Scientific Research Publishing

Journal of Computer and Communications, 2026, 14(7), 1-11
https://www.scirp.org/journal/jcc
ISSN Online: 2327-5227
ISSN Print: 2327-5219

Less Is More: When Data Augmentation Hurts Small CNN Generalization

Rui Huang

Independent Researcher, Santa Clara, USA
Email: ruihuang99@outlook.com

How to cite this paper: Huang, R. (2026) Less Is More: When Data Augmentation Hurts Small CNN Generalization. *Journal of Computer and Communications*, **14**, 1-11.
https://doi.org/10.4236/jcc.2026.147001

Received: June 9, 2026
Accepted: July 4, 2026
Published: July 7, 2026

Abstract

Data augmentation is widely adopted as a default regularization strategy for training deep neural networks, yet its effectiveness in low-data regimes with capacity-constrained models remains poorly characterized despite its practical importance. We present a systematic empirical study of one cumulative augmentation hierarchy for small convolutional neural networks trained on limited subsets of CIFAR-10 and FashionMNIST. The study contains 35 experimental conditions, each evaluated with ten fixed random seeds, for 350 total model fits. These conditions cover seven progressively aggressive augmentation levels for CIFAR-10 SmallCNN at three training-set sizes, CIFAR-10 TinyCNN at $n = 1000$, and FashionMNIST SmallCNN at $n = 1000$. Augmentation beyond a single horizontal flip consistently and substantially degrades test accuracy in this hierarchy. On CIFAR-10 with 500 training samples, maximum augmentation reduces accuracy by 10.9 percentage points below the no-augmentation baseline, and on 5000 samples the degradation reaches 16.6 points. Statistical analysis confirms that the high-augmentation degradations are significant after Bonferroni correction (paired t-test, $p < 0.001$, Cohen's $d > 4.0$) and appear across architectures and datasets. Our analysis reveals that aggressive augmentation causes a transition from overfitting to underfitting of the augmented training distribution. These findings challenge the common assumption that stacking augmentation transforms necessarily improves generalization and provide actionable guidance for practitioners working with limited labeled data and constrained model budgets in edge AI, medical imaging, and domain-specific applications.

Keywords

Data Augmentation, Small CNNs, Low-Data Learning, Generalization, CIFAR-10, FashionMNIST, Model Capacity, Underfitting

1. Introduction

Data augmentation has become an indispensable component of modern deep learning pipelines. Since the pioneering work of Krizhevsky *et al.* [1], who demonstrated that random crops and horizontal flips substantially improved ImageNet classification, the field has developed an extensive repertoire of augmentation transforms. These include geometric perturbations such as rotations and affine transformations [2], regional dropout methods including Cutout [3] and CutMix [4], sample-level mixing strategies like mixup [5], and fully automated augmentation policies discovered through search [6]-[8]. The prevailing wisdom suggests that more diverse augmentation leads to better generalization, and practitioners routinely stack multiple transforms without carefully evaluating their individual or combined contributions.

This assumption, however, rests primarily on experiments conducted with large datasets and high-capacity models. ImageNet contains over one million training images, and modern architectures such as ResNet [9] exceed tens of millions of parameters. These models have sufficient representational capacity to extract useful features from heavily augmented training distributions without losing sight of the underlying visual concepts. In contrast, many practical applications operate under constraints that differ fundamentally from these benchmark conditions. Medical imaging studies may have only hundreds of labeled examples due to annotation costs and patient privacy. Edge deployment scenarios favor compact models with limited representational capacity to meet latency and memory requirements [10] [11]. Domain-specific tasks in industrial inspection, agricultural monitoring, satellite imagery analysis, educational open response grading [12], or rare event detection frequently lack the data volumes that standard augmentation strategies were designed and validated for.

