# AGVBench: A Reliability-Oriented Benchmark of Data Augmentation for Vein Recognition

Haiyang Li⋆, Yuming Fu⋆, Qun Song⋆, Hongchao Liao, Jing Chen,
*Senior, IEEE,* Mounim A. El-Yacoubi, Xin Jin†

***Abstract*—Vein recognition is a secure biometric technology, but its progress is often constrained by limited annotated data and domain-specific imaging variations. Data augmentation is widely used to mitigate this issue, yet strategies designed for natural images may disrupt the fine-grained topology and high-frequency textures that are essential for vein identity discrimination. To provide a systematic evaluation, we present AGVBench, a comprehensive benchmark and open-source codebase for studying data augmentation in vein recognition. AGVBench evaluates 30 representative augmentation strategies on five public palm- and finger-vein datasets with seven backbone architectures, covering classic CNNs, vision transformers, and vein-specific recognition models. Our results show that multi-image mixing methods, such as MixUp, PuzzleMix, and StarMixup, generally provide the strongest recognition and verification performance. However, they are often poorly calibrated and vulnerable to adversarial perturbations, revealing a clear accuracy-robustness trade-off. We also find that geometric transformations frequently degrade vein recognition by disturbing vascular topology, and that augmentation effectiveness varies across palm and finger vein datasets. These findings show that accuracy-centric evaluation is insufficient for biometric augmentation. AGVBench provides a standardized protocol and reusable implementation to support reproducible research and guide the design of more reliable, robust, and deployment-oriented vein recognition systems.**

***Index Terms*—Data Augmentation, Vein Identification, Biometrics, Computer Vision.**

## I. Introduction

In the digital era, biometric authentication, particularly vein recognition [1]–[4], has become a cornerstone of security due to its unique internal, spoof-resistant, and privacy-preserving vascular patterns. While deep learning architectures have significantly enhanced discriminative performance, their efficacy remains constrained by a persistent "small-sample" dilemma inherent in biometric acquisition. To mitigate this data scarcity, researchers extensively employ data augmentation [5]–[9] (DA). However, most studies inherit strategies optimized for natural images, often overlooking the fundamental morphological differences between semantic objects and vascular structures. This empirical transfer can be counterproductive, as aggressive transformations risk obliterating the delicate topological and high-frequency details essential for identity discrimination. These issues underscore a critical need for a specialized evaluation framework.

***Why do we call for a vein-specific augmentation benchmark?*** While advanced augmentation strategies [10]–[12] have shown immense potential in natural image tasks, their direct application to vascular biometrics remains poorly understood due to the delicate and high-frequency nature of vein patterns. Without a systematic evaluation, it is uncertain whether complex augmentations, such as policy-based [5], [6], [13], [14] or mixup-based [8], [15] methods, consistently serve as superior alternatives to traditional ones across diverse neural architectures like Convolutional Neural Networks [16] (CNNs) and Vision Transformers [17] (ViTs). Moreover, a thorough and standardized assessment of how these techniques affect biometric fidelity is conspicuously missing in the community. A benchmark is precisely the mechanism to establish such an understanding, playing a pivotal role in driving research progress by integrating an agreed-upon set of tasks, impartial comparisons, and rigorous assessment criteria. However, there has been a lack of a comprehensive benchmark for vein recognition to facilitate unbiased comparisons and practical deployment.

***Why do we need an open-source vein augmentation codebase?*** Notably, most existing data augmentation techniques used in vein recognition are implemented with varied settings, hyperparameters, and coding styles across different research papers. This lack of standardization not only hinders user-friendly reproduction and deployment but also imposes costly trial-and-error on practitioners to determine the most effective augmentation strategy for their specific biometric needs in real-world applications. Hence, it is essential to develop a unified vein representation learning codebase for standardized data pre-processing, diverse augmentation module selection, network architecture integration, model training, and empirical analysis. Such a platform would bridge the gap between theoretical research and practical implementation, fostering further innovation in robust vein recognition systems.

In this paper, we introduce AGVBench, the first comprehensive benchmark specifically curated to rethink and evaluate data augmentation strategies for vein recognition. Our work addresses the existing knowledge gap by providing a rigorous experimental framework and actionable insights. The primary contributions of this research can be summarized as:

- We construct AGVBench, the first large-scale benchmark for data augmentation in vein recognition, systematically

H. Li, Y. Fu, J. Chen and Q. Song are with Chongqing Technology and Business University, Chongqing 400067, China (E-mail: xxx).

X. Jin is with the School of Engineering, Westlake University, Hangzhou, Zhejiang Province, China (E-mail: jinxin86@westlake.edu.cn).

M. A. El-Yacoubi is with SAMOVAR, Telecom SudParis, Institute Polytechnique de Paris, 91120 Palaiseau, France (E-mail: mounim.el_yacoubi@telecom-sudparis.eu).

H. Liao is with the Guangzhou College of Applied Science and Technology, Guangzhou, Guangdong Province, China (E-mail: liaohongchao@gzasc.edu.cn).

⋆: Equal contribution. †: Corresponding author.

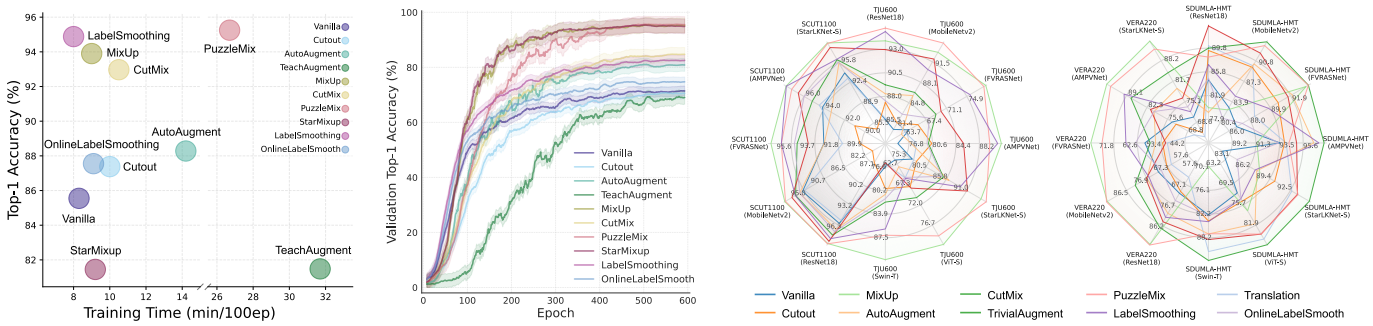


Fig. 1. Comprehensive performance evaluation of various data augmentation and regularization methods. (a) Trade-off of Accuracy *vs.* Training Time. (b) The fitting curve of accuracy and training epochs. (c) Generalization evaluation across various vein datasets and network architectures via radar charts.

evaluating 30 methods on five datasets and seven architectures across six dimensions: recognition and verification performance, calibration, corruption robustness, adversarial robustness, occlusion robustness, and computational efficiency.

- We reveal a critical accuracy-robustness trade-off: top-performing mixup methods are simultaneously the least well calibrated and most adversarially fragile, demonstrating that accuracy-centric evaluation is insufficient for assessing biometric augmentation.
- We establish a standardized experimental protocol and release our complete library to the community. This serves as a foundational resource for facilitating reproducible research and guiding the design of more robust, vein-specific augmentation pipelines.

## II. Background and Related Works

### A. Problem Formulation

The core objective of vein recognition is to learn a discriminative mapping function that projects raw vascular patterns into a high-dimensional feature space for identity verification. Let $\mathcal{X} = \{x_i \mid x_i \in \mathbb{R}^{H\times W\times C}\}_{i=1}^{N}$ denote the input space of vein images, where $H$, $W$, and $C$ represent the height, width, and the number of channels, respectively. Although raw near-infrared vein images are typically grayscale ($C = 1$), $C$ is retained to accommodate standard backbone architectures that may duplicate channels for compatibility. Let $\mathcal{Y} = \{1, 2, \ldots, K\}$ be the discrete label space, where $K$ is the total number of identity classes. A typical vein dataset containing $N$ samples is defined as $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, where $x_i \in \mathcal{X}$ and its corresponding label $y_i \in \mathcal{Y}$.

