.9	56.7
Scale-aware AutoAug (MS test)	Faster R-CNN	ResNet-101-DCN-C4	47.0	68.6	52.1	32.3	49.3	60.4
Scale-aware AutoAug	FCOS	ResNet-101	44.0	62.7	47.3	28.2	47.8	56.1
Scale-aware AutoAug	FCOS	ResNeXt-32x8d-101-DCN	48.5	67.2	52.8	31.5	51.9	63.0
Scale-aware AutoAug (1200 size)	FCOS	ResNeXt-32x8d-101-DCN	49.6	68.5	54.1	35.7	52.5	62.4
Scale-aware AutoAug (MS Test)	ResNeXt-32x8d-101-DCN	FCOS	51.4	69.6	57.0	37.4	54.2	65.1

R-CNN (FRCNN), combined with synthetic data, achieves the best mean Average Precision (mAP) on the VOC 2007 dataset. Table 11 shows that DetAdvProp achieves the highest scores on the VOC 2012 dataset, outperforming AutoAugment [73]. The results are evaluated using mean Average Precision (mAP), Average Precision (AP) at Intersection over Union (IOU) of 0.5 (AP50), and AP at IOU of 0.75 (AP75) metrics. This detailed comparison provides valuable insights into the performance and suitability of different data augmentation techniques for object detection tasks on these datasets.

C. SEMANTIC SEGMENTATION

This subsection presents the results of semantic segmentation on the PASCAL VOC and Cityscapes datasets, which are frequently used in various research studies. We compiled the validation set results in Table 12 and Table 13, showing the impact of SOTA data augmentation methods on the semantic segmentation task. The results are reported in terms of mean Intersection over Union (mIoU), which measures accuracy on the Cityscapes dataset, and on the PASCAL VOC dataset, as shown in Table 12 and Table 13, respectively. We observed performance gains in several metrics, including mIoU and mean Average Precision (mAP), with a variety of semantic segmentation models such as Deeplabv3+ [114], DeepLab-v2 [70], Xception-65 [114], ExFuse [115], and Efficient-L2 [116]. Incorporating data augmentation techniques has been shown to enhance the performance of these models significantly. Notably, advanced image data augmentation methods have demonstrated greater improvements in performance compared to traditional techniques. Table 12 and Table 13 provide detailed evidence of these improvements. The traditional data augmentation methods include rotation, scaling, flipping, and shifting [72]. By comparing these methods, we highlight the significant contributions that advanced data augmentation techniques make to the accuracy and efficiency of semantic segmentation models on these benchmark datasets. This comprehensive analysis underscores the importance of selecting appropriate augmentation strategies to optimize model performance for semantic segmentation tasks.

IV. DISCUSSION

A. UNDERFITTING CAUSED BY DATA AUGMENTATION

Underfitting occurs when a model is unable to capture the underlying trends in data, often because of its simplicity or limited capacity. Data augmentation techniques are mostly used for mitigation of overfitting via bringing in heterogeneity into training data; however, they can also indirectly impact underfitting.

Here are some of the ways that data augmentation might lead to underfitting:

- **Over-regularization:** Augmentation techniques that are excessive or wrongly applied may cause over-regularization thus limiting the ability of the model to learn from training. This results in average performance and possible under-fitting.
- **Loss of Information:** For instance, some augmentation strategies such as aggressive transformations or distortions may introduce noise and other unnecessary variations to the data. In case these variations hide vital features or patterns necessary for learning by the model it might not generalize well leading to under-fitting.
- **Inadequate Diversity:** If the techniques for improvement are not sufficient to diversify the training data, then during training process it may be impossible to model encounter a wide range of scenarios. This limited exposure to diverse examples can hamper the model's ability to generalize well to unseen data, contributing to underfitting.

While primarily meant for dealing overfitting, the way in which data augmentation impacts on underfitting is usually indirect and is influenced by factors such as over-regularization, loss of information as well as lack of diversity in augmented data. Therefore, appropriate adoption and use of augmentation techniques is necessary so that we strike a balance between preventing overfitting and mitigating underfitting.

B. CURRENT APPROACHES

Currently, image mixing methods and AutoAugment methods are successful for image classification tasks, *scale aware based AutoAugment* methods are showing promising results in detection tasks and semantic segmentation tasks. But these data augmentation performances can vary with the type of data augmentation applied, as it is known that combined data augmentation methods show better performance than single one [34], [117]. Additionally, effectiveness level for each task is shown in Table 1, sorted alphabetically by the Augmentation Category.

C. THEORETICAL FOUNDATIONS OF DATA AUGMENTATION

Data augmentation methods are based on several theoretical concepts that explain the advantage of data augmentation.

1) DATA DIVERSITY AND GENERALIZATION

The main reason behind data augmentation is to increase the variability of the training data set artificially. The model is able to see more examples because it performs rotations, translations, and color modifications on the model. This helps in reducing overfitting as the model learns better to predict data it has never seen before. Prior research has also found that a larger and more diverse training set enables a smoother approximation of the underlying data distribution, thus enhancing the overall generalization of the model [183].

2) REGULARIZATION EFFECT

Data augmentation can also be considered as a regularizer since it involves adding some noise into the training process.

TABLE 10. VOC 2007 test detection average precision (%). FRCN* refers to FRCN with training schedule in [146] and SD refers to synthetic data.

Method	TSet	mAP	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
FRCN [146]	7	66.9	74.5	78.3	69.2	53.2	36.6	77.3	78.2	82.0	40.7	72.7	67.9	79.6	79.2	73.0	69.0	30.1	65.4	70.2	75.8	65.8
FRCN* [147]	7	69.1	75.4	80.8	67.3	59.9	37.6	81.9	80.0	84.5	50.0	77.1	68.2	81.0	82.5	74.3	69.9	28.4	71.1	70.2	75.8	66.6
ASDN [147]	7	71.0	74.4	81.3	67.6	57.0	46.6	81.0	79.3	86.0	52.9	75.9	73.7	82.6	83.2	77.7	72.7	37.4	66.3	71.2	78.2	74.3
IRE	7	70.5	75.9	78.9	69.0	57.7	46.4	81.7	79.5	82.9	49.3	76.9	67.9	81.5	83.3	76.7	73.2	40.7	72.8	66.9	75.4	74.2
ORE	7	71.0	75.1	79.8	69.7	60.8	46.0	80.4	79.0	83.8	51.6	76.2	67.8	81.2	83.7	76.8	73.8	43.1	70.8	67.4	78.3	75.6
I+ORE	7	71.5	76.1	81.6	69.5	60.1	45.6	82.2	79.2	84.5	52.5	78.7	71.6	80.4	83.3	76.7	73.9	39.4	68.9	69.8	79.2	77.4
FRCN [146]	7+12	70.0	77.0	78.1	69.3	59.4	38.3	81.6	78.6	86.7	42.8	78.8	68.9	84.7	82.0	76.6	69.9	31.8	70.1	74.8	80.4	70.4
FRCN* [147]	7+12	74.8	78.5	81.0	74.7	67.9	53.4	85.6	84.4	86.2	57.4	80.1	72.2	85.2	84.2	77.6	76.1	45.3	75.7	72.3	81.8	77.3
IRE	7+12	75.6	79.0	84.1	76.3	66.9	52.7	84.5	84.4	88.7	58.0	82.9	71.1	84.8	84.4	78.6	76.7	45.5	77.1	76.3	82.5	76.8
ORE	7+12	75.8	79.4	81.6	75.6	66.5	52.7	85.5	84.7	88.3	58.7	82.9	72.8	85.0	84.3	79.3	76.3	46.3	76.3	74.9	86.0	78.2
I+ORE	7+12	76.2	79.6	82.5	75.7	70.5	55.1	85.2	84.4	88.4	58.6	82.6	73.9	84.2	84.7	78.8	76.3	46.7	77.9	75.9	83.3	79.3
SSD	7+12	77.4	81.7	85.4	75.7	69.6	49.9	84.9	85.8	87.4	61.5	82.3	79.2	86.6	87.1	84.7	78.9	50.0	77.4	79.1	86.2	76.3
SSD + SD(1x)	7+12	78.1	83.2	84.5	76.1	72.1	50.2	85.2	86.3	87.8	63.7	82.8	80.1	85.2	87.2	84.8	80.0	51.5	77.0	82.0	86.1	76.9
SSD + SD(2x)	7+12	78.3	83.6	85.0	76.2	72.0	51.3	85.1	87.2	87.6	64.2	82.5	81.9	85.5	86.5	85.9	81.2	51.2	72.3	82.8	86.9	78.4
SSD + SD(3x)	7+12	77.8	80.4	85.0	76.3	70.1	50.4	84.8	86.3	88.2	61.0	83.5	79.5	87.2	86.9	85.9	78.8	51.2	76.9	79.4	86.5	77.9
FRCN	7+12	73.2	76.5	79.0	70.9	65.5	52.1	83.1	84.7	86.4	52.0	81.9	65.7	84.8	84.6	77.5	76.7	38.8	73.6	73.9	83.0	72.6
FRCN + SD(1x)	7	79.9	85.1	86.6	78.6	75.7	65.2	83.5	88.4	88.9	65.8	83.6	74.3	86.4	84.7	85.5	88.0	62.0	75.5	75.3	87.7	76.3