The interaction between augmentation design, dataset size, and model capacity creates a tension that existing literature has not systematically addressed. RandAugment [7] demonstrated that the optimal number of augmentation transforms N depends on model and dataset size, showing that larger models benefit from higher N. However, this finding was established using WideResNet and ResNet architectures on full CIFAR-10 and ImageNet, leaving the low-data regime unexplored. Cai *et al.* [13] showed that standard augmentation can hurt tiny models and proposed network augmentation as an architectural alternative, but did not characterize the data augmentation failure mode itself or identify the transition point from beneficial to harmful augmentation. Theoretical work by Rajput *et al.* [14] established that augmentation introduces a bias-variance tradeoff, where aggressive augmentation can increase bias by shifting the effective training distribution away from the true data distribution. Chen *et al.* [15] further analyzed this phenomenon through similarity and diversity metrics. These theoretical insights predict that a crossover point should exist where augmentation-induced bias overwhelms variance reduction, but empirical validation in the low-data regime with small models remains sparse.

We address this gap with a controlled empirical study that measures augmentation scaling behavior for small CNNs trained on limited data. Our experimental design treats augmentation as an ordinal cumulative pipeline with seven levels, from no augmentation through progressively aggressive transform stacks. The primary CIFAR-10 SmallCNN study evaluates all seven levels at three dataset sizes (500, 1000, and 5000 training samples). Two focused ablations then evaluate all seven levels for TinyCNN on CIFAR-10 at $n = 1000$ and for SmallCNN on FashionMNIST at $n = 1000$. Thus the study contains 35 conditions, not a full factorial crossing of all datasets, sizes, and architectures. Every condition is evaluated with ten random seeds, yielding 350 total model fits with rigorous statistical analysis including paired t-tests with Bonferroni correction and Cohen's d effect sizes.

The contributions of this paper are as follows:

• We demonstrate the existence of an augmentation saturation point for small CNNs in low-data regimes, beyond which additional transforms actively degrade test accuracy. This saturation occurs at a single horizontal flip for all tested configurations on CIFAR-10, and at zero augmentation for FashionMNIST.

• We provide quantitative evidence that aggressive augmentation in low-data settings produces accuracy substantially below the no-augmentation baseline, with degradations ranging from 10.4 to 16.6 percentage points, accompanied by a transition from overfitting to underfitting of the augmented training distribution.

• We offer practical guidelines for augmentation selection under data constraints: a single geometric transform suffices for horizontally symmetric datasets, and stacking additional transforms is counterproductive.

2. Related Work

The development of data augmentation techniques for deep learning has progressed rapidly over the past decade, driven by the recognition that augmentation acts as an implicit regularizer that expands the effective training set size. Early approaches relied on label-preserving geometric transforms such as horizontal flips, random crops, and rotations, which Krizhevsky *et al.* [1] demonstrated were essential for training AlexNet on ImageNet. Subsequent work expanded the augmentation toolkit substantially. Shorten and Khoshgoftaar [2] catalogued dozens of transforms spanning geometric, photometric, and kernel-based categories, establishing a taxonomy that remains widely referenced. More recently, Yang *et al.* [16] and Maharana *et al.* [17] extended these surveys to cover modern approaches including neural style transfer and generative augmentation. Unlike these comprehensive surveys, our work does not propose new augmentation techniques but instead studies the scaling behavior of existing ones when applied cumulatively in data-scarce settings.

Regional dropout and sample mixing methods represent a distinct augmentation paradigm. DeVries and Taylor [3] proposed Cutout, which randomly masks contiguous square regions of input images, forcing the network to attend to distributed features. Zhong *et al.* [18] introduced Random Erasing with similar motivation but randomized aspect ratios. On the mixing side, Zhang *et al.* [5] pro-

posed mixup, which creates virtual training examples by linearly interpolating pairs of images and their labels, and Yun *et al.* [4] combined cutout with mixup in CutMix. While each of these methods has demonstrated benefits independently on full datasets with large models, their behavior when stacked together in low-data regimes has not been systematically evaluated.