The goal is to optimize a deep neural network $f_\theta(\cdot)$, where $\theta$ denotes the learnable parameters of the model. We aim to minimize the empirical risk:

$$\mathcal{R}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_{\text{CE}}\big(f_\theta(x_i), y_i\big), \tag{1}$$

where $\mathcal{L}_{\text{CE}}(\cdot)$ represents the standard Cross-Entropy loss function. For a sample $(x_i, y_i)$, it is formulated as:

$$\mathcal{L}_{\text{CE}}\big(f_\theta(x_i), y_i\big) = -\log\big(f_\theta(x_i)_{y_i}\big), \tag{2}$$

However, in the "small-sample" context of AGVBench, the empirical distribution of $\mathcal{D}$ is a sparse approximation of the true distribution $P(\mathcal{X}, \mathcal{Y})$. To bridge this gap, here we introduce a generalized augmentation policy $\mathcal{A}$ that generates virtual samples $(\tilde{x}, \tilde{y})$ from the vicinity of the original training data $(x, y)$. Since a practical training pipeline implicitly incorporates both original and synthesized data (e.g., by treating the identity mapping as a subset of $\mathcal{A}$), the augmented training objective can be rigorously formulated by conditioning the augmentation distribution on the original empirical distribution:

$$\min_{\theta}\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\mathbb{E}_{(\tilde{x},\tilde{y})\sim\mathcal{A}(x,y)}[\mathcal{L}_{\text{CE}}\big(f_\theta(\tilde{x}), \tilde{y}\big)]\right]. \tag{3}$$

Here, policy $\mathcal{A}$ acts as a specific conditional data augmentation mapping, and different designs of this mapping can significantly alter the distribution of vein features in the latent space.

### B. Data Augmentation

Data augmentation techniques artificially expand dataset diversity to enhance model generalization and robustness, broadly encompassing Single Image Augmentation, Multi-Image Augmentation, and Label Enhancement. Single-image augmentation methods apply various transformations to individual training samples. These approaches have evolved from foundational geometric and pixel-level perturbations to advanced strategies that force networks to learn globally distributed features via spatial occlusion [7], [18]. To address the suboptimal nature of manual designs, researchers leveraged search algorithms to dynamically identify optimal transformation sequences [5], [6], [13], while also developing sophisticated operations such as saliency-guided [19] and learnable transformations [14]. To expand the virtual sample space, multi-image augmentation methods construct smoother decision boundaries by blending multiple images and interpolating their corresponding labels. This paradigm originated from global linear interpolation across sample pixels [8], and was refined by spatial mask-based cutting and pasting to preserve local structural integrity while blending [9], [15], [20], [21]. To prevent the loss of critical foreground information, subsequent works incorporated saliency-guided priors [22], [23] and automated end-to-end adaptive mixing frameworks [10], [11]. Beyond manipulating visual inputs, label enhancement approaches alleviate model overconfidence by directly regularizing the training targets. The methodology has evolved from applying static uniform softening to one-hot vectors [24], [25], to generating adaptive dynamic soft labels using historical predictions [26] or parameterized distributions [27].

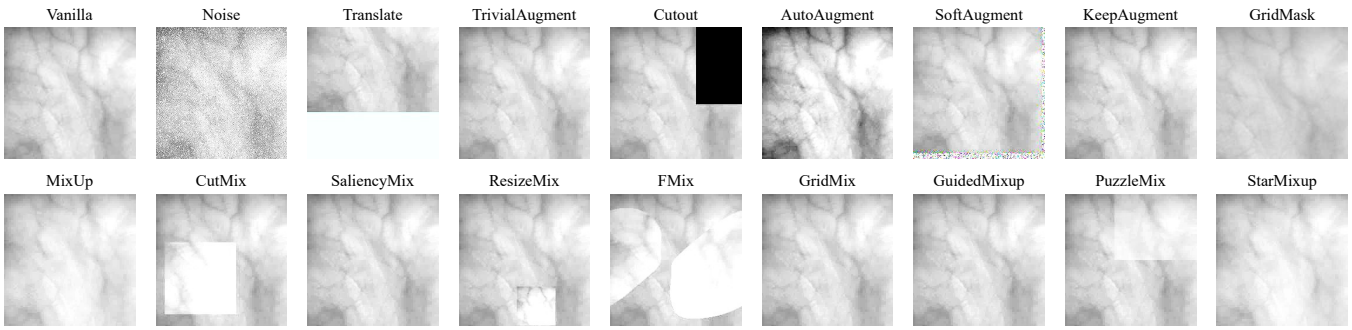


Fig. 2. Visualization of various data augmentation techniques applied to a sample vein image. The original (Vanilla) image is shown alongside standard single image transformations (top row) and advanced image mixing strategies (bottom row).

### C. Advanced Vein Recognition

Due to the strong absorption of near-infrared light by hemoglobin, infrared imaging captures highly discriminative vein patterns for biometrics [3]. However, the fine-grained nature and topological sensitivity of vein structures make robust feature extraction particularly challenging.

Early vein recognition primarily relied on traditional methods, based on hand-crafted feature extractors [28]–[30] or traditional machine learning algorithms [4], [31], [32]. Despite their computational efficiency, these heuristic-driven approaches heavily depend on prior knowledge. Consequently, they exhibit limited representational capacity and robustness against image noise, spatial misalignment, and complex backgrounds, leading to their gradual obsolescence in tackling real-world modality degradations. To overcome the bottlenecks of manual feature engineering, deep learning has been widely adopted due to its powerful end-to-end automated representation learning capabilities. Deep learning-based methods can automatically extract profound, highly discriminative features directly from raw pixels. Besides optimizing system security [33] and lightweight architectures [34], deep learning-based vein recognition methods primarily focus on improving feature representation. To address the complex topological branches of blood vessels, recent network architectures emphasize extracting structure-aware information. These methods typically incorporate attention mechanisms [1] or adaptive spatial modeling components [2], [3], [35] to capture fine-grained geometric features, thereby improving the generalization performance of the model.

Nevertheless, the superior performance of deep neural networks is inherently contingent upon massive amounts of training data. Constrained by the requirement for specialized infrared capture devices and privacy concerns, existing publicly available vein datasets are generally small-scale and lack sample diversity. This severe data scarcity easily causes heavily parameterized deep models to overfit. Consequently, investigating and applying effective data augmentation techniques to artificially expand dataset scale and feature boundaries has become crucial for unlocking the full potential of deep learning and building highly robust vein recognition systems.

## III. AGVBench: Benchmark Design

This section introduces the AGVBench framework design from five aspects: supported augmentation methods, backbone models, datasets, evaluation protocol, and the experimental pipeline of the codebase. AGVBench provides a unified framework implemented in PyTorch and built upon the OpenMMLab Computer Vision Foundation (MMCV) ecosystem for model design, training, and evaluation. We start with the overview of the composition of AGVBench. As illustrated in Fig. 3, AGVBench is decoupled into multiple modular components, including model architectures (`.agvbench.models.backbones`), data preprocessing pipelines (`.agvbench.datasets`), augmentation methods (`.agvbench.models.augments`), and execution scripts (`.tools`). Vision models are divided into standard building blocks (e.g., necks and heads) in `.agvbench.models`. This enables researchers to easily construct and modify models by combining different components through centralized configuration files in `.configs`. Consequently, users can readily customize specific vein recognition pipelines and training strategies. The detailed benchmarking configurations and theoretical designs are discussed in the subsequent subsections.

### A. Augmentation Schemes

**Single Image Augmentation Methods:** Single-image augmentation schemes incorporate methods ranging from basic operations to policy-driven methods. Basic operations perform foundational geometric and photometric perturbations, including Flip, Rotate, Affine Transformation (Trans.), Blur, and Noise. Additionally, to evaluate regional spatial robustness, Cutout [7], RandomErasing [18], and GridMask [36] implement hand-crafted spatial occlusion and dropping policies. Policy-driven methods, such as AutoAugment [5], RandAugment [6], and TrivialAugment [13], use predefined search spaces to discover optimal augmentation policies. Furthermore, advanced dynamic and domain-adaptive strategies are integrated, where TeachAugment [14] and KeepAugment [19] apply importance-guided or teacher-guided transformations, while YOCO [37], SoftAugment [38], and RQ [39] perform structural and feature-level augmentations specific to complex visual patterns.