TABLE 11. Results on PASCAL VOC 2012. The proposed DetAdvProp gives the highest score on every model and metric. It largely outperforms AutoAugment [73] when facing domain shift.

Model	mAP	AP50	AP75
EfficientDet-D0	55.6	77.6	61.4
+ AutoAugment	55.7 (+0.1)	77.7 (+0.1)	61.8 (+0.4)
+ Det-AdvProp	55.9 (+0.3)	77.9 (+0.3)	62.0 (+0.6)
EfficientDet-D1	60.8	82.0	66.7
+ AutoAugment	61.0 (+0.2)	82.2 (+0.2)	67.2 (+0.5)
+ Det-AdvProp	61.2 (+0.4)	82.3 (+0.3)	67.4 (+0.7)
EfficientDet-D2	63.3	83.6	69.3
+ AutoAugment	62.7 (-0.6)	83.3 (-0.3)	69.2 (-0.1)
+ Det-AdvProp	63.5 (+0.2)	83.8 (+0.2)	69.7 (+0.4)
EfficientDet-D3	65.7	85.3	71.8
+ AutoAugment	65.2 (-0.5)	85.1 (-0.2)	71.3 (-0.5)
+ Det-AdvProp	66.2 (+0.5)	85.9 (+0.6)	72.5 (+0.7)
EfficientDet-D4	67.0	86.0	73.0
+ AutoAugment	67.0 (+0.0)	86.3 (+0.3)	73.5 (+0.5)
+ Det-AdvProp	67.5 (+0.5)	86.6 (+0.6)	74.0 (+1.0)
EfficientDet-D5	67.4	86.9	73.8
+ AutoAugment	67.6 (+0.2)	87.2 (+0.3)	74.2 (+0.4)
+ Det-AdvProp	68.2 (+0.8)	87.6 (+0.7)	74.7 (+0.9)

This noise hinders the model from learning the training data and forces the model to focus on more general information. From the theory perspective, it is similar to adding a regularization term to the loss function, which would punish the complex learning models in favor of simpler and more general solutions [184]. Dropout and weight decay methods achieve a similar goal but, unlike it, data enhancement manipulates the initial data, which is often more reasonable and efficient.

3) MANIFOLD LEARNING

The input data in computer vision tasks can be significantly high dimensional but the data lies on low dimensional manifold. Data augmentation allows to better cover this manifold by generating new data samples that conform to the same underlying structure. This is particularly important as it enhances the performance of the model in learning the manifold and hence improving performance in applications such as image classification and object detection [185], [186]. Recent works on the theoretical properties of Manifold Learning also suggest that better generalization can be achieved by sampling more of the manifold.

4) INVARIANT REPRESENTATIONS

Most data augmentation methods attempt to generate translations in which the model should not change its prediction. For instance, a model that predicts the category of an image should predict an object's class independent from its orientation and lighting. Through training on mixtures of augmented data, the model is encouraged to learn invariant representations that are necessary for robustness. This can be related to theoretical concept of equivariance which states that specific changes in input should lead to specific changes in the result [187], [188].

5) INFORMATION THEORY

From an information theory point of view, the idea can be interpreted as one of maximizing the conditional mutual information between the input and the desired output. The model can then hold more features that relate to the target labels when more variant examples are offered. This mutual information is able to increase performance capabilities and generalization of a model since the knowledge about the input-output dependency is improved [189], [190].

Empirical Evidence: Many empirical researches have also affirmed the theoretical background of data augmentation. For example, experiments demonstrate that additional training data may increase the performance of models on benchmark like CIFAR-10 and ImageNet. These empirical results provide strong support for the theoretical principles discussed above.

D. TRADE-OFFS BETWEEN COMPUTATIONAL COMPLEXITY AND PERFORMANCE GAINS

In this subsection, we explore the trade-offs between computational complexity and performance gains for different data augmentation techniques. Data augmentation enhances model generalization but comes with varying computational costs. Understanding these trade-offs is crucial for selecting the most appropriate augmentation methods, especially in resource-constrained environments.

Table 14 summarizes the performance gains and computational complexity associated with different data augmentation techniques. From Table 14, we observe that augmentation techniques such as geometric image augmentation and non-geometric augmentation offer moderate performance gains with low computational complexity. These methods, including rotation, translation, flipping, cropping, and resizing, are computationally efficient and can provide significant improvements in model generalization.

Color space augmentation and image erasing techniques, which include jitter, kernel filter, cutout, and random erasing, offer higher performance gains at a slightly increased computational complexity. These methods introduce variations in color and texture, enhancing the robustness of models against variations in input data.

Image mixing techniques, such as local augment, self-augmentation, and mixup, provide high performance gains but require medium to high computational resources. These methods involve complex operations such as blending multiple images or patches, which can be computationally intensive.

AutoAugment techniques, including fast AutoAugment, faster AutoAugment, and RandAugment, offer high performance gains but come with high computational complexity. These methods involve automated policy search or augmentation strength adjustment, which requires significant computational resources.

Feature-based augmentation methods, such as FeatMatch and dataset augmentation in feature space, provide moderate

TABLE 12. Results of Performance (mIoU) on Cityscapes validation set, sorted by Full column in ascending order.