Automated augmentation policy search represents a parallel and influential line of research. AutoAugment [6] used reinforcement learning to discover dataset-specific augmentation policies, achieving state-of-the-art results on CIFAR-10 and ImageNet at the cost of thousands of GPU hours. RandAugment [7] simplified this search to two hyperparameters: the number of transforms N applied per image and their shared magnitude M, demonstrating that optimal N increases with model capacity and dataset size. TrivialAugment [8] further reduced the search space by randomly selecting a single transform with random magnitude per image, matching AutoAugment performance without any search. While RandAugment's finding that optimal N varies with scale is directly related to our investigation, these automated methods were validated exclusively on full datasets with large models. Our work complements these findings by characterizing the augmentation scaling behavior in the low-data regime where practitioners most need guidance.

The effectiveness of augmentation under data constraints has received comparatively less systematic attention. Perez and Wang [19] studied augmentation effectiveness across CIFAR-10 subsets and found that augmentation provides larger relative gains at smaller dataset sizes, but did not investigate stacking behavior or identify a saturation point. Chen *et al.* [15] analyzed augmentation effectiveness through the lens of similarity and diversity between augmented and original distributions. Kumar *et al.* [20] proposed an adaptive augmentation framework for few-shot learning that adjusts augmentation to the domain. While these studies suggest that augmentation effectiveness depends on data availability, none provide the systematic measurement of the augmentation-versus-accuracy curve across data sizes that our work delivers.

Research on small and efficient models has highlighted capacity-related limitations of standard training practices. Cai *et al.* [13] showed that data augmentation degrades accuracy of tiny deep learning models and proposed network augmentation as an alternative. Srivastava *et al.* [21] showed that dropout prevents overfitting by randomly zeroing activations, and Rajput *et al.* [14] provided theoretical analysis showing that augmentation can both increase margin and introduce bias. Our work bridges these perspectives by empirically demonstrating that the bias introduced by aggressive augmentation dominates any regularization benefit when training data is scarce and model capacity is limited.

3. Method

3.1. Problem Formulation

We formulate the relationship between augmentation level and generalization performance as a controlled experiment with the cumulative pipeline level as the

primary independent variable. Let $\mathcal{A}_k$ denote the k th pipeline in our fixed hierarchy, where $\mathcal{A}_0$ represents tensor conversion plus normalization and later levels add or intensify specific transforms in a predetermined order. Given a training set $\mathcal{D}_{\text{train}}$ of size n, a model f_θ with parameter count $|\theta|$, and a fixed training protocol $\mathcal{T}$, we measure the test accuracy $\text{Acc}\left(f_\theta, \mathcal{D}_{\text{train}}, \mathcal{A}_k, \mathcal{T}\right)$ as a function of the pipeline level k for varying n and f_θ.

Our central hypothesis posits that $\text{Acc}(k)$ exhibits a peak at a small value of k that depends on n and $|\theta|$. Beyond this *augmentation saturation point* $k^*\left(n,|\theta|\right)$, the chosen cumulative pipeline may introduce distributional shift between the augmented training data and the test distribution, degrading generalization. The design therefore tests one practically motivated pipeline ordering rather than the isolated causal effect of transform count.

3.2. Augmentation Hierarchy

We define seven augmentation levels ($k = 0,1,\cdots,6$) with a fixed ordering from simple geometric transforms to aggressive combinations. Levels 1 - 5 add one transform family at a time, while Level 6 both adds affine translation and increases selected magnitudes. Consequently, "level" should be interpreted as membership in this cumulative hierarchy, not as transform count in isolation. The hierarchy is:

- **Level 0** (No Augmentation): tensor conversion and normalization only.
- **Level 1** (Flip): adds random horizontal flip ($p = 0.5$).
- **Level 2** (+Crop): adds random crop with 4-pixel padding.
- **Level 3** (+Jitter): adds color jitter (brightness/contrast/saturation: 0.2, hue: 0.1).
- **Level 4** (+Rotation): adds random rotation up to 15.
- **Level 5** (+Erasing): adds random erasing ($p = 0.5$, scale: [0.02, 0.2]).
- **Level 6** (Maximum): increases color jitter and rotation magnitudes, increases erasing scale, and adds affine translation.