**Multi Image Augmentation Methods:** Multi-image augmentation methods involve cross-sample feature interaction and label interpolation. Foundational methods like MixUp [8] perform global pixel-level or feature-level convex interpolation. To preserve local structural integrity, CutMix [15], ResizeMix [40], GridMix [20], and RICAP [21] implement hand-crafted spatial cutting and pasting policies. FMix [41] utilizes Fourier-guided cutting masks to blend frequency domain representations. To

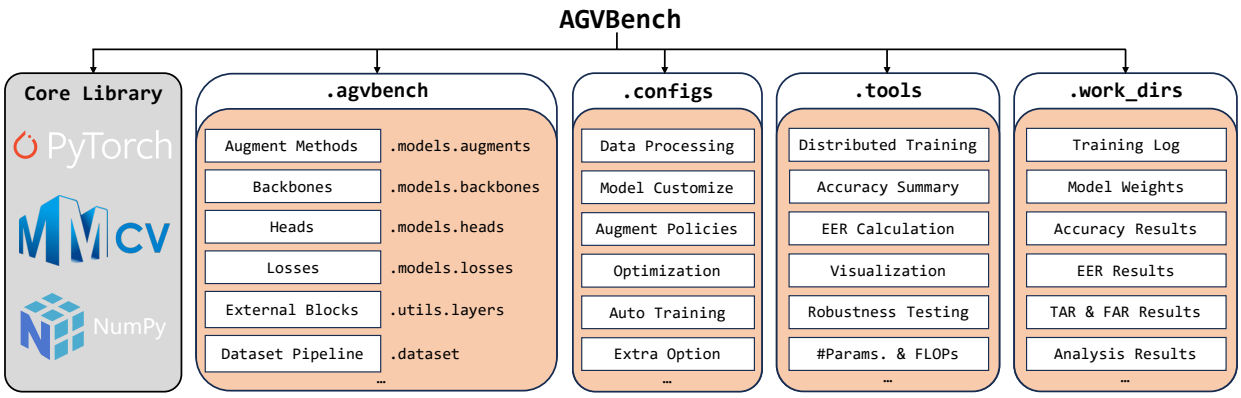


Fig. 3. Overview of the AGVBench codebase framework. Built upon PyTorch and MMCV, the framework is structured into four functional modules: (1) `.agvbench` for data pipeline and model component registration; (2) `.configs` for centralized hyperparameter and experimental setup; (3) `.tools` for distributed training execution and comprehensive metric evaluation; and (4) `.work_dirs` for storing training logs, model weights, and analysis results.

further optimize the mixing regions, SaliencyMix [22] and GuidedMixup [12] apply saliency-guided and attention-guided cutting, respectively, ensuring the preservation of critical foreground textures. Some dynamic approaches like PuzzleMix [23] utilize optimal transport-based alignment before interpolation. Moreover, the latest automated blending mechanism, like StarMixup [9], performs end-to-end online-optimizable mixing, dynamically adapting the blending policy to the current training state of the model.

**Label Enhancement Methods:** Regarding label enhancement, AGVBench implements methods that directly manipulate the target probability distributions to mitigate model overconfidence. LabelSmooth [24] and ConfidencePenalty [25] apply static, hand-crafted uniform distribution softening to the ground-truth vectors. To mitigate the impact of potentially noisy annotations and leverage self-guided learning, Bootstrapping [42] dynamically updates the training targets by blending the original hard labels with current predictive distributions of the model. OnlineLabelSmooth [26] performs dynamic label updating based on the statistical distribution of historical predictions. Furthermore, DirichletLabelSmooth [27] exploits parameterized Dirichlet distributions to generate dynamic soft labels, adapting the regularization intensity through training optimization. These label-level strategies provide critical calibration when deep architectures fit highly complex biometric features.

### B. Backbones

To evaluate how augmentations influence backbone architectures, AGVBench incorporates two representative models: classic visual backbones and domain-specific vein recognition models. For the first category, we select models spanning both CNNs and ViTs. Specifically, we employ MobileNetv2 (Mobv2) [43] as a representative of lightweight CNNs, ResNet18 (R18) [16] for standard residual learning, and ViT-small (ViT-S) [17] along with Swin Transformer-Tiny (Swin-T) [44] to evaluate global and hierarchical attention mechanisms, respectively. In parallel, we evaluate domain-specific architectures tailored for vein pattern extraction, namely FVRASNet (FVN) [33], AMPVNet (APN) [2], and StarLKNet-S (SLN-S) [9], which possess specialized priors for biometric recognition.

config.py — Step 1. Start

```python
# define dataset config file path
_base_ = "path/dataset_config_file.py"

# model settings
model = dict(
    type="",              # classification method
    aug_mode="",          # augmentation method
    aug_args=dict(...),   # augmentation configs
    backbone=dict(...),   # backbone model
    head=dict(...)        # classification head
)

# training setups
optimizer=dict(...)       # optimizer settings
runner=dict(...)          # runtime settings
other_config=dict(...)    # e.g., lr scheduler
```

execution.sh — Step 2. Execution

```
Python ./tools/train.py config.py   # train
Python ./tools/test.py               # test
```

Fig. 4. Experimental pipeline in AGVBench codebase. The workflow is configuration-driven, where users define all experimental settings in a `config.py` file before running the standardized execution scripts.

### C. Datasets

AGVBench benchmarks 5 vein identification datasets, including two vascular modalities, to facilitate cross-domain validation. Specifically, three palm vein datasets, namely SCUT1100 [2], TJU600 [45], and VERA220 [46], and two finger vein datasets, namely FV-USM [47] and SDUMLA-HMT [48], are selected. These datasets exhibit significant changes in sensor hardware, environmental lighting, and image quality, providing a challenging scenario for testing the adaptive capabilities of different augmentation operators. The details of these vein datasets are given in Section IV-A.

### D. Evaluation Protocol

**Performance Metrics:** To quantitatively evaluate the recognition performance and the robustness of the learned feature representations, we employ three primary metrics: Top-1 Accuracy (Acc.), Equal Error Rate (EER), and True Acceptance Rate at a False Acceptance Rate of $10^{-4}$ (TAR@FAR=0.0001). Accuracy measures the proportion of correctly identified samples among the total test queries, reflecting the basic

TABLE I
TOP-1 ACCURACY (%)↑, EER (%)↓, AND TAR@FAR (T@R)=0.0001 (%)↑ OF VARIOUS AUGMENTATIONS ACROSS DIFFERENT MODELS ON VERA220.