Method	Model	1/8	1/4	1/2	7/8	Full
DST-CBC [154]	DeepLab-v2	48.7	60.5	64.4	-	-
French et al [153]	DeepLab-v2	51.20	60.34	63.87	-	-
ClassMix-Seg [70]	DeepLab-v2	54.07	61.35	63.63	66.29	-
s4GAN [152]	DeepLab-v2	-	59.3	61.9	-	65.8
Adversarial [151]	DeepLab-v2	-	58.8	62.3	65.7	-
DeepLab V3plus [72]	MobileNet	-	-	-	-	72.6
Baseline+ CutMix (p = 1) [72]	MobileNet	-	-	-	-	72.8
Baseline+ ObjectAug [72]	MobileNet	-	-	-	-	73.5
DeepLab V3plus [72]	MobileNet	-	-	-	-	73.5
ECS [155]	DeepLabv3Plus	67.4	70.7	72.9	-	74.8
ClassMix [70]	DeepLabV2	61.4	63.6	66.3	-	66.2
CutMix [153]	DeepLabV2	60.3	63.87	-	-	67.7
S4GAN + MT [152]	DeepLabV2	59.3	61.9	-	-	65.8
SSBN [115]	DeepLabV3Plus	74.1	77.8	78.7	-	78.7
DSBN [115]	DeepLabV2	67.6	69.3	70.7	-	70.1
AdvSemi [151]	DeepLabV2	58.8	62.3	65.7	-	66.0
DST-CBC [154]	DeepLabV2	60.5	64.4	-	-	66.9
ECS [155]	DeepLabv3Plus	67.4	70.7	72.9	-	74.8
SDA [115]	DeepLabV3Plus	74.1	-	-	-	-
SDA + DSBN [115]	DeepLabV3Plus	69.5	-	-	-	-
SDA + DSBN [115]	DeepLabV3Plus	69.5	-	-	-	-
SDA [115]	DeepLabV3Plus	-	-	-	-	78.7
SDA + DSBN [115]	DeepLabV3Plus	-	-	-	-	79.2
SDA [115]	DeepLabV3Plus	-	-	-	71.4	-
SDA + DSBN [115]	DeepLabV3Plus	-	-	-	72.5	-
DeepLab V3plus [72]	ResNet-50	-	-	-	-	76.9
DeepLab V3plus [72]	ResNet-101	-	-	-	-	78.5

to high performance gains with medium computational complexity. These techniques involve augmenting features extracted from images, which can enhance model robustness without requiring excessive computational resources.

Neural style transfer techniques, including neural transferable style, data augmentation via style randomization, and style-transfer data augmentation, offer high performance gains but require high computational complexity. These methods involve synthesizing images with specific artistic styles, which can significantly impact model performance but involve computationally intensive style transfer processes.

Generative augmentation methods, such as satSynth and DiffusionMix, provide high performance gains but come with high computational complexity. These techniques involve generating synthetic data using generative models, which can enhance model generalization but require substantial computational resources for training.

By analyzing the computational costs and performance benefits of each technique, practitioners can make informed decisions based on their specific constraints and objectives. This balanced approach ensures that the chosen data augmentation strategy aligns with the available resources while maximizing model performance.

E. FUTURE DIRECTIONS

In this subsection, we provide research questions as future directions. Despite the success of data augmentation techniques in different Computer Vision tasks, there are still unresolved challenges in SOTA data augmentation techniques. After thoroughly reviewing SOTA data augmentation approaches, we have identified several challenges and difficulties that remain to be dealt, as listed below:

- **Label Smoothing in Image Manipulation and Erasing:** In image mixing techniques, label smoothing has been effectively used, where the mixing of image portions corresponds to the mixing of labels. However, this concept has not been explored for image manipulation and image erasing subcategories, where parts of the image are removed. For example, in cutout data augmentation, where a portion of the image is randomly cut out, the corresponding label should also be adjusted. Investigating label smoothing for these subcategories presents an interesting open research question and could potentially improve model performance by providing more accurate label representations.
- **Importance-Based Data Augmentation:** Currently, data augmentation is applied uniformly across all training examples without considering the difficulty level of each example. However, not all examples are equally challenging for the neural network to learn. Applying augmentation selectively to more difficult examples, based on their importance or difficulty, could lead to better model performance. Future research could explore methods to quantify example difficulty and develop strategies for targeted data augmentation. Understanding how neural networks behave when augmented data is applied selectively to difficult examples could provide new insights into training more robust models.

TABLE 13. Results of Performance mean intersection over union (mIoU) on the Pascal VOC 2012 validation set.

Method	Model	1/100	1/50	1/20	1/8	1/4	Full
GANSeg [156]	VGG16	-	-	-	-	64.1	
AdvSemSeg [151]	ResNet-101	-	-	-	-	68.4	
CCT [157]	ResNet-50	-	-	-	-	69.4	
PseudoSeg [158]	ResNet-101	-	-	-	-	73.2	
DSBN [115]	ResNet-101	-	-	-	-	75.0	
DSBN [115]	Xception-65	-	-	-	-	79.3	
Fully supervised [115]	ResNet-101	-	-	-	-	78.3	
Fully supervised [115]	Xception-65	-	-	-	-	79.2	
Adversarial [151]	DeepLab-v2	-	57.2	64.7	69.5	72.1	-
s4GAN [152]	DeepLab-v2	-	63.3	67.2	71.4	-	75.6
French et.el [153]	DeepLab-v2	53.79	64.81	66.48	67.60	-	-
DST-CBC [154]	DeepLab-v2	61.6	65.5	69.3	70.7	71.8	-
ClassMix:Seg* [70]	DeepLab-v2	54.18	66.15	67.77	71.00	72.45	-
Mixup [52]	IRNet	-	-	-	-	-	49
CutOut [41]	IRNet	-	-	-	-	-	48.9
CutMix [15]	IRNet	-	-	-	-	-	49.2
Random pasting [71]	IRNet	-	-	-	-	-	49.8
CCNN [170]	VGG16	-	-	-	-	-	35.6
SEC [171]	VGG16	-	-	-	-	-	51.1
STC [172]	VGG16	-	-	-	-	-	51.2
AdvEra [173]	VGG16	-	-	-	-	-	55.7
DCSP [174]	ResNet101	-	-	-	-	-	61.9
MDC [175]	VGG16	-	-	-	-	-	60.8
MCOF [176]	ResNet101	-	-	-	-	-	61.2
DSRG [177]	ResNet101	-	-	-	-	-	63.2
AffinityNet [178]	ResNet-38	-	-	-	-	-	63.7
IRNet [179]	ResNet50	-	-	-	-	-	64.8
FickleNet [180]	ResNet101	-	-	-	-	-	65.3
SEAM [181]	ResNet38	-	-	-	-	-	65.7
ICD [182]	ResNet101	-	-	-	-	-	64.3
IRNet + CDA [71]	ResNet50	-	-	-	-	-	66.4
SEAM + CDA [71]	ResNet38	-	-	-	-	-	66.8
DeepLab V3 [72]	MobileNet	-	-	-	-	-	71.9
DeepLab V3 [72]	ResNet-50	-	-	-	-	-	77.8
DeepLab V3 [72]	ResNet-101	-	-	-	-	-	78.4
DeepLab V3plus [72]	MobileNet	-	-	-	-	-	73.8
DeepLab V3plus [72]	ResNet-50	-	-	-	-	-	78.8
DeepLab V3plus [72]	ResNet-101	-	-	-	-	-	79.6
Baseline+R.Rotation [72]	ObjectAug	-	-	-	-	-	69.5
Baseline +R.Scaling [72]	ObjectAug	-	-	-	-	-	70.3
Baseline + R.Flipping [72]	ObjectAug	-	-	-	-	-	69.6
Baseline + R.Shifting [72]	ObjectAug	-	-	-	-	-	70.7
Baseline + All [72]	ObjectAug	-	-	-	-	-	73.8
Baseline + CutOut (16×16, p = 0.5) [72]	MobileNet	-	-	-	-	-	71.9
Baseline + CutOut (16×16, p = 1) [72]	MobileNet	-	-	-	-	-	72.3
Baseline + CutMix (p = 0.5) [72]	MobileNet	-	-	-	-	-	72.7
Baseline + CutMix (p = 1) [72]	MobileNet	-	-	-	-	-	72.4
Baseline + ObjectAug [72]	MobileNet	-	-	-	-	-	73.8
Baseline + CutOut (16×16, p=0.5) + ObjectAug [72]	MobileNet	-	-	-	-	-	73.9
Baseline + CutMix (p=0.5) + ObjectAug [72]	MobileNet	-	-	-	-	-	74.1
DeepLabv3+ [183]	EfficientNet-B7	-	-	-	-	-	84.6
ExFuse [116]	EfficientNet-B7	-	-	-	-	-	85.8
Eff-B7 [117]	EfficientNet-B7	-	-	-	-	-	85.2
Eff-L2 [117]	EfficientNet-B7	-	-	-	-	-	88.7
Eff-B7 NAS-FPN [65]	EfficientNet-B7	-	-	-	-	-	83.9
Eff-B7 NAS-FPN w/ Copy-Paste pre-training [65]	EfficientNet-B7	-	-	-	-	-	86.6

- **Mixing Multiple Salient Parts in Image Mixing Techniques:** In current image mixing data augmentation approaches, such as RICAP [67], the mixing of image parts often involves more than two images. However, the impact of mixing truly salient parts from multiple images on model accuracy and robustness against adversarial attacks has not been thoroughly investigated. Future research should explore the effects of mixing salient parts of more than two images, with corresponding label adjustments, to determine its influence on model performance and adversarial robustness.
- **Order of Augmentation Methods in Auto Augmentation:** In the realm of random data augmentation, particularly within the auto augmentation category, the sequence in which augmentation methods are applied has not been extensively studied. The order

TABLE 14. Performance Gains vs. computational complexity of data augmentation techniques.