3.3. Experimental Design

We evaluate the hierarchy in three targeted blocks rather than a full factorial design. The primary block uses CIFAR-10 and SmallCNN across $n \in \{500,1000,5000\}$ for all seven augmentation levels, producing 21 conditions. The capacity block uses CIFAR-10, $n = 1000$, and TinyCNN for all seven levels. The cross-dataset block uses FashionMNIST, $n = 1000$, and SmallCNN for all seven levels. These blocks produce $21 + 7 + 7 = 35$ conditions. SmallCNN uses three convolutional layers (32, 64, 64 filters) with batch normalization and max pooling, followed by a 256-unit classifier with 30% dropout. TinyCNN uses two convolutional layers (16, 32 filters) with a 128-unit classifier. The measured trainable parameter counts are 321,610 for SmallCNN and 268,746 for TinyCNN.

3.4. Training and Evaluation Protocol

All models are trained with SGD (learning rate 0.01, momentum 0.9, weight

decay 5×10^{-4}), batch size 64, and cosine annealing. Models with $n \leq 1000$ train for 15 epochs; $n = 5000$ for 8 epochs. Each condition uses the same ten seeds: 42, 123, 456, 789, 1024, 2048, 3072, 4096, 5120, and 6144. For each dataset, training size, and seed, we draw a stratified training subset with equal class allocation using NumPy's seeded random generator; the resulting subset is reused across augmentation levels and model architectures for that seed and size. CIFAR-10 uses a pre-cached tensor version of the standard training and test splits. FashionMNIST uses the standard torchvision training and test splits and applies the same stratified training-subset procedure. For evaluation, both datasets use a fixed stratified 1000-image test subset, selected once with random seed 0 as 100 examples per class and reused across all augmentation levels, training seeds, and models. We report mean ± std with 95% confidence intervals over the ten model fits per condition. Statistical comparisons use paired *t*-tests with Bonferroni correction across 30 planned comparisons ($\alpha_{\text{adj}} = 0.0017$) and Cohen's d effect sizes.

4. Experiments

4.1. Implementation Details

All experiments used PyTorch 2.11.0 on a 96-core x86_64 CPU server (235 GB RAM, no GPU). CIFAR-10 was loaded from a pre-cached tensor file; FashionMNIST via torchvision. Experiments were parallelized with 16 concurrent processes (2 OpenMP threads each), completing 350 model fits in approximately 90 minutes.

4.2. Primary Study: Augmentation Scaling on CIFAR-10

Table 1 summarizes the main accuracy results. At every CIFAR-10 dataset size, Level 1 (horizontal flip) achieves the highest observed test accuracy, and performance declines monotonically from Level 2 onward. With $n = 500$, the baseline achieves 43.9% ± 1.3% and flip-only achieves 45.0% ± 1.1% ($p = 0.00095$, $d = -1.52$), which is significant under $\alpha_{\text{adj}} = 0.0017$. Level 6 drops to 33.0% ± 2.3%, a 10.9-point degradation. With $n = 1000$, the baseline is 50.0% ± 1.6%, Level 1 is 50.8% ± 1.3% ($p = 0.008$), a nominal improvement that is not significant after Bonferroni correction. Level 6 is 38.5% ± 1.6% (−11.5 points). With $n = 5000$, the baseline is 61.6% ± 1.4%, Level 1 is 62.0% ± 0.9% ($p = 0.495$), and Level 6 is 45.0% ± 1.1% (−16.6 points). All CIFAR-10 SmallCNN Level 0 vs Level 6 comparisons are significant after correction ($p < 10^{-6}$, $d > 4.0$).