| VERA220 | R18 | | | Mobv2 | | | FVN | | | APN | | | SLK-S | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F |
| Vanilla | 71.45 | 5.20 | 51.00 | 71.55 | 5.94 | 47.18 | 59.73 | 6.38 | 35.91 | 74.00 | 3.95 | 57.18 | 70.27 | 5.46 | 44.55 |
| Flip | 67.55 | 6.27 | 42.55 | 67.27 | 6.40 | 43.82 | 51.09 | 10.98 | 19.36 | 66.73 | 4.71 | 46.82 | 63.45 | 7.31 | 35.27 |
| Rotate | 63.27 | 7.54 | 37.73 | 63.00 | 8.77 | 43.36 | 50.55 | 10.90 | 19.45 | 66.55 | 7.49 | 37.27 | 60.36 | 9.34 | 37.45 |
| Translation | 64.00 | 7.33 | 42.82 | 65.45 | 8.21 | 41.45 | 51.00 | 11.07 | 22.27 | 66.64 | 5.93 | 44.27 | 64.00 | 8.28 | 38.82 |
| Noise | 68.82 | 5.39 | 44.64 | 69.73 | 5.82 | 49.64 | 39.64 | 14.27 | 7.00 | 70.00 | 4.44 | 50.45 | 68.45 | 6.46 | 39.45 |
| Cutout [7] | 70.55 | 5.33 | 50.45 | 67.55 | 6.35 | 48.91 | 52.82 | 10.06 | 25.73 | 72.73 | 5.45 | 54.27 | 65.36 | 6.66 | 47.18 |
| GridMask [36] | 66.00 | 8.15 | 40.55 | 68.45 | 6.45 | 44.82 | 37.09 | 14.55 | 10.00 | 60.36 | 8.26 | 41.91 | 57.55 | 10.56 | 29.73 |
| RandomErasing [18] | 62.73 | 7.20 | 44.00 | 69.82 | 6.17 | 47.73 | 48.73 | 12.07 | 18.09 | 74.45 | 3.66 | 54.45 | 68.18 | 6.25 | 42.91 |
| RandomQuant [39] | 81.91 | 1.91 | 65.00 | 82.09 | 2.98 | 62.55 | 40.36 | 9.63 | 8.18 | 81.27 | 3.39 | 65.55 | 78.73 | 2.72 | 61.45 |
| AutoAugment [5] | 80.82 | 2.55 | 65.09 | 76.91 | 3.33 | 59.73 | 55.55 | 7.27 | 24.45 | 80.00 | 2.87 | 64.45 | 79.45 | 2.83 | 62.91 |
| RandAugment [6] | 74.91 | 3.47 | 56.73 | 82.09 | 2.66 | 68.91 | 36.18 | 11.69 | 8.91 | 76.64 | 3.08 | 60.91 | 84.18 | **1.94** | 65.64 |
| KeepAugment [19] | 74.36 | 5.10 | 52.73 | 70.82 | 5.96 | 47.00 | **60.45** | **6.45** | **32.36** | 75.82 | 4.18 | 51.09 | 68.27 | 4.93 | 42.55 |
| TrivialAugment [13] | **87.36** | **1.52** | **74.82** | **82.45** | **1.99** | **70.45** | 57.64 | 7.26 | 20.45 | **87.64** | **1.81** | **80.64** | **84.45** | 2.45 | **70.36** |
| TeachAugment [14] | 69.27 | 4.84 | 45.82 | 59.82 | 9.11 | 36.64 | 51.27 | 9.71 | 22.64 | 58.55 | 11.46 | 53.00 | 65.18 | 7.78 | 29.64 |
| SoftAugment [38] | 67.18 | 6.58 | 45.64 | 69.82 | 5.77 | 47.00 | 56.45 | 8.48 | 27.27 | 75.55 | 3.91 | 56.36 | 69.27 | 5.49 | 45.00 |
| YOCO [37] | 64.73 | 8.52 | 27.73 | 68.00 | 10.07 | 33.82 | 43.73 | 14.72 | 11.91 | 74.55 | 7.13 | 34.18 | 63.55 | 8.46 | 34.45 |
| RICAP [21] | 78.64 | 4.73 | 59.91 | 70.73 | 6.55 | 46.45 | 40.82 | 11.15 | 11.82 | 71.82 | 5.30 | 51.27 | 65.36 | 8.54 | 41.09 |
| MixUp [8] | 95.27 | 0.91 | 92.27 | 95.55 | 0.87 | **93.73** | 80.73 | 3.14 | 59.64 | 95.64 | 0.74 | **93.45** | **94.55** | **0.87** | **91.18** |
| CutMix [15] | 84.73 | 2.92 | 74.00 | 77.36 | 4.49 | 60.00 | 51.64 | 8.38 | 20.64 | 81.18 | 2.74 | 69.82 | 76.36 | 6.24 | 58.09 |
| FMix [41] | 81.64 | 3.36 | 64.36 | 75.55 | 4.65 | 58.45 | 56.18 | 7.89 | 27.27 | 81.64 | 3.17 | 71.09 | 70.64 | 7.10 | 51.73 |
| GridMix [20] | 78.09 | 3.87 | 61.82 | 75.09 | 4.70 | 57.55 | 42.45 | 11.37 | 12.27 | 73.36 | 4.82 | 52.27 | 68.45 | 8.83 | 48.64 |
| ResizeMix [40] | 83.00 | 3.56 | 70.27 | 75.82 | 4.89 | 63.09 | 51.18 | 9.62 | 13.18 | 81.18 | 3.37 | 68.00 | 70.36 | 7.19 | 48.27 |
| SaliencyMix [22] | 83.36 | 2.80 | 72.45 | 79.09 | 3.64 | 61.55 | 55.09 | 7.64 | 25.91 | 82.18 | 3.12 | 72.00 | 73.45 | 6.73 | 58.91 |
| PuzzleMix [23] | **95.55** | **0.83** | **93.36** | **95.91** | **0.83** | 91.45 | 76.36 | 3.55 | 51.18 | 94.36 | 1.08 | 89.73 | 92.00 | 1.26 | 86.27 |
| GuidedMixup [12] | 66.09 | 7.19 | 44.45 | 69.73 | 6.54 | 44.27 | – | – | – | 65.73 | 29.64 | 0.64 | 69.27 | 6.29 | 44.36 |
| StarMixup [9] | 94.91 | 0.96 | 92.27 | 92.64 | 1.11 | 87.55 | **83.45** | **2.62** | **67.82** | **96.27** | **0.71** | 92.91 | 93.09 | 1.16 | 89.73 |
| LabelSmoothing [24] | 82.64 | **2.54** | **66.09** | 75.91 | 4.88 | 55.27 | **67.09** | 5.35 | **47.91** | 89.91 | 2.19 | 76.00 | 72.00 | 5.28 | 46.09 |
| OnlineLabelSmooth [26] | 74.82 | 4.24 | 54.91 | 70.18 | 5.08 | 50.00 | 63.36 | 5.91 | 33.73 | 80.00 | 3.50 | 63.82 | **73.55** | **5.15** | **48.91** |
| ConfidencePenalty [25] | 70.73 | 4.82 | 52.27 | 70.82 | 5.72 | 49.91 | 61.18 | 6.80 | 34.09 | 74.64 | 3.36 | 53.27 | 66.45 | 7.81 | 39.00 |
| DirichletLabelSmooth [27] | **83.82** | 2.76 | 64.82 | **77.45** | **4.26** | **55.82** | 66.64 | **4.82** | 44.82 | **91.45** | **2.11** | **81.18** | 71.09 | 5.57 | 39.64 |
| Bootstrapping [42] | 65.64 | 6.63 | 42.73 | 52.55 | 10.99 | 24.55 | 61.55 | 6.76 | 33.36 | 76.45 | 3.59 | 55.82 | 43.00 | 13.55 | 12.36 |

classification capability of models. Considering that biometric verification is essentially a threshold-dependent matching task, we adopt EER as a more rigorous criterion, defined as the point where the False Acceptance Rate (FAR) equals the False Rejection Rate (FRR). A lower EER indicates a better balance between general security and convenience. Furthermore, to evaluate the reliability under highly stringent security scenarios of models, we report TAR@FAR=0.0001. This metric specifically highlights the capacity of the system to recognize genuine users while strictly suppressing impostors, signifying superior discriminative power in the embedding space.

**Robustness Evaluation:** To comprehensively assess the reliability and robustness of vein recognition models in real-world complex scenarios, AGVBench integrates a multi-dimensional robustness evaluation suite. We introduce these evaluations as follows.

*1) Calibration:* In biometric security systems like vein recognition, a well-calibrated model is crucial because the output probability should reliably reflect the likelihood of a correct match. However, modern deep neural networks, while highly accurate, often suffer from overconfidence, where the predicted probabilities are systematically higher than the actual precision, particularly when trained with intense data augmentations. To quantitatively assess this, we adopt the Expected Calibration Error (ECE) as the primary metric. The ECE partitions the predictions into $M$ equally spaced bins (e.g., $M = 10$) based on their confidence scores and calculates the weighted average of the gap between accuracy and average confidence of each bin:

$$\mathrm{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{n} |\mathrm{Acc.}(B_m) - \mathrm{Conf.}(B_m)|, \tag{4}$$

where $n$ is the total number of samples and $|B_m|$ is the number of samples in the $m$-th bin. In our benchmark, we investigate how different augmentation strategies affect model calibration.

*2) Corruption:* In practical deployments, vein recognition systems frequently encounter unpredictable environmental and sensor-induced degradations. Therefore, the corruption evaluation aims to explore the robustness and generalization of models against these out-of-distribution image distortions. Following previous works [10], [11], [49] and the robustness evaluation protocol of ImageNet-C [50], our benchmark implements a suite of 19 distinct corruption methods, including various degradations such as Gaussian noise and motion blur. Our experiments (Full results in Tables XIV and XIII) reveal a critical vulnerability: existing vein recognition models suffer catastrophic performance collapse even at the lowest severity level (C3). We hypothesize that this sensitivity arises because even minor corruptions are sufficient to obliterate the fine-grained topological textures of veins that are essential for accurate identity inference. Consequently, to enable a more granular and meaningful assessment, we expanded the evaluation spectrum by introducing two novel, lower-intensity severity levels (C1/C2) specifically tailored to the fragility of vein patterns.

TABLE II
TOP-1 ACCURACY (%)↑, EER (%)↓, AND TAR@FAR (T@R)=0.0001 (%)↑ OF VARIOUS AUGMENTATIONS ACROSS DIFFERENT MODELS ON TJU600.