Augmentation Category	Performance Gain	Computational Complexity
AutoAugment	High	High
Color Space Augmentation	High	Medium
Feature-based Augmentation	Moderate-High	Medium
Generative diffusion Augmentation	High	High
Geometric Image Augmentation	Moderate	Low
Image Erasing	High	Medium
Image Mixing	High	Medium-High
Neural Style Transfer	High	High
Non-Geometric Augmentation	Moderate	Low

of augmentation methods may significantly impact model performance. Exploring various sequences, such as applying traditional data augmentation methods followed by image mixing, or ordering methods based on their performance impact, could uncover optimal augmentation strategies. Researching the potential benefits of different augmentation orders could lead to more effective and efficient data augmentation pipelines.

- **Optimal Order and Number of Augmented Samples:** Finding the optimal order of data augmentation methods and the optimal number of samples to augment are open challenges, particularly in methods like randAugment. The sequence in which augmentation methods are applied and the number of augmented samples can significantly impact model performance. Future research should aim to develop systematic approaches to determine these optimal parameters. Tackling these challenges could enhance the effectiveness of data augmentation in improving model accuracy and robustness.
- **Generative adversarial networks and variational autoencoders (VAEs):** GANs and VAEs are the methods for generating synthetic data which can help to increase the diversity of the training datasets and thus make the obtained models more general. GAN-based augmentation methods have the ability to generate a high degree of realism and diversity, which is highly useful for developing robust models [37], [38]. VAEs further offer a probabilistic framework that allows a new set of data to be generated using a latent representation of the given input. These generative models can be particularly interesting in scenarios when labeled data is scarce.
  However, these approaches have both limitations. One such example is that training GANs might be problematic due to concerns such as mode collapse or instability. They need to be carefully tuned and used with a huge amount of computational resources. VAEs are more stable which might result in less sharp image quality compared to GANs, which might impact the application.
- **Augmentation for spaces:** Most of the data augmentation approaches have been explored on the image level - data space. Very few research works have explored data on feature level - feature space. The challenge here arises, in which space should we apply data augmentation, data space, or feature space? It is another interesting aspect that can be explored. For the current approaches, it seems like it depends on the dataset, model architecture, and task. Currently, approaches are conducting experiments in data space and feature space and then selecting the best one [97]. It is not the optimal way to find data augmentation for specific space. It is still an open challenge to be solved.
- **Choosing Data Augmentation Techniques**
  The selection of data augmentation techniques plays a crucial role in enhancing model performance and generalization across various computer vision tasks. This decision is heavily influenced by both the characteristics of the dataset and the architecture of the model being employed [11]. For instance, in datasets such as MNIST [40], geometric transformations like rotations can pose challenges as they may alter labels of digits like 6 and 9. Conversely, densely parameterized CNNs are more susceptible to overfitting on weakly augmented datasets, whereas shallowly parameterized CNNs may benefit from more aggressive data augmentation strategies. Current practices typically involve empirical validation through extensive experimentation to determine the optimal combination of model architecture and data augmentation techniques. However, there is a critical need for systematic approaches that integrate dataset-specific characteristics and model architecture considerations into the selection process. This systematic approach would ensure that data augmentation strategies are tailored to maximize model performance and robustness across different application domains. Tackling these challenges will advance our understanding of how to effectively leverage data augmentation to improve the efficiency and effectiveness of deep learning models in real-world applications.

V. CONCLUSION

This survey provides an extensive review of modern image data augmentation methods aimed at mitigating overfitting in computer vision tasks. By categorizing these methods and exploring their fundamental principles, we offer a comprehensive understanding of their applicability across diverse domains within computer vision. We have collected the result to show their effectiveness through comprehensive tables, demonstrating their impact on tasks such as image

classification, object detection, and semantic segmentation. These evaluations underscore the pivotal role of data augmentation in improving model generalization and performance across different datasets and model architectures. Our analysis covers both supervised and semi-supervised learning scenarios, facilitating a nuanced comparison and highlighting similarities across advanced algorithms tailored for different learning paradigms. Moreover, we have identified current limitations in existing data augmentation techniques, suggesting avenues for future research and development. By integrating more results and their associated discussions from the tables presented in this paper, our conclusion reinforces the significance of advanced data augmentation methods in advancing the state-of-the-art in computer vision and underscores their potential for further enhancing model robustness and accuracy in real-world applications.

REFERENCES

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[2] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1–11.

[3] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[4] A. M. Roy, J. Bhaduri, T. Kumar, and K. Raj, "WilDect-YOLO: An efficient and robust computer vision-based accurate object localization model for automated endangered wildlife detection," *Ecol. Informat.*, vol. 75, Jul. 2023, Art. no. 101919.

[5] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[6] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[7] J. Kuruvilla, D. Ukumaran, A. Sankar, and S. Joy, "A review on image processing and image segmentation," in *Proc. Int. Conf. Data Mining and Adv. Comput. (SAPIENCE)*, 2016, pp. 198–203.

[8] V. Lempitsky, P. Kohli, C. Rother, and T. Sharp, "Image segmentation with a bounding box prior," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep. 2009, pp. 277–284.

[9] J. Liew, Y. Wei, W. Xiong, S.-H. Ong, and J. Feng, "Regional interactive image segmentation networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2746–2754.

[10] X. Liu, Z. Deng, and Y. Yang, "Recent progress in semantic image segmentation," *Artif. Intell. Rev.*, vol. 52, no. 2, pp. 1089–1106, Aug. 2019.

[11] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 1, pp. 1–48, Dec. 2019.

[12] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[13] T. Kumar, J. Park, M. S. Ali, A. F. M. S. Uddin, J. H. Ko, and S.-H. Bae, "Binary-classifiers-enabled filters for semi-supervised learning," *IEEE Access*, vol. 9, pp. 167663–167673, 2021.

[14] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[15] S. Yun, D. Han, S. Chun, S. J. Oh, Y. Yoo, and J. Choe, "CutMix: Regularization strategy to train strong classifiers with localizable features," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6022–6031.

[16] D. Hendrycks, K. Zhao, S. Basart, J. Steinhardt, and D. Song, "Natural adversarial examples," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 15257–15266.

[17] Z. Zhao, D. Dua, and S. Singh, "Generating natural adversarial examples," 2017, *arXiv:1710.11342*.

[18] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," 2017, *arXiv:1706.06083*.

[19] M. Turab, T. Kumar, M. Bendechache, and T. Saber, "Investigating multi-feature selection and ensembling for audio classification," 2022, *arXiv:2206.07511*.

[20] L. Nanni, G. Maguolo, and M. Paci, "Data augmentation approaches for improving animal audio classification," *Ecol. Informat.*, vol. 57, May 2020, Art. no. 101084.