4.3. Model Capacity Ablation

TinyCNN (269 K params) on CIFAR-10 with $n = 1000$ shows the same pattern: baseline 46.4% ± 1.4%, peak at Level 1 (47.9% ± 1.3%, $p = 0.0011$), and decline to 35.9% ± 1.7% at Level 6. The Level 1 gain remains significant under $\alpha_{\text{adj}} = 0.0017$, and the observed saturation point does not shift between architectures.

Table 1. Test accuracy (%) across augmentation levels and dataset sizes on CIFAR-10 (SmallCNN, 10 seeds). Bold indicates best per row. Daggers (†) indicate statistically significant difference from baseline under $\alpha_{adj} = 0.0017$.

Data Size	No Aug	Flip	+Crop	+Jitter	+Rot	+Erase	+All
$n = 500$	43.9 ± 1.3	**45.0 ± 1.1**†	40.8 ± 2.2 †	40.6 ± 1.4†	38.3 ± 1.9†	37.4 ± 2.4†	33.0 ± 2.3†
$n = 1000$	50.0 ± 1.6	**50.8 ± 1.3**	46.0 ± 0.9†	45.9 ± 1.5†	43.4 ± 1.8†	43.0 ± 1.8†	38.5 ± 1.6†
$n = 5000$	61.6 ± 1.4	**62.0 ± 0.9**	56.4 ± 1.0†	55.9 ± 1.4†	52.3 ± 1.4†	50.7 ± 1.3†	45.0 ± 1.1†
Model capacity ablation (*CIFAR*-10, $n = 1000$):							
TinyCNN	46.4 ± 1.4	**47.9 ± 1.3**†	43.9 ± 1.7†	43.1 ± 1.7†	40.9 ± 1.4†	40.1 ± 1.8†	35.9 ± 1.7†
Cross-dataset validation ($n = 1000$):							
FashionMNIST	**81.5 ± 0.8**	80.9 ± 0.5	75.3 ± 0.8†	76.0 ± 1.2†	74.8 ± 1.1†	73.4 ± 0.9†	71.1 ± 1.1†

4.4. Cross-Dataset Validation

On FashionMNIST ($n = 1000$), SmallCNN peaks at Level 0 (81.5% ± 0.8%). Flip provides no reliable benefit (80.9% ± 0.5%, $p = 0.034$), which is non-significant under $\alpha_{adj} = 0.0017$. Level 6 reaches 71.1% ± 1.1% (−10.4 points, $d = 8.87$), a significant degradation. The absence of flip benefit is consistent with the weaker horizontal-invariance assumption for many clothing items.

5. Results

Figure 1 presents the augmentation scaling curves for SmallCNN across three CIFAR-10 subset sizes. All curves exhibit a slight increase from Level 0 to Level 1, followed by a steep decline through Level 6. The decline rate is approximately 1.5 to 3.0 percentage points per higher pipeline level. Confidence intervals widen at higher augmentation levels, indicating that aggressive augmentation increases run-to-run variability.