| TJU600 | R18 | | | Mobv2 | | | FVN | | | APN | | | SLK-S | | | ViT-S | | | Swin-T | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F |
| Vanilla | 85.55 | 1.72 | 81.23 | 80.33 | 2.68 | 74.80 | 64.33 | 5.31 | 48.83 | 75.02 | 3.76 | 66.91 | 76.95 | 4.78 | 57.05 | 61.77 | 5.73 | 46.76 | 76.55 | 2.83 | 66.06 |
| Flip | 79.50 | 2.75 | 73.55 | 76.15 | 3.65 | 69.18 | 66.83 | 7.16 | 45.85 | 69.68 | 4.91 | 59.46 | 73.80 | 5.79 | 49.81 | 58.58 | 6.48 | 42.81 | 71.90 | 3.71 | 61.90 |
| Rotate | 82.75 | 1.86 | 77.83 | 79.75 | 3.06 | 73.75 | 59.58 | 5.83 | 46.26 | 73.97 | 3.98 | 65.56 | 75.30 | 4.09 | 58.61 | 64.85 | 4.98 | 51.78 | 75.55 | 2.91 | 65.75 |
| Translation | 85.67 | 1.76 | 81.73 | 81.52 | 2.76 | 75.85 | 63.57 | 4.95 | 50.58 | 78.07 | 3.58 | 71.78 | 77.30 | 3.76 | 58.11 | 65.87 | 4.85 | 52.66 | 76.13 | 2.60 | 67.36 |
| Noise | 85.17 | 2.38 | 77.61 | 81.28 | 2.86 | 75.88 | 59.90 | 6.08 | 44.13 | 73.87 | 3.91 | 64.11 | 80.03 | 2.63 | 73.61 | 59.70 | 6.01 | 44.91 | 67.30 | 4.61 | 53.83 |
| Cutout [7] | 87.38 | 1.35 | 84.35 | 81.38 | 2.46 | 74.53 | 65.95 | **4.38** | 53.03 | **80.53** | **2.53** | 74.45 | 78.00 | 2.98 | 69.75 | 67.88 | 3.85 | 56.13 | 79.93 | 2.08 | 72.00 |
| GridMask [36] | 81.82 | 2.22 | 73.63 | 77.73 | 3.18 | 69.58 | 56.52 | 6.28 | 36.56 | 69.45 | 4.39 | 57.73 | 75.20 | 3.51 | 65.26 | 61.68 | 4.64 | 47.75 | 73.75 | 3.00 | 63.45 |
| RandomErasing [18] | 87.37 | 1.56 | 83.68 | 82.92 | 2.69 | 77.53 | 63.43 | 4.94 | 49.70 | 78.77 | 2.91 | 72.11 | 78.23 | 4.21 | 58.70 | 64.53 | 4.61 | 52.05 | 77.53 | 2.58 | 68.55 |
| RandomQuant [39] | 82.82 | 2.56 | 78.16 | 80.33 | 3.03 | 74.68 | 62.83 | 5.96 | 47.80 | 70.58 | 5.20 | 61.35 | 83.82 | 2.01 | 78.85 | 68.70 | 3.66 | 57.55 | 56.32 | 6.50 | 40.85 |
| AutoAugment [5] | 88.28 | 1.59 | 85.23 | 85.93 | 2.00 | 82.03 | 62.93 | 5.78 | 48.06 | 77.17 | 3.83 | 70.08 | **86.43** | **1.88** | **83.55** | 63.33 | 5.25 | 48.05 | 80.82 | 2.12 | 74.05 |
| RandAugment [6] | 87.78 | 1.50 | 85.00 | 82.85 | 2.76 | 77.65 | 66.83 | 4.77 | 53.33 | 80.12 | 3.12 | 74.86 | 86.27 | 2.03 | 76.71 | 71.25 | **3.58** | 61.05 | 81.12 | 2.19 | 74.46 |
| KeepAugment [19] | 85.17 | 1.86 | 80.73 | 81.88 | 2.89 | 76.91 | 63.92 | 5.16 | 50.35 | 75.80 | 4.22 | 68.38 | 79.70 | 2.63 | 73.36 | 62.95 | 5.70 | 48.71 | 76.67 | 2.71 | 67.83 |
| TrivialAugment [13] | **89.23** | **1.18** | **87.78** | **86.45** | **1.93** | **83.40** | **69.23** | 4.63 | **56.65** | 79.97 | 3.00 | 74.55 | 84.97 | 2.70 | 73.73 | 70.40 | 4.01 | 59.98 | **82.02** | 2.09 | **76.21** |
| TeachAugment [14] | 81.48 | 2.84 | 77.28 | 75.20 | 4.94 | 69.75 | 42.68 | 9.56 | 23.13 | 60.28 | 10.63 | 51.95 | 70.47 | 6.28 | 63.25 | **72.50** | 3.68 | **62.43** | 81.42 | **1.85** | 74.88 |
| SoftAugment [38] | 84.83 | 1.52 | 80.96 | 81.78 | 2.70 | 75.98 | 64.12 | 5.03 | 50.61 | 76.13 | 3.51 | 69.50 | 77.52 | 2.94 | 70.11 | 63.42 | 5.48 | 49.21 | 76.53 | 2.60 | 66.70 |
| YOCO [37] | 81.45 | 2.70 | 73.28 | 82.52 | 2.03 | 77.56 | 62.48 | 5.53 | 48.26 | 80.10 | 3.09 | **75.03** | 80.00 | 2.46 | 73.86 | 61.50 | 5.48 | 49.90 | 68.83 | 3.68 | 57.60 |
| RICAP [21] | 87.60 | 1.47 | 82.97 | 84.17 | 2.03 | 77.17 | 63.85 | 5.29 | 48.40 | 69.77 | 5.01 | 57.73 | 87.18 | 1.77 | 82.88 | 53.13 | 13.47 | 41.80 | 61.73 | 6.87 | 48.65 |
| MixUp [8] | 93.90 | 0.84 | 92.51 | 93.10 | 1.04 | 90.98 | 78.32 | 3.36 | 67.70 | 91.80 | 1.43 | 89.85 | 94.70 | 0.68 | 92.73 | **81.12** | **2.68** | **74.40** | 90.87 | 1.18 | 88.21 |
| CutMix [15] | 92.97 | 0.79 | 91.16 | 91.98 | 1.00 | 89.43 | 70.15 | 4.42 | 58.41 | 85.68 | 2.24 | 80.75 | 91.20 | 0.99 | 88.55 | 68.23 | 6.68 | 58.68 | 75.95 | 3.56 | 67.81 |
| FMix [41] | 93.13 | 0.86 | 90.01 | 89.30 | 1.33 | 85.25 | 70.48 | 4.44 | 54.95 | 82.42 | 2.66 | 76.10 | 87.82 | 1.46 | 83.45 | 65.40 | 6.91 | 54.86 | 81.08 | 2.31 | 73.73 |
| GridMix [20] | 86.88 | 1.58 | 83.16 | 86.37 | 1.88 | 81.70 | 44.47 | 9.11 | 24.03 | 80.33 | 2.71 | 75.75 | 84.58 | 2.13 | 80.45 | 66.93 | 6.51 | 56.76 | 77.03 | 3.50 | 67.51 |
| ResizeMix [40] | 92.85 | 0.79 | 90.96 | 91.38 | 0.95 | 88.76 | 70.60 | 3.88 | 58.58 | 84.67 | 2.38 | 78.53 | 90.57 | 0.96 | 88.48 | 64.28 | 8.49 | 53.56 | 75.07 | 3.90 | 65.03 |
| SaliencyMix [22] | 92.80 | 0.85 | 90.73 | 91.27 | 1.02 | 88.13 | 70.40 | 4.50 | 58.26 | 86.05 | 2.21 | 81.40 | 90.53 | 1.18 | 88.76 | 69.40 | 6.01 | 61.21 | 78.77 | 2.90 | 71.26 |
| PuzzleMix [23] | **95.25** | **0.46** | **94.45** | **94.63** | **0.66** | **93.80** | 77.97 | **2.96** | 69.13 | 89.92 | 1.50 | 86.86 | **96.02** | **0.40** | **95.45** | 79.17 | 3.08 | 72.26 | 87.12 | 1.46 | 83.21 |
| GuidedMixup [12] | 77.45 | 4.58 | 69.81 | 74.98 | 4.50 | 67.90 | 36.03 | 14.55 | 18.70 | 70.78 | 6.52 | 61.87 | 80.65 | 2.82 | 74.62 | 65.97 | 7.53 | 51.23 | 71.43 | 5.66 | 57.71 |
| StarMixup [9] | 93.68 | 0.74 | 91.66 | 92.90 | 0.98 | 91.01 | **79.65** | 3.18 | **70.03** | **92.55** | 1.19 | **89.98** | 94.97 | 0.71 | 92.98 | 78.77 | 3.23 | 70.20 | **90.98** | 1.18 | **88.31** |
| LabelSmoothing [24] | **94.88** | 0.84 | 93.71 | 90.23 | **1.58** | 87.11 | 75.43 | 3.73 | 63.51 | 91.20 | **1.63** | 87.80 | 89.33 | 1.38 | 84.75 | 65.98 | **4.64** | 51.56 | 86.15 | **1.51** | 79.68 |
| OnlineLabelSmooth [26] | 87.55 | 1.38 | 84.63 | 82.85 | 2.25 | 78.56 | 68.02 | 4.48 | 55.66 | 79.75 | 2.96 | 73.80 | 84.67 | 2.09 | 77.50 | 65.78 | 4.68 | 51.61 | 81.50 | 2.33 | 73.10 |
| ConfidencePenalty [25] | 86.17 | 1.90 | 81.48 | 80.45 | 2.99 | 73.85 | 65.95 | 4.83 | 52.21 | 76.23 | 3.73 | 67.41 | 79.65 | 2.76 | 71.83 | 62.78 | 5.48 | 48.20 | 77.20 | 2.81 | 67.81 |
| DirichletLabelSmooth [27] | 94.83 | **0.72** | **93.73** | **91.08** | 1.78 | **87.68** | **76.63** | **3.66** | **64.21** | **91.42** | 1.93 | **87.95** | **90.20** | **1.28** | **85.48** | **66.18** | 4.71 | **51.80** | **86.50** | 1.64 | **79.70** |
| Bootstrapping [42] | 76.13 | 3.11 | 67.43 | 64.23 | 5.68 | 53.20 | 32.97 | 11.34 | 15.56 | 76.50 | 3.94 | 68.76 | 66.97 | 4.83 | 55.53 | 62.92 | 5.31 | 48.61 | 76.93 | 2.71 | 67.68 |

*3) Adversarial Attacks:* To assess the robustness of security and features in vein recognition systems under malicious perturbations, we introduce a series of white-box adversarial attack experiments. Specifically, we implement two representative attack algorithms: FGSM [51] and PGD [52]. Considering the low contrast of vein textures, we set the maximum perturbation to $\epsilon = 0.2/255$ under the $\ell_\infty$ norm, which represents a significant challenge for vein recognition. For the PGD attack, we employ a step size of $\alpha = 0.05/255$ and 10 iterations to iteratively search for imperceptible perturbations within the $\ell_\infty$ bound that maximize the classification loss and provoke model misclassification.