[21] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *Proc. 16th Annu. Conf. Int. Speech Commun. Assoc.*, Sep. 2015, pp. 1–26.

[22] A. Chandio, Y. Shen, M. Bendechache, I. Inayat, and T. Kumar, "AUDD: Audio Urdu digits dataset for automatic audio Urdu digit recognition," *Appl. Sci.*, vol. 11, no. 19, p. 8842, Sep. 2021.

[23] J. Park, T. Kumar, and S. Bae, "Search of an optimal sound augmentation policy for environmental sound classification with deep neural networks," in *Proc. Korean Soc. Broadcast Eng. Conf.*, 2020, pp. 18–21.

[24] S. Y. Feng, V. Gangal, D. Kang, T. Mitamura, and E. Hovy, "GenAug: Data augmentation for finetuning text generators," 2020, *arXiv:2010.01794*.

[25] P. Liu, X. Wang, C. Xiang, and W. Meng, "A survey of text data augmentation," in *Proc. Int. Conf. Comput. Commun. Netw. Secur. (CCNS)*, Aug. 2020, pp. 191–195.

[26] C. Shorten, T. M. Khoshgoftaar, and B. Furht, "Text data augmentation for deep learning," *J. Big Data*, vol. 8, no. 1, pp. 1–34, Dec. 2021.

[27] M. Bayer, M.-A. Kaufhold, and C. Reuter, "A survey on data augmentation for text classification," *ACM Comput. Surveys*, vol. 55, no. 7, pp. 1–39, Jul. 2023.

[28] S. Y. Feng, V. Gangal, J. Wei, S. Chandar, S. Vosoughi, T. Mitamura, and E. Hovy, "A survey of data augmentation approaches for NLP," in *Findings of The Association for Computational Linguistics: ACL-IJCNLP 2021*. Association for Computational Linguistics, 2021, pp. 968–988. [Online]. Available: https://aclanthology.org/2021.findings-acl.84/

[29] Q. Zhu, L. Fan, and N. Weng, "Advancements in point cloud data augmentation for deep learning: A survey," *Pattern Recognit.*, vol. 153, Sep. 2024, Art. no. 110532.

[30] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[31] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," 2017, *arXiv:1712.04621*.

[32] X. Wang, K. Wang, and S. Lian, "A survey on face data augmentation for the training of deep neural networks," *Neural Comput. Appl.*, vol. 32, no. 19, pp. 15503–15531, Oct. 2020.

[33] C. Khosla and B. S. Saini, "Enhancing performance of deep learning models with different data augmentation techniques: A survey," in *Proc. Int. Conf. Intell. Eng. Manage. (ICIEM)*, Jun. 2020, pp. 79–85.

[34] S. Yang, W. Xiao, M. Zhang, S. Guo, J. Zhao, and F. Shen, "Image data augmentation for deep learning: A survey," 2022, *arXiv:2204.08610*.

[35] M. Xu, S. Yoon, A. Fuentes, and D. S. Park, "A comprehensive survey of image augmentation techniques for deep learning," *Pattern Recognit.*, vol. 137, May 2023, Art. no. 109347.

[36] A. Mumuni and F. Mumuni, "Data augmentation: A comprehensive survey of modern approaches," *Array*, vol. 4, May 2022, Art. no. 100258.

[37] X. Su, "A survey on data augmentation methods based on GAN in computer vision," in *Proc. Int. Conf. Natural Comput., Fuzzy Syst. Knowl. Discovery*, 2020, pp. 852–865.

[38] F. Yue, C. Zhang, M. Yuan, C. Xu, and Y. Song, "Survey of image augmentation based on generative adversarial network," *J. Phys. Conf. Ser.*, vol. 2203, no. 1, Feb. 2022, Art. no. 012052.

[39] F. Garcea, A. Serra, F. Lamberti, and L. Morra, "Data augmentation for medical imaging: A systematic literature review," *Comput. Biol. Med.*, vol. 152, Jan. 2023, Art. no. 106391.

[40] L. Deng, "The mnist database of handwritten digit images for machine learning research," *IEEE Signal Process. Mag.*, vol. 29, pp. 141–142, May 2012.

[41] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," 2017, *arXiv:1708.04552*.