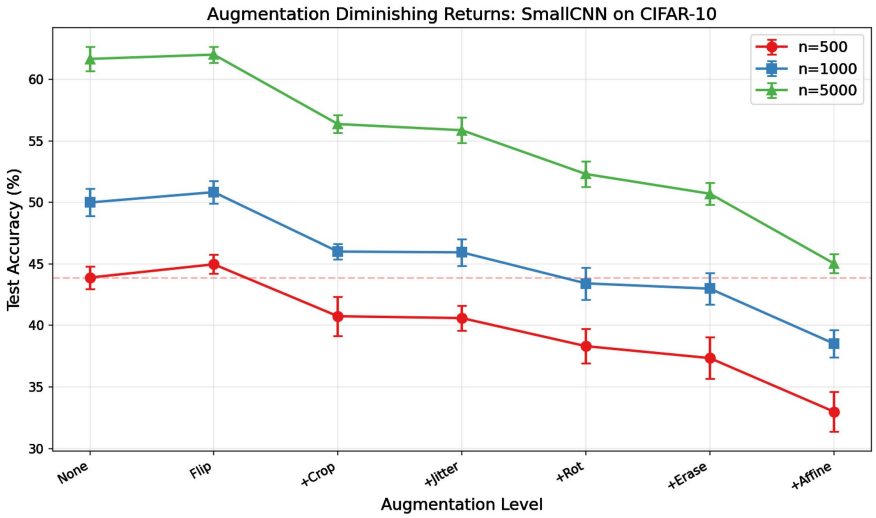

Figure 1. Test accuracy vs. augmentation level for SmallCNN on CIFAR-10 across three dataset sizes. Error bars show 95% confidence intervals. All curves peak at Level 1 (horizontal flip) and decline monotonically thereafter.

Figure 2 confirms that the two architectures produce parallel scaling curves with a consistent 3 - 4 point offset. The train-test accuracy gap reveals a transition from overfitting at Level 0 to near-zero or negative gaps at Level 6, indicating that aggressive augmentation transforms the training task from memorizable to difficult to fit.

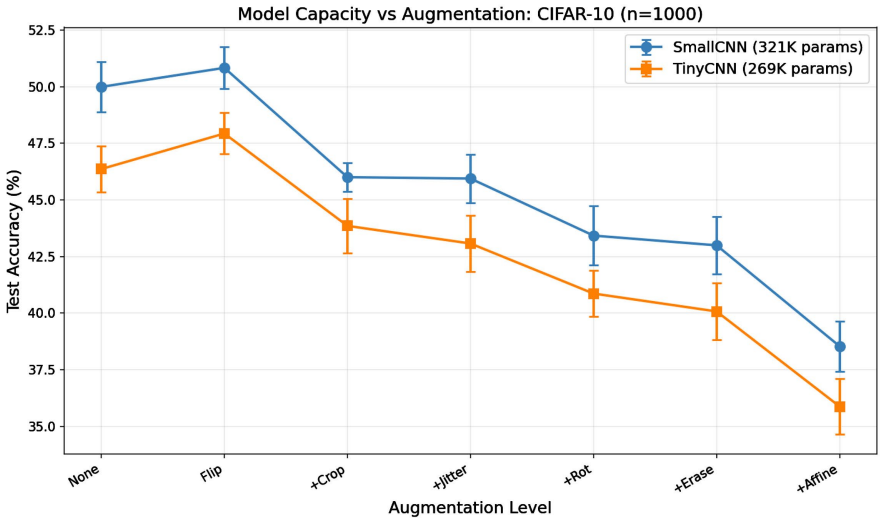

Figure 2. Model capacity comparison: SmallCNN vs. TinyCNN on CIFAR-10 ($n=1000$). The curves are nearly parallel, showing that model capacity affects absolute performance but not augmentation scaling behavior.

6. Discussion

The mechanism underlying the degradation can be understood through the distributional mismatch framework [14]. When a small training set is augmented aggressively, the augmented distribution can diverge from the test distribution. A model trained on this shifted distribution may learn augmentation-specific features rather than semantically meaningful ones. Table 2 supports this interpretation across every evaluated block: Level 0 shows high training accuracy and a positive train-test gap, whereas Level 6 reduces training accuracy to 31.0% - 68.8% and drives the gap to near-zero or negative values. For CIFAR-10 SmallCNN at $n=500$, training accuracy drops from 93.0% at Level 0 to 32.9% at Level 6, indicating that the model cannot fit the aggressive augmented distribution.

Table 2. Training accuracy and train-test gap (%) at key augmentation levels. Each cell reports train accuracy/train-test gap.