*4) Occlusion:* In practical scenarios, vein patterns are frequently obscured by sensor smudges or finger misalignment. A robust biometric model must therefore leverage global topological structures rather than relying on easily compromised local patches. To systematically evaluate this fault tolerance, we introduce a spatial occlusion assessment by randomly masking continuous square regions in the input images with zero values. Specifically, we vary the occlusion ratio (the masked area relative to the total image dimensions) from $0\%$ to $50\%$ with a $2\%$ increment, constructing 25 distinct test subsets. We evaluate the performance with 10 augmentations on these subsets.

**Efficiency Evaluation:** We further evaluate the computational efficiency of the classifier integrated with various augmentation strategies. We postulate that an ideal augmentation approach should achieve a superior trade-off between performance gain and resource consumption. To quantify this, we employ three key metrics: training time occupancy ($T_{\text{train}}$) per epoch, peak memory footprint ($M_{\text{peak}}$), and computational complexity measured in GFLOPs. We also consider the ratio of extra learnable parameters, denoted as $P_{\text{aug}}$, defined as $P_{\text{aug}} = \frac{P_{\text{extra}}+P_{\text{base}}}{P_{\text{base}}}$, where $P_{\text{base}}$ and $P_{\text{extra}}$ denote the number of parameters of the vanilla backbone and the number of parameters of the additional modules introduced by the augmentation strategies, respectively.

To provide a holistic evaluation of these diverse augmentation strategies, we introduce the Augmentation Performance and Efficiency Excellence (**APEX**) score based on the principle of Pareto Efficiency. Instead of an arbitrary weighted sum, we identify the Non-dominated solutions across the multi-dimensional objective space (Performance, $T_{\text{train}}$, $M_{\text{peak}}$, GFLOPs, and $P_{\text{aug}}$) as a multi-objective optimization problem.

Non-dominated augmentations are assigned Rank-1 APEX, such that no other augmentation can improve one metric without degrading at least one other. By iteratively removing Rank-1 individuals, the subsequent non-dominated sets are defined as Rank-2 and higher. This ranking paradigm classifies methods beyond pure accuracy based on their strategic location on the efficiency-performance frontier and effectively eliminates redundant methods with high computational costs but only marginal performance gains.

### *E. Experimental Pipeline of AGVBench Codebase*

As illustrated in Fig. 4, the experimental workflow of AGVBench is structured to support the complete process of vein recognition evaluations. The pipeline is primarily controlled through configuration files located in `.configs` directory. In these files, users specify the target dataset, data

TABLE III
TOP-1 ACCURACY (%)↑, EER (%)↓, AND TAR@FAR (T@F)=0.0001 (%)↑ OF VARIOUS AUGMENTATIONS ACROSS DIFFERENT MODELS ON SCUT1100.

| SCUT1100 | R18 | | | Mobv2 | | | FVN | | | APN | | | SLK-S | | | ViT-S | | | Swin-T | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F |
| Vanilla | 96.05 | 0.30 | 97.30 | 95.27 | 0.32 | 96.80 | 92.96 | 0.59 | 94.12 | 94.20 | 0.55 | 95.43 | 93.96 | 0.49 | 95.30 | 76.42 | 2.56 | 72.60 | 94.07 | 0.36 | 95.80 |
| Flip | 94.89 | 0.41 | 96.05 | 93.44 | 0.56 | 94.85 | 92.60 | 0.66 | 93.80 | 92.07 | 0.70 | 92.85 | 91.65 | 0.78 | 92.94 | 67.55 | 3.90 | 60.90 | 93.67 | 0.43 | 95.03 |
| Rotate | 95.05 | 0.42 | 96.34 | 94.53 | 0.51 | 95.89 | 89.91 | 0.99 | 90.63 | 93.07 | 0.59 | 94.16 | 93.38 | 0.56 | 94.63 | 74.29 | 2.73 | 69.87 | 93.87 | 0.45 | 95.58 |
| Translation | 95.60 | 0.32 | 97.05 | 94.84 | 0.46 | 96.21 | 91.45 | 0.72 | 92.40 | 94.53 | 0.54 | 95.85 | 93.05 | 0.63 | 94.69 | 75.96 | 2.87 | 71.85 | 95.02 | 0.34 | 96.16 |
| Noise | 88.65 | 0.96 | 89.00 | 94.62 | 0.40 | 96.07 | 85.87 | 1.40 | 85.49 | 92.49 | 0.65 | 93.61 | 93.73 | 0.43 | 95.21 | 72.40 | 3.54 | 66.52 | 90.56 | 0.74 | 91.41 |
| Cutout [7] | 86.89 | 1.05 | 85.83 | 80.67 | 1.67 | 77.41 | 90.05 | 0.85 | 89.78 | 91.02 | 0.69 | 91.70 | 84.45 | 1.27 | 83.50 | 82.71 | 1.72 | 80.70 | 94.22 | 0.41 | 95.69 |
| GridMask [36] | 85.80 | 1.42 | 81.61 | 84.13 | 2.11 | 76.40 | 79.80 | 2.16 | 72.21 | 81.35 | 2.36 | 68.14 | 80.65 | 2.09 | 73.89 | 71.07 | 2.87 | 64.47 | 88.49 | 0.92 | 88.69 |
| RandomErasing [18] | 96.00 | 0.30 | 97.34 | 94.71 | 0.39 | 96.16 | 91.35 | 0.76 | 92.10 | 95.00 | **0.40** | 96.12 | 93.87 | 0.48 | 95.25 | 75.64 | 2.54 | 71.40 | 94.76 | 0.36 | 96.36 |
| RandomQuant [39] | 95.64 | 0.40 | 97.01 | 95.27 | 0.41 | 96.45 | 84.49 | 1.69 | 82.94 | 92.58 | 0.71 | 93.72 | 95.24 | 0.43 | 96.67 | 84.82 | 1.34 | 83.78 | 92.25 | 0.52 | 93.29 |
| AutoAugment [5] | 96.95 | 0.25 | 98.25 | 95.89 | 0.34 | 97.29 | 92.93 | 0.67 | 94.29 | 93.35 | 0.54 | 94.94 | 95.65 | 0.32 | 97.03 | 85.58 | 1.25 | 85.09 | 96.38 | 0.20 | 97.92 |
| RandAugment [6] | 97.36 | 0.24 | 98.47 | **96.82** | **0.23** | 97.96 | 94.78 | 0.44 | 96.41 | 94.78 | 0.47 | 96.09 | **97.09** | **0.23** | **98.29** | 87.71 | **1.00** | 87.92 | 96.29 | 0.20 | 98.03 |
| KeepAugment [19] | 95.89 | 0.32 | 97.12 | 94.91 | 0.44 | 96.10 | 92.89 | 0.66 | 93.92 | 93.93 | 0.67 | 95.07 | 94.00 | 0.43 | 95.60 | 76.76 | 2.54 | 72.96 | 94.67 | 0.36 | 96.18 |
| TrivialAugment [13] | **97.82** | **0.18** | **98.83** | 96.73 | 0.27 | **98.20** | **95.11** | **0.39** | **96.56** | **95.22** | 0.43 | **96.63** | 96.20 | 0.31 | 97.60 | 86.89 | 1.10 | 86.72 | **96.58** | **0.17** | **98.09** |
| TeachAugment [14] | – | – | – | 79.44 | 2.01 | 74.65 | – | – | – | 49.09 | 6.98 | 32.94 | – | – | – | **88.42** | 1.05 | **88.43** | 96.45 | 0.18 | 97.94 |
| SoftAugment [38] | 95.42 | 0.40 | 96.98 | 94.56 | 0.47 | 95.94 | 92.31 | 0.70 | 93.49 | 93.98 | 0.45 | 95.43 | 94.02 | 0.50 | 94.96 | 75.36 | 2.81 | 70.50 | 93.87 | 0.45 | 95.25 |
| YOCO [37] | 92.15 | 3.34 | 50.03 | 78.02 | 2.63 | 69.85 | 82.02 | 7.61 | 42.63 | 84.96 | 8.14 | 19.90 | 74.60 | 46.47 | 0.00 | 75.31 | 2.47 | 71.45 | 92.04 | 0.56 | 93.54 |
| RICAP [21] | 98.05 | 0.20 | 98.84 | 95.73 | 0.45 | 95.98 | 85.62 | 1.97 | 81.20 | 93.07 | 0.85 | 92.51 | 97.56 | 0.20 | 98.40 | 80.93 | 2.35 | 77.25 | 93.13 | 0.77 | 93.73 |
| MixUp [8] | 99.09 | **0.07** | 99.63 | 98.98 | 0.08 | 99.56 | **97.22** | **0.25** | **97.98** | 97.80 | **0.23** | 98.47 | 99.02 | **0.05** | **99.60** | 92.42 | **0.74** | 92.67 | **98.87** | **0.07** | **99.63** |
| CutMix [15] | 98.76 | 0.10 | 99.41 | 97.49 | 0.25 | 98.23 | 94.22 | 0.48 | 94.90 | 96.60 | 0.38 | 97.78 | 98.25 | 0.11 | 99.27 | 89.11 | 1.09 | 88.78 | 97.60 | 0.14 | 98.72 |
| FMix [41] | 98.49 | 0.10 | 99.38 | 96.65 | 0.29 | 97.76 | 95.20 | 0.48 | 96.20 | 95.49 | 0.45 | 96.43 | 97.16 | 0.18 | 98.29 | 82.78 | 1.78 | 80.98 | 97.47 | 0.10 | 98.85 |
| GridMix [20] | 97.73 | 0.22 | 98.34 | 96.18 | 0.34 | 97.23 | 92.40 | 0.77 | 92.61 | 93.51 | 0.67 | 94.58 | 97.24 | 0.21 | 98.09 | 85.47 | 1.52 | 84.07 | 95.91 | 0.27 | 97.43 |
| ResizeMix [40] | 98.45 | 0.08 | 99.21 | 97.51 | 0.16 | 98.27 | 94.71 | 0.58 | 95.34 | 96.20 | 0.41 | 97.32 | 98.07 | 0.11 | 98.90 | 83.62 | 1.92 | 81.69 | 96.87 | 0.20 | 98.25 |
| SaliencyMix [22] | 98.60 | 0.09 | 99.41 | 97.82 | 0.18 | 98.69 | 94.75 | 0.56 | 95.54 | 96.58 | 0.31 | 97.50 | 98.33 | 0.09 | 99.21 | 89.78 | 0.96 | 89.98 | 97.49 | 0.12 | 98.69 |
| PuzzleMix [23] | 99.13 | 0.10 | 99.43 | 98.84 | 0.14 | 99.29 | 96.38 | 0.41 | 96.94 | 97.25 | 0.38 | 97.74 | 98.96 | 0.11 | 99.50 | 92.00 | 0.81 | 91.96 | 98.07 | 0.16 | 98.87 |
| GuidedMixup [12] | **99.35** | **0.07** | **99.75** | **99.15** | **0.06** | **99.73** | 96.91 | 0.26 | 97.84 | 97.91 | 0.29 | 98.56 | **99.05** | 0.09 | 99.58 | **92.45** | 0.76 | **92.85** | 92.85 | 0.47 | 94.10 |
| StarMixup [9] | 99.25 | **0.07** | 99.70 | 99.11 | 0.08 | 99.61 | 96.95 | 0.33 | 97.70 | **98.00** | 0.26 | **98.65** | 98.93 | 0.12 | 99.45 | 91.38 | 0.89 | 91.94 | 98.78 | **0.07** | 99.49 |
| LabelSmoothing [24] | 98.35 | 0.18 | 98.81 | 96.78 | 0.31 | **97.45** | 96.38 | **0.39** | 96.90 | 97.78 | 0.30 | 98.21 | 96.27 | **0.29** | 97.25 | 77.45 | 2.76 | 72.78 | **97.16** | 0.23 | 97.50 |
| OnlineLabelSmooth [26] | 97.29 | 0.18 | 98.36 | 95.85 | **0.27** | 97.16 | 95.33 | 0.41 | 96.63 | 96.02 | 0.36 | 97.00 | 95.75 | **0.29** | 96.81 | **78.04** | **2.05** | **74.50** | 95.98 | 0.25 | 97.45 |
| ConfidencePenalty [25] | 96.05 | 0.32 | 97.40 | 94.69 | 0.48 | 96.16 | 92.65 | 0.63 | 93.54 | 94.04 | 0.48 | 95.60 | 93.95 | 0.53 | 95.25 | 76.15 | 2.70 | 72.43 | 94.65 | 0.38 | 96.16 |
| DirichletLabelSmooth [27] | **98.45** | **0.16** | **98.94** | **96.85** | 0.32 | 97.41 | **96.56** | 0.45 | **97.05** | **97.84** | **0.29** | **98.23** | **96.95** | 0.32 | **97.60** | 77.78 | 2.74 | 73.18 | 96.89 | **0.19** | **97.60** |
| Bootstrapping [42] | 94.47 | 0.51 | 95.67 | 85.16 | 1.54 | 84.38 | 92.76 | 0.64 | 94.18 | 94.56 | 0.41 | 95.80 | 88.76 | 1.03 | 88.89 | 76.82 | 2.67 | 72.90 | 94.67 | 0.34 | 96.20 |