[42] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 13001–13008.
[43] K. Kumar Singh, H. Yu, A. Sarmasi, G. Pradeep, and Y. Jae Lee, "Hide-and-seek: A data augmentation technique for weakly-supervised localization and beyond," 2018, *arXiv:1811.02545*.
[44] P. Chen, S. Liu, H. Zhao, X. Wang, and J. Jia, "GridMask data augmentation," 2020, *arXiv:2001.04086*.
[45] Y. Kim, A. F. M. S. Uddin, and S.-H. Bae, "Local augment: Utilizing local bias property of convolutional neural networks for data augmentation," *IEEE Access*, vol. 9, pp. 15191–15199, 2021.
[46] J.-W. Seo, H.-G. Jung, and S.-W. Lee, "Self-augmentation: Generalizing deep networks to unseen classes for few-shot learning," *Neural Netw.*, vol. 138, pp. 140–149, Jun. 2021.
[47] J. Choi, C. Lee, D. Lee, and H. Jung, "SalfMix: A novel single image-based data augmentation technique using a saliency map," *Sensors*, vol. 21, no. 24, p. 8444, Dec. 2021.
[48] C. Gong, D. Wang, M. Li, V. Chandra, and Q. Liu, "KeepAugment: A simple information-preserving data augmentation approach," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 1055–1064.
[49] J. Han, P. Fang, W. Li, J. Hong, M. Ali Armin, I. Reid, L. Petersson, and H. Li, "You only cut once: Boosting data augmentation with a single cut," 2022, *arXiv:2201.12078*.
[50] T. Xie, X. Cheng, X. Wang, M. Liu, J. Deng, T. Zhou, and M. Liu, "Cut-thumbnail: A novel data augmentation for convolutional neural network," in *Proc. 29th ACM Int. Conf. Multimedia*, 2021, pp. 1627–1635.
[51] J. Lemley, S. Bazrafkan, and P. Corcoran, "Smart augmentation learning an optimal data augmentation strategy," *IEEE Access*, vol. 5, pp. 5858–5869, 2017.
[52] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond empirical risk minimization," 2017, *arXiv:1710.09412*.
[53] A. F. M. Shahab Uddin, M. Sirazam Monira, W. Shin, T. Chung, and S.-H. Bae, "SaliencyMix: A saliency guided data augmentation strategy for better regularization," 2020, *arXiv:2006.01791*.
[54] T. Kumar, A. Mileo, R. Brennan, and M. Bendechache, "RSMDA: Random slices mixing data augmentation," *Appl. Sci.*, vol. 13, no. 3, p. 1711, Jan. 2023.
[55] J. H. Kim, W. Choo, and H. O. Song, "Puzzle mix: Exploiting saliency and local statistics for optimal mixup," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 5275–5285.
[56] S. Huang, X. Wang, and D. Tao, "Snapmix: Semantically proportional mixing for augmenting fine-grained data," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 1628–1636.
[57] E. Harris, A. Marcu, M. Painter, M. Niranjan, A. Prugel-Bennett, and J. Hare, "FMix: Enhancing mixed sample data augmentation," 2020, *arXiv:2002.12047*.
[58] A. Ramé, R. Sun, and M. Cord, "MixMo: Mixing multiple inputs for multiple outputs via deep subnetworks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 803–813.
[59] M. Hong, J. Choi, and G. Kim, "StyleMix: Separating content and style for enhanced data augmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 14857–14865.
[60] X. Liu, F. Shen, J. Zhao, and C. Nie, "RandoMix: A mixed sample data augmentation method with multiple mixed modes," 2022, *arXiv:2205.08728*.
[61] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffelaffel, "Mixmatch: A holistic approach to semi-supervised learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 1–23.
[62] D. Berthelot, N. Carlini, E. D. Cubuk, A. Kurakin, K. Sohn, H. Zhang, and C. Raffel, "ReMixMatch: Semi-supervised learning with distribution alignment and augmentation anchoring," 2019, *arXiv:1911.09785*.
[63] K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. A. Raffel, E. D. Cubuk, A. Kurakin, and C. L. Li, "Fixmatch: Simplifying semi-supervised learning with consistency and confidence," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 596–608.
[64] D. Hendrycks, N. Mu, E. D. Cubuk, B. Zoph, J. Gilmer, and B. Lakshminarayanan, "AugMix: A simple data processing method to improve robustness and uncertainty," 2019, *arXiv:1912.02781*.
[65] G. Ghiasi, Y. Cui, A. Srinivas, R. Qian, T.-Y. Lin, E. D. Cubuk, Q. V. Le, and B. Zoph, "Simple copy-paste is a strong data augmentation method for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 2917–2927.
[66] C. Summers and M. J. Dinneen, "Improved mixed-example data augmentation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2019, pp. 1262–1270.
[67] R. Takahashi, T. Matsubara, and K. Uehara, "Ricap: Random image cropping and patching data augmentation for deep CNNs," in *Proc. Asian Conf. Mach. Learn.*, 2018, pp. 786–798.
[68] J. Yoo, N. Ahn, and K.-A. Sohn, "Rethinking data augmentation for image super-resolution: A comprehensive analysis and a new strategy," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8372–8381.
[69] J. Qin, J. Fang, Q. Zhang, W. Liu, X. Wang, and X. Wang, "ResizeMix: Mixing data with preserved object information and true labels," 2020, *arXiv:2012.11101*.
[70] V. Olsson, W. Tranheden, J. Pinto, and L. Svensson, "ClassMix: Segmentation-based data augmentation for semi-supervised learning," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 1368–1377.
[71] Y. Su, R. Sun, G. Lin, and Q. Wu, "Context decoupling augmentation for weakly supervised semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 6984–6994.
[72] J. Zhang, Y. Zhang, and X. Xu, "ObjectAug: Object-level data augmentation for semantic image segmentation," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2021, pp. 1–8.
[73] E. D. Cubuk, B. Zoph, D. Mané, V. Vasudevan, and Q. V. Le, "AutoAugment: Learning augmentation strategies from data," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 113–123.
[74] R. Hataya, J. Zdenek, K. Yoshizoe, and H. Nakayama, "Faster AutoAugment: Learning augmentation strategies using backpropagation," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 1–16.
[75] M. Laskin, K. Lee, A. Stooke, L. Pinto, P. Abbeel, and A. Srinivas, "Reinforcement learning with augmented data," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 19884–19895.
[76] S. Lin, T. Yu, R. Feng, X. Li, X. Jin, and Z. Chen, "Local patch AutoAugment with multi-agent collaboration," 2021, *arXiv:2103.11099*.
[77] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le, "Randaugment: Practical automated data augmentation with a reduced search space," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 3008–3017.
[78] B. Zoph, E. D. Cubuk, G. Ghiasi, T. Y. Lin, J. Shlens, and Q. Le, "Learning data augmentation strategies for object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 566–583.
[79] Y. Chen, Y. Li, T. Kong, L. Qi, R. Chu, L. Li, and J. Jia, "Scale-aware automatic augmentation for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 9558–9567.
[80] S. Mehta, S. Naderiparizi, F. Faghri, M. Horton, L. Chen, A. Farhadi, O. Tuzel, and M. Rastegari, "RangeAugment: Efficient online augmentation with range learning," 2022, *arXiv:2212.10553*.
[81] S. Behpour, K. M. Kitani, and B. D. Ziebart, "ADA: Adversarial data augmentation for object detection," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2019, pp. 1243–1252.
[82] J. Guo and S. Gould, "Deep CNN ensemble with data augmentation for object detection," 2015, *arXiv:1506.07224*.
[83] X. Chen, C. Xie, M. Tan, L. Zhang, C.-J. Hsieh, and B. Gong, "Robust and accurate object detection via adversarial learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 16617–16626.
[84] K. Wang, B. Fang, J. Qian, S. Yang, X. Zhou, and J. Zhou, "Perspective transformation data augmentation for object detection," *IEEE Access*, vol. 8, pp. 4935–4943, 2020.
[85] X. Zhang, Z. Wang, D. Liu, Q. Lin, and Q. Ling, "Deep adversarial data augmentation for extremely low data regimes," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 1, pp. 15–28, Jan. 2021.
[86] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. (2007). *The PASCAL Visual Object Classes Challenge*. [Online]. Available: http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html
[87] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. (2012). *The PASCAL Visual Object Classes Challenge*. [Online]. Available: http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html
[88] T. Lin, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

[89] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

[90] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *Proc. NIPS Workshop Deep Learn. Unsupervised Feature Learn.*, 2011, p. 4.

[91] A. Krizhevsky and G. Hinton, *Learning Multiple Layers of Features From Tiny Images*. Princeton, NJ, USA: Citeseer, 2009.

[92] S. Lim, I. Kim, T. Kim, C. Kim, and S. Kim, "Fast auto augment," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 1–26.

[93] C. W. Kuo, C. Y. Ma, J. B. Huang, and Z. Kira, "Featmatch: Feature-based augmentation for semi-supervised learning," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 479–495.

[94] T. DeVries and G. W. Taylor, "Dataset augmentation in feature space," 2017, *arXiv:1702.05538*.

[95] P. Chu, X. Bian, S. Liu, and H. Ling, "Feature space augmentation for long-tailed data," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 694–710.

[96] R. Volpi, P. Morerio, S. Savarese, and V. Murino, "Adversarial feature augmentation for unsupervised domain adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5495–5504.

[97] S. C. Wong, A. Gatt, V. Stamatescu, and M. D. McDonnell, "Understanding data augmentation for classification: When to warp?" in *Proc. Int. Conf. Digit. Image Computing: Techn. Appl. (DICTA)*, Nov. 2016, pp. 1–6.

[98] X. Zheng, T. Chalasani, K. Ghosal, S. Lutz, and A. Smolic, "STaDA: Style transfer as data augmentation," 2019, *arXiv:1909.01056*.

[99] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 594–611, Apr. 2006.

[100] G. Griffin, A. Holub, and P. Perona, *Caltech-256 Object Category Dataset*. Berkeley, CA, USA: Univ. of California Press, 2007.

[101] P. T. G. Jackson, A. A. Abarghouei, S. Bonner, T. P. Breckon, and B. Obara, "Style augmentation: Data augmentation via style randomization," in *Proc. CVPR Workshops*, 2019, pp. 10–11.

[102] P. A. Cicalese, A. Mobiny, P. Yuan, J. Becker, C. Mohan, and H. V. Nguyen, "StyPath: Style-transfer data augmentation for robust histology image classification," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent*, 2020, pp. 351–361.

[103] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," 2015, *arXiv:1508.06576*.

[104] A. Toker, M. Eisenberger, D. Cremers, and L. Leal-Taixé, "SatSynth: Augmenting image-mask pairs through diffusion models for aerial semantic segmentation," 2024, *arXiv:2403.16605*.

[105] K. Islam, M. Zaigham Zaheer, A. Mahmood, and K. Nandakumar, "DiffuseMix: Label-preserving data augmentation with diffusion models," 2024, *arXiv:2405.14881*.

[106] Y. Tian, L. Fan, P. Isola, H. Chang, and D. Krishnan, "Stablerep: Synthetic images from text-to-image models make strong visual representation learners," in *Proc. Adv. Neural Inf. Process. Syst.*, 2024, pp. 1–16.