Setting	Level 0	Level 1	Level 6
CIFAR-10 SmallCNN, $n=500$	93.0/49.1	76.4/31.5	32.9/−0.1
CIFAR-10 SmallCNN, $n=1000$	93.5/43.5	77.4/26.6	35.3/−3.3
CIFAR-10 SmallCNN, $n=5000$	76.3/14.7	69.8/7.8	38.9/−6.1
CIFAR-10 TinyCNN, $n=1000$	82.7/36.3	68.1/20.2	31.0/−4.9
FashionMNIST SmallCNN, $n=1000$	96.8/15.4	92.9/12.0	68.8/−2.3

Horizontal flip is the only consistently non-harmful augmentation on CIFAR-10 in this hierarchy and is significantly beneficial in the $n = 500$ SmallCNN and $n = 1000$ TinyCNN settings. It preserves structure and texture while increasing exposure to plausible views of horizontally symmetric content. Random crop with padding appears to introduce sufficient spatial distortion at 32 × 32 resolution to degrade performance when anchoring examples are scarce. The FashionMNIST results confirm that augmentation value depends on domain invariances: flip does not provide a statistically reliable gain for clothing items.

These results complement RandAugment's finding that augmentation policy depends on scale by showing that, for this cumulative low-data pipeline, the best observed CIFAR-10 setting occurs at Level 1 and FashionMNIST at Level 0. Practitioners in edge AI, medical imaging, and industrial applications should validate simple, domain-appropriate transforms before adopting standard multi-transform recipes.

7. Limitations

All experiments were conducted on CPU without GPU acceleration, limiting epoch counts (15 for $n \le 1000$, 8 for $n = 5000$). Longer training might shift the saturation point. Test evaluation used a fixed stratified 1000-image subsample with 100 examples per class, introducing additional variance relative to full-test-set evaluation. The study covers two datasets and two architectures with similar parameter counts (269 K and 321 K), and only a targeted 35-condition design was run rather than the full dataset-size-by-architecture-by-dataset factorial. Wider diversity in model scales, image resolutions, data domains, and independently permuted augmentation orders would strengthen generalizability. No learning rate sensitivity analysis was performed.

8. Conclusion

We demonstrated that data augmentation exhibits strong diminishing returns for small CNNs in low-data regimes under one cumulative augmentation hierarchy. Across 350 model fits, augmentation beyond horizontal flip consistently degrades performance by 10.4 - 16.6 percentage points. The mechanism is consistent with a transition from beneficial regularization to harmful distributional shift and underfitting of the augmented training distribution. Practitioners training small models with fewer than 5000 examples should start with at most one domain-appropriate geometric transform and validate multi-transform pipelines rather than assuming that policies validated on large-scale data will transfer.

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

References

[1] Krizhevsky, A., Sutskever, I. and Hinton, G.E. (2012) ImageNet Classification with

Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems*, **25**, 1097-1105.

[2] Shorten, C. and Khoshgoftaar, T.M. (2019) A Survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, **6**, 1-48. https://doi.org/10.1186/s40537-019-0197-0

[3] DeVries, T. and Taylor, G.W. (2017) Improved Regularization of Convolutional Neural Networks with Cutout. arXiv:1708.04552.

[4] Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J. and Yoo, Y. (2019) CutMix: Regularization Strategy to Train Strong Classifiers with Localizable Features. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Seoul, 27 October 2019-2 November 2019, 6023-6032.

[5] Zhang, H., Cisse, M., Dauphin, Y.N. and Lopez-Paz, D. (2018) Mixup: Beyond Empirical Risk Minimization. *Proceedings of the* 6*th International Conference on Learning Representations* (*ICLR* 2018), Vancouver, 30 April-3 May 2018.