augmentation methods, backbone architecture, and training hyperparameters. During initialization, the framework reads these configurations and instantiates the required components from `.agvbench.models`, which maintains a registry of backbones, classification heads, and loss functions. The data pipeline is also built automatically according to the specified dataset format. For the execution phase, `.tools` directory provides scripts for both single-node and distributed training. It also includes specific evaluation utilities to extract feature embeddings and calculate verification metrics such as the Equal Error Rate (EER). Finally, the repository is accompanied by basic documentation and baseline results to assist researchers in reproducing the benchmarks or integrating new algorithms.

## IV. EXPERIMENTS AND RESULTS

In this section, we present the experimental evaluation conducted using the AGVBench framework. We first detail the experimental setup, including dataset configurations, implementation details, and training hyperparameters. Next, we report the baseline performance of the selected general and specialized backbone architectures on multiple public vein datasets. Subsequently, we systematically analyze the impact of various data augmentation strategies on model generalization and robustness. For all experiments, we reported accuracy using the median of the last 10 epochs and marked the best and second-best results in **bold** and underlined, respectively. Results worse than the Vanilla are marked in gray. The best result in each column is marked with the blue background.

### A. Experimental Setup

**Dataset Information:** We evaluate the methods on 5 vein datasets: (1) For palm vein datasets, **VERA220** [46] comprises 110 subjects with 10 samples each (2,200 images), where 5 images per subject are used for training and 5 for testing. It is acquired in open environments with slight pose variations and ambient light interference. **TJU600** [45] contains 300 subjects and 40 samples per subject (12,000 images), from which 10 images per subject are allocated for training and 10 for testing. This dataset is collected across two sessions in a semi-enclosed space with diverse hand postures and illumination conditions. **SCUT1100** [2] includes 550 subjects with 20 samples per subject (11,000 images), partitioned into 5 training and 5 testing images per subject, captured in unconstrained dynamic scenarios with obvious out-of-plane rotation and grayscale variations. (2) For finger vein datasets, **FV-USM** [47] has 123 subjects and 12 samples per subject (1,476 images) from two collection sessions, evenly split into 6 for training and 6 for testing to evaluate temporal intra-class robustness. **SDUMLA-HMT** [48] involves 106 subjects with 36 samples per subject (3,816 images), providing multi-finger vein data with rich variations in finger placement and orientation, where 4 images per finger class are used for training and 2 for testing.

**Implementation Details:** All experiments are implemented within the proposed AGVBench framework, which is built upon PyTorch and the MMCV ecosystem. During the data preprocessing stage, all input vein images are resized to a unified resolution of $224 \times 224$ to ensure compatibility across all selected backbone architectures. The models are trained

TABLE IV
TOP-1 ACCURACY (%)↑, EER (%)↓, AND TAR@FAR (T@F)=0.0001 (%)↑ OF VARIOUS AUGMENTATIONS ACROSS DIFFERENT MODELS ON FV-USM.