[107] B. Trabucco, K. Doherty, M. Gurinas, and R. Salakhutdinov, "Effective data augmentation with diffusion models," 2023, *arXiv:2302.07944*.

[108] Z. Wang, L. Wei, T. Wang, H. Chen, Y. Hao, X. Wang, X. He, and Q. Tian, "Enhance image classification via inter-class image mixup with diffusion model," 2024, *arXiv:2403.19600*.

[109] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.

[110] S. Azizi, S. Kornblith, C. Saharia, M. Norouzi, and D. J. Fleet, "Synthetic data from diffusion models improves ImageNet classification," 2023, *arXiv:2304.08466*.

[111] J. Liu, B. Liu, H. Zhou, H. Li, and Y. Liu, "Tokenmix: Rethinking image mixing for data augmentation in vision transformers," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 455–471.

[112] H. Xiao, W. Zheng, Z. Zhu, J. Zhou, and J. Lu, "Token-label alignment for vision transformers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, vol. 34, Oct. 2023, pp. 5472–5481.

[113] A. Radford, J. W. Kim, C. Hallacy, A. Goh, and S. Sastry, "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8748–8763.

[114] J. Yuan, Y. Liu, C. Shen, Z. Wang, and H. Li, "A simple baseline for semi-supervised semantic segmentation with strong data augmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 8209–8218.

[115] Z. Zhang, X. Zhang, C. Peng, X. Xue, and J. Sun, "Exfuse: Enhancing feature fusion for semantic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 269–284.

[116] B. Zoph, G. Ghiasi, T. Y. Lin, Y. Cui, H. Liu, E. D. Cubuk, and Q. Le, "Rethinking pre-training and self-training," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 3833–3845.

[117] P. Pawara, E. Okafor, L. Schomaker, and M. Wiering, "Data augmentation for plant classification," in *Proc. Int. Conf. Adv. Concepts Intell. Vis. Syst.*, 2017, pp. 615–626.

[118] T. Miyato, S.-I. Maeda, M. Koyama, and S. Ishii, "Virtual adversarial training: A regularization method for supervised and semi-supervised learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1979–1993, Aug. 2019.

[119] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 11–16.

[120] Y. Chen, X. Zhu, and S. Gong, "Semi-supervised deep learning with memory," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 268–283.

[121] S. Qiao, W. Shen, Z. Zhang, B. Wang, and A. Yuille, "Deep co-training for semi-supervised image recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 135–152.

[122] B. Athiwaratkun, M. Finzi, P. Izmailov, and A. Gordon Wilson, "There are many consistent explanations of unlabeled data: Why you should average," 2018, *arXiv:1806.05594*.

[123] V. Verma, K. Kawaguchi, A. Lamb, J. Kannala, A. Solin, Y. Bengio, and D. Lopez-Paz, "Interpolation consistency training for semi-supervised learning," 2019, *arXiv:1903.03825*.

[124] A. Iscen, G. Tolias, Y. Avrithis, and O. Chum, "Label propagation for deep semi-supervised learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5065–5074.

[125] Y. Luo, J. Zhu, M. Li, Y. Ren, and B. Zhang, "Smooth neighbors on teacher graphs for semi-supervised learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8896–8905.

[126] E. Arazo, D. Ortego, P. Albert, N. E. O'Connor, and K. McGuinness, "Pseudo-labeling and confirmation bias in deep semi-supervised learning," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2020, pp. 1–8.

[127] M. Sajjadi, M. Javanmardi, and T. Tasdizen, "Regularization with stochastic transformations and perturbations for deep semi-supervised learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 1–26.

[128] D. Lee, "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," in *Proc. Workshop Challenges Represent. Learn.*, 2013, p. 896.

[129] B. Kim, J. Choo, Y.-D. Kwon, S. Joe, S. Min, and Y. Gwon, "SelfMatch: Combining contrastive self-supervision and consistency for semi-supervised learning," 2021, *arXiv:2101.06480*.

[130] T. Kumar, J. Park, M. S. Ali, A. F. M. Uddin, and S. H. Bae, "Class specific autoencoders enhance sample diversity," *J. Broadcast Eng.*, vol. 26, pp. 844–854, Jun. 2021.

[131] S. Laine and T. Aila, "Temporal ensembling for semi-supervised learning," 2016, *arXiv:1610.02242*.

[132] J. Jackson and J. Schulman, "Semi-supervised learning by label gradient alignment," 2019, *arXiv:1902.02336*.

[133] A. Rasmus, M. Berglund, M. Honkala, H. Valpola, and T. Raiko, "Semi-supervised learning with ladder networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 1–22.

[134] S. Ren, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. In Neural Inf. Process. Syst.*, 2015, pp. 1–17.

[135] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "CenterNet: Keypoint triplets for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6568–6577.

[136] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2999–3007.

[137] N. Hernandez-Cruz, D. Cato, and J. Favela, "Neural style transfer as data augmentation for improving COVID-19 diagnosis classification," *Social Netw. Comput. Sci.*, vol. 2, no. 5, pp. 1–12, Sep. 2021.