[6] Cubuk, E.D., Zoph, B., Mané, D., Vasudevan, V. and Le, Q.V. (2019) Autoaugment: Learning Augmentation Strategies from Data. 2019 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), Long Beach, 15-20 June 2019, 113-123. https://doi.org/10.1109/cvpr.2019.00020

[7] Cubuk, E.D., Zoph, B., Shlens, J. and Le, Q.V. (2020) Randaugment: Practical Automated Data Augmentation with a Reduced Search Space. 2020 *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops* (*CVPRW*), Seattle, 14-19 June 2020, 702-703. https://doi.org/10.1109/cvprw50498.2020.00359

[8] Muller, S.G. and Hutter, F. (2021) TrivialAugment: Tuning-Free Yet State-of-the-Art Data Augmentation. 2021 *IEEE/CVF International Conference on Computer Vision* (*ICCV*), Montreal, 10-17 October 2021, 774-782. https://doi.org/10.1109/iccv48922.2021.00081

[9] He, K., Zhang, X., Ren, S. and Sun, J. (2016) Deep Residual Learning for Image Recognition. 2016 *IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), Las Vegas, 27-30 June 2016, 770-778. https://doi.org/10.1109/cvpr.2016.90

[10] Huang, R. (2026) Edge-Centric Generative AI: A Survey on Efficient Inference for Large Language Models in Resource-Constrained Environments. *Journal of Computer and Communications*, **14**, 238-253. https://doi.org/10.4236/jcc.2026.144012

[11] Huang, R. (2026) ConfSched: Confidence-Threshold Routing for Efficient Edge-Cloud Activity Recognition. *World Journal of Engineering and Technology*, **14**, 471-486. https://doi.org/10.4236/wjet.2026.142027

[12] Erickson, J.A., Botelho, A.F., Peng, Z., Huang, R., Kasal, M.V. and Heffernan, N.T. (2021) Is It Fair? Automated Open Response Grading. *Proceedings of the* 14*th International Conference on Educational Data Mining*, 29 June-2 July 2021, 682-687.

[13] Cai, H., Gan, C., Lin, J. and Han, S. (2022) Network Augmentation for Tiny Deep Learning. *Proceedings of the* 10*th International Conference on Learning Representations* (*ICLR* 2022), Virtual, 25-29 April 2022.

[14] Rajput, S., Feng, Z., Charles, Z., Loh, P. and Papailiopoulos, D. (2019) Does Data Augmentation Lead to Positive Margin? *Proceedings of the* 36*th International Conference on Machine Learning*, Long Beach, 9-15 June 2019, 5321-5330.

[15] Yang, S., Guo, S., Zhao, J. and Shen, F. (2024) Investigating the Effectiveness of Data Augmentation from Similarity and Diversity: An Empirical Study. *Pattern Recognition*, **148**, Article 110204. https://doi.org/10.1016/j.patcog.2023.110204

[16] Alomar, K., Aysel, H.I. and Cai, X. (2023) Data Augmentation in Classification and

Segmentation: A Survey and New Strategies. *Journal of Imaging*, **9**, Article 46. https://doi.org/10.3390/jimaging9020046

[17] Maharana, K., Mondal, S. and Nemade, B. (2022) A Review: Data Pre-Processing and Data Augmentation Techniques. *Global Transitions Proceedings*, **3**, 91-99. https://doi.org/10.1016/j.gltp.2022.04.020

[18] Zhong, Z., Zheng, L., Kang, G., Li, S. and Yang, Y. (2020) Random Erasing Data Augmentation. *Proceedings of the AAAI Conference on Artificial Intelligence*, **34**, 13001-13008. https://doi.org/10.1609/aaai.v34i07.7000

[19] Perez, L. and Wang, J. (2017) The Effectiveness of Data Augmentation in Image Classification Using Deep Learning. arXiv:1712.04621.

[20] Pintelas, E., Livieris, I.E. and Pintelas, P. (2024) Adaptive Augmentation Framework for Domain Independent Few Shot Learning. *Knowledge-Based Systems*, **299**, Article 112047. https://doi.org/10.1016/j.knosys.2024.112047

[21] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R. (2014) Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, **15**, 1929-1958.