[138] G. Ghiasi, T. Y. Lin, and Q. V. Le, "Dropblock: A regularization method for convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 1–26.
[139] Z. Zhang, T. He, H. Zhang, Z. Zhang, J. Xie, and M. Li, "Bag of freebies for training object detection neural networks," 2019, *arXiv:1902.04103*.
[140] H. Wang, Q. Wang, F. Yang, W. Zhang, and W. Zuo, "Data augmentation for object detection via progressive and selective instance-switching," 2019, *arXiv:1906.00358*.
[141] Y. Chen, P. Zhang, Z. Li, Y. Li, X. Zhang, L. Qi, J. Sun, and J. Jia, "Dynamic scale training for object detection," 2020, *arXiv:2004.12432*.
[142] H.-S. Fang, J. Sun, R. Wang, M. Gou, Y.-L. Li, and C. Lu, "InstaBoost: Boosting instance segmentation via probability map guided copy-pasting," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 682–691.
[143] B. Singh and L. S. Davis, "An analysis of scale invariance in object detection–SNIP," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3578–3587.
[144] B. Singh, M. Najibi, and L. S. Davis, "Sniper: Efficient multi-scale training," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 1–26.
[145] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.
[146] X. Wang, A. Shrivastava, and A. Gupta, "A-Fast-RCNN: Hard positive generation via adversary for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3039–3048.
[147] W. Liu, D. Anguelov, D. Erhan, and C. Szegedy, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 21–37.
[148] Y. Wu, A. Kirillov, F. Massa, W. Lo, and R. Girshick. (2019). *Detectron2*. [Online]. Available: https://github.com/facebookresearch/detectron2
[149] Y. Li, G. Hu, Y. Wang, T. Hospedales, N. M. Robertson, and Y. Yang, "DADA: Differentiable automatic data augmentation," 2020, *arXiv:2003.03780*.
[150] W.-C. Hung, Y.-H. Tsai, Y.-T. Liou, Y.-Y. Lin, and M.-H. Yang, "Adversarial learning for semi-supervised semantic segmentation," 2018, *arXiv:1802.07934*.
[151] S. Mittal, M. Tatarchenko, and T. Brox, "Semi-supervised semantic segmentation with High- and low-level consistency," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 4, pp. 1369–1379, Apr. 2021.
[152] G. French, T. Aila, S. Laine, M. Mackiewicz, and G. Finlayson. (2019). *Semi-Supervised Semantic Segmentation Needs Strong, High-Dimensional Perturbations*. [Online]. Available: https://openreview.net/forum?id=B1eBoJStwr
[153] Z. Feng, Q. Zhou, Q. Gu, X. Tan, G. Cheng, X. Lu, J. Shi, and L. Ma, "DMT: Dynamic mutual training for semi-supervised learning," 2020, *arXiv:2004.08514*.
[154] R. Mendel, L. A. De Souza, D. Rauber, J. P. Papa, and C. Palm, "Semi-supervised segmentation based on error-correcting supervision," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 141–157.
[155] N. Souly, C. Spampinato, and M. Shah, "Semi supervised semantic segmentation using generative adversarial network," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5689–5697.
[156] Y. Ouali, C. Hudelot, and M. Tami, "Semi-supervised semantic segmentation with cross-consistency training," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 12671–12681.
[157] Y. Zou, Z. Zhang, H. Zhang, C.-L. Li, X. Bian, J.-B. Huang, and T. Pfister, "PseudoSeg: Designing pseudo labels for semantic segmentation," 2020, *arXiv:2010.09713*.
[158] K. Weiss, T. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *J. Big Data*, vol. 3, pp. 1–40, May 2016.
[159] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, pp. 84–90, May 2017.
[160] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, May 2014.
[161] D. Erhan, A. Courville, Y. Bengio, and P. Vincent, "Why does unsupervised pre-training help deep learning?" in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 201–208.
[162] W. Khan, K. Raj, T. Kumar, A. M. Roy, and B. Luo, "Introducing Urdu digits dataset with demonstration of an efficient and robust noisy decoder-based pseudo example generator," *Symmetry*, vol. 14, no. 10, p. 1976, Sep. 2022.
[163] L. Shao, F. Zhu, and X. Li, "Transfer learning for visual categorization: A survey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 5, pp. 1019–1034, May 2015.
[164] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
[165] G. Bravo-Rocca, P. Liu, J. Guitart, A. Dholakia, D. Ellison, and M. Hodak, "Human-in-the-loop online multi-agent approach to increase trustworthiness in ML models through trust scores and data augmentation," 2022, *arXiv:2204.14255*.
[166] A. Halevy, P. Norvig, and F. Pereira, "The unreasonable effectiveness of data," *IEEE Intell. Syst.*, vol. 24, no. 2, pp. 8–12, Mar. 2009.
[167] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting unreasonable effectiveness of data in deep learning era," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 843–852.
[168] S. Xie, H. Lin, and Y. Liu, "Semi-supervised extractive speech summarization via co-training algorithm," in *Proc. Interspeech*, Sep. 2010, pp. 2522–2525.
[169] D. Pathak, P. Krähenbuhl, and T. Darrell, "Constrained convolutional neural networks for weakly supervised segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1796–1804.
[170] A. Kolesnikov and C. Lampert, "Seed, expand and constrain: Three principles for weakly-supervised image segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 695–711.
[171] Y. Wei, X. Liang, Y. Chen, X. Shen, M.-M. Cheng, J. Feng, Y. Zhao, and S. Yan, "STC: A simple to complex framework for weakly-supervised semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 11, pp. 2314–2320, Nov. 2017.
[172] Y. Wei, J. Feng, X. Liang, M.-M. Cheng, Y. Zhao, and S. Yan, "Object region mining with adversarial erasing: A simple classification to semantic segmentation approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6488–6496.
[173] A. Chaudhry, P. K. Dokania, and P. H. S. Torr, "Discovering class-specific pixels for weakly-supervised semantic segmentation," 2017, *arXiv:1707.05821*.
[174] Y. Wei, H. Xiao, H. Shi, Z. Jie, J. Feng, and T. S. Huang, "Revisiting dilated convolution: A simple approach for weakly- and semi-supervised semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7268–7277.
[175] X. Wang, S. You, X. Li, and H. Ma, "Weakly-supervised semantic segmentation by iteratively mining common object features," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1354–1362.
[176] Z. Huang, X. Wang, J. Wang, W. Liu, and J. Wang, "Weakly-supervised semantic segmentation network with deep seeded region growing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7014–7023.
[177] J. Ahn and S. Kwak, "Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4981–4990.
[178] J. Ahn, S. Cho, and S. Kwak, "Weakly supervised learning of instance segmentation with inter-pixel relations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2204–2213.
[179] J. Lee, E. Kim, S. Lee, J. Lee, and S. Yoon, "FickleNet: Weakly and semi-supervised semantic image segmentation using stochastic inference," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5262–5271.
[180] Y. Wang, J. Zhang, M. Kan, S. Shan, and X. Chen, "Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 12272–12281.
[181] J. Fan, Z. Zhang, C. Song, and T. Tan, "Learning integral objects with intra-class discriminator for weakly-supervised semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4282–4291.
[182] L. C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder–decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 801–818.
[183] E. Bisong and E. Bisong, "Regularization for deep learning," *Building Mach. Learn. Deep Learn. Models Google Cloud Platform*, vol. 2, pp. 415–421, Jun. 2019.
[184] C. Bishop, "Training with noise is equivalent to Tikhonov regularization," *Neural Comput.*, vol. 7, pp. 108–116, May 1995.

[185] C. Fefferman, S. Mitter, and H. Narayanan, "Testing the manifold hypothesis," *J. Amer. Math. Soc.*, vol. 29, no. 4, pp. 983–1049, Feb. 2016.
[186] H. Narayanan and S. Mitter, "Sample complexity of testing the manifold hypothesis," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 1–19.
[187] K. Lenc and A. Vedaldi, "Understanding image representations by measuring their equivariance and equivalence," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 991–999.
[188] M. Jaderberg, K. Simonyan, and A. Zisserman, "Spatial transformer networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 1–26.
[189] F. Bayat and S. Wei, "Information bottleneck problem revisited," in *Proc. 57th Annu. Allerton Conf. Commun., Control, Comput. (Allerton)*, Sep. 2019, pp. 40–47.
[190] R. Shwartz-Ziv and N. Tishby, "Opening the black box of deep neural networks via information," 2017, *arXiv:1703.00810*.

TEERATH KUMAR received the bachelor's degree (Hons.) in computer science from the National University of Computer and Emerging Science (NUCES), Islamabad, Pakistan, and the master's degree in computer science and engineering from Kyung Hee University, South Korea. He is currently pursuing the Ph.D. degree with Dublin City University, Ireland. His research interests include advanced data augmentation, deep learning for medical imaging, generative adversarial networks semi-supervised learning, and neuro-symbolic AI.

ROB BRENNAN (Senior Member, IEEE) graduated from Dublin City University and Queens University Belfast. He is a Lecturer/Assistant Professor with the School of Computer Science, University College Dublin, Ireland; and a Funded Investigator with the Science Foundation Ireland ADAPT Centre for AI–Driven Digital Content Technology. In ADAPT, he leads the Value and Risk Challenge in the Transparent Digital Governance Research Strand. His main research interests include data protection, data value, data quality, semantics, data governance, and AI governance.

ALESSANDRA MILEO received the Ph.D. degree in computer science from the University of Milan, Italy. She is an Associate Professor with the School of Computing, Dublin City University; a Principal Investigator with the INSIGHT Centre for Data Analytics; and a Funded Investigator with the I-Form Centre for Advanced Manufacturing. She secured over one Million euros in national, international, and industry-funded projects. She has published more than 90 papers in the area of the Internet of Things, knowledge graphs, stream reasoning, neuro-symbolic computing, and explainable AI.

MALIKA BENDECHACHE received the Ph.D. degree in computer science from University College Dublin, Ireland. She is a Lecturer/Assist Professor with the School of Computer Science, University of Galway, Ireland; and a Funded Investigator with the ADAPT Centre for AI–Driven Digital Content Technology. She designs novel big data analytics and machine learning techniques to enhance the capability and efficiency of complex systems. She also leverages complex systems to improve the effectiveness, privacy, and trustworthiness of analytics/machine learning techniques. Her research interests include big data analytics, machine learning, AI governance and data governance, security, and privacy.