| FV-USM | R18 | | | Mobv2 | | | FVN | | | APN | | | SLK-S | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F |
| Vanilla | 38.52 | 16.26 | 19.58 | 43.50 | 13.62 | 24.66 | 29.00 | 21.69 | 14.84 | 69.61 | 4.91 | 52.74 | 36.42 | 17.69 | 18.43 |
| Flip | 36.45 | 16.19 | 21.04 | 41.16 | 15.25 | 22.49 | 23.14 | 24.80 | 7.72 | 61.96 | 7.18 | 41.26 | 30.12 | 20.09 | 12.97 |
| Rotate | 34.52 | 16.87 | 20.19 | 37.74 | 16.43 | 21.54 | 25.43 | 25.00 | 10.40 | 46.14 | 13.46 | 27.17 | 30.45 | 19.17 | 17.07 |
| Translation | 31.44 | 19.41 | 17.14 | 36.35 | 15.65 | 21.88 | 24.16 | 22.39 | 14.46 | 55.10 | 8.81 | 35.40 | 36.48 | 18.66 | 19.38 |
| Noise | 40.07 | 14.83 | 18.12 | 48.75 | 12.57 | 27.41 | 31.03 | 20.18 | 14.74 | 66.26 | 5.86 | 48.81 | 41.87 | 14.66 | 22.36 |
| Cutout [7] | 46.85 | 14.39 | 27.74 | 45.36 | 15.34 | 25.71 | 35.67 | 19.41 | 18.33 | 56.50 | 9.24 | 38.31 | 44.85 | 15.74 | 29.17 |
| GridMask [36] | 27.45 | 21.13 | 14.33 | 33.54 | 17.94 | 16.50 | 17.50 | 25.43 | 11.79 | 40.89 | 16.32 | 20.05 | 28.32 | 23.13 | 11.79 |
| RandomErasing [18] | 39.74 | 16.22 | 21.82 | 46.38 | 14.15 | 27.07 | 30.25 | 21.32 | 15.18 | 67.11 | 5.76 | 52.71 | 38.92 | 17.71 | 20.93 |
| RandomQuant [39] | 80.35 | 3.22 | 69.82 | 82.59 | 2.87 | 73.04 | 60.30 | 6.67 | 41.67 | 83.74 | 2.65 | 74.86 | 73.34 | 3.97 | 58.13 |
| AutoAugment [5] | 88.45 | 1.84 | 85.30 | 85.74 | 2.04 | 80.93 | 53.79 | 8.86 | 33.91 | 92.14 | 1.39 | 90.89 | 83.67 | 2.48 | 77.64 |
| RandAugment [6] | **93.90** | **1.15** | **93.50** | **94.44** | **1.06** | **94.65** | **79.44** | **5.22** | 67.34 | **95.63** | **0.71** | **95.80** | **92.31** | 1.38 | **91.50** |
| KeepAugment [19] | 41.29 | 14.95 | 21.04 | 49.63 | 11.31 | 32.62 | 30.56 | 20.22 | 16.16 | 67.34 | 5.79 | 52.00 | 38.86 | 16.54 | 19.00 |
| TrivialAugment [13] | 92.21 | 1.26 | 91.12 | 92.51 | 1.09 | 91.33 | 78.42 | 5.49 | **65.31** | 94.72 | 0.92 | 94.55 | 89.80 | **1.35** | 87.53 |
| TeachAugment [14] | 25.76 | 19.21 | 12.06 | 58.42 | 9.83 | 40.65 | 26.24 | 21.31 | 9.28 | 47.27 | 15.45 | 42.34 | 28.35 | 48.92 | 0.00 |
| SoftAugment [38] | 39.30 | 14.16 | 23.17 | 42.89 | 12.12 | 23.88 | 29.17 | 19.31 | 15.62 | 64.57 | 6.10 | 46.92 | 42.62 | 14.29 | 23.04 |
| YOCO [37] | 41.80 | 13.58 | 25.00 | 42.62 | 10.60 | 27.51 | 28.31 | 24.83 | 9.45 | 66.06 | 5.29 | 47.80 | 37.13 | 14.53 | 21.04 |
| RICAP [21] | 58.81 | 10.03 | 41.87 | 61.75 | 8.67 | 46.99 | 31.06 | 22.15 | 9.38 | 58.84 | 10.33 | 42.21 | 53.73 | 13.42 | 41.43 |
| MixUp [8] | 86.75 | 3.58 | **84.25** | 87.30 | 3.45 | **85.23** | **74.59** | **3.87** | **67.92** | 82.86 | 4.74 | **80.18** | 86.14 | 3.83 | 83.03 |
| CutMix [15] | 61.42 | 10.57 | 49.93 | 59.76 | 9.72 | 47.43 | 38.21 | 22.26 | 17.85 | 82.86 | 6.44 | 64.60 | 53.22 | 14.57 | 42.82 |
| FMix [41] | 51.29 | 14.03 | 38.72 | 53.01 | 10.26 | 38.52 | 36.25 | 21.14 | 18.22 | 68.29 | 7.24 | 53.18 | 46.27 | 16.91 | 31.57 |
| GridMix [20] | 42.04 | 17.22 | 30.18 | 42.95 | 15.34 | 30.42 | 28.26 | 24.25 | 14.40 | 55.69 | 9.69 | 46.04 | 41.77 | 17.55 | 26.66 |
| ResizeMix [40] | 52.88 | 11.04 | 37.87 | 57.69 | 9.35 | 44.99 | 33.50 | 21.24 | 16.12 | 71.21 | 5.30 | 57.05 | 47.59 | 12.60 | 29.88 |
| SaliencyMix [22] | 62.36 | 9.38 | 52.24 | 60.67 | 10.47 | 50.27 | 37.26 | 23.21 | 16.60 | 77.20 | 4.21 | 65.41 | 59.21 | 12.19 | 48.07 |
| PuzzleMix [23] | **87.20** | **3.36** | 82.38 | **88.35** | **2.99** | 84.15 | 72.19 | 4.13 | 58.71 | **84.42** | **3.83** | 79.23 | **87.36** | **3.25** | **83.81** |
| GuidedMixup [12] | 29.35 | 18.50 | 24.09 | 40.72 | 17.95 | 22.12 | 16.52 | 24.06 | 15.65 | 48.71 | 16.26 | 36.72 | 28.66 | 13.37 | 30.49 |
| StarMixup [9] | 85.98 | 3.55 | 82.76 | 86.89 | 3.62 | 85.06 | 69.38 | 4.81 | 48.41 | 81.71 | 4.88 | 77.41 | 84.93 | 3.42 | 80.18 |
| LabelSmooth [24] | **61.31** | **7.99** | **47.46** | **54.13** | **9.62** | **38.28** | 42.48 | **15.89** | **22.97** | **93.97** | **0.98** | 88.14 | 45.90 | 12.66 | 28.25 |
| OnlineLabelSmooth [26] | 46.99 | 12.16 | 26.69 | 45.12 | 13.55 | 28.01 | 31.00 | 20.46 | 13.62 | 81.71 | 4.00 | 56.47 | 42.01 | 16.23 | 23.07 |
| ConfidencePenalty [25] | 44.99 | 13.59 | 25.27 | 45.49 | 12.94 | 30.08 | 30.79 | 20.59 | 14.40 | 68.12 | 5.45 | 51.29 | 42.01 | 14.93 | 21.75 |
| DirichletLabelSmooth [27] | 59.38 | 9.31 | 44.58 | 51.22 | 10.61 | 35.06 | **43.60** | 16.90 | 22.49 | 93.60 | 1.06 | **88.45** | **52.64** | **10.94** | **33.71** |
| Bootstrapping [42] | 28.36 | 20.96 | 14.70 | 25.10 | 18.60 | 15.65 | 17.52 | 22.70 | 12.40 | 68.12 | 5.76 | 51.76 | 26.54 | 19.10 | 17.14 |

from scratch without relying on external pre-trained weights. All benchmarking experiments are conducted on a workstation equipped with a single NVIDIA A100 GPU.

**Training Settings:** For ResNet18, MobileNetv2, and StarLKNet-S, we employ the SGD [53] optimizer with an initial learning rate of 0.01. For FVRASNet, AMPVNet, ViT-S, and Swin-T, we utilize the AdamW [54] optimizer with an initial learning rate of 0.001 and a weight decay rate of 0.01. Across all configurations, the learning rate is dynamically decayed using a cosine annealing schedule, gradually decreasing to a minimum learning rate of 0 by the end of the training process. All models are trained for 600 epochs with a batch size of 32.

### *B. Results of Vein Recognition*

**Accuracy:** Across the five evaluated datasets, multi-image augmentation methods (MixUp, PuzzleMix, StarMixup) consistently dominate Top-1 Accuracy, particularly on palm vein datasets. For instance, as shown in Table I, MixUp and PuzzleMix on VERA220 using ResNet18 achieve 95.27% and 95.55% respectively, yielding substantial gains over the Vanilla baseline (71.45%). However, augmentation efficacy exhibits strong modality dependence. On the finger vein dataset SDUMLA-HMT (Table V), MixUp degrades ResNet18 accuracy to 79.87% (vs. 84.51% baseline), whereas automated policies like RandAugment excel (89.70%). Conversely, basic single-image geometric transformations (Flip, Rotate, Translate) frequently degrade performance below the baseline across most datasets, confirming that aggressive spatial operations easily disrupt the delicate topological structure of vascular patterns.

**EER and TAR@FAR=0.0001:** Strict biometric verification metrics reveal even more pronounced performance gaps. Mixup augmentations drastically reduce the EER and elevate the True Acceptance Rate (TAR). On SCUT1100 (Table III), MixUp reduces the ResNet18 EER from 0.30% to 0.07% and boosts TAR@FAR=0.0001 from 97.30% to 99.63%, highlighting its superiority in refining inter-class boundaries even on already saturated large-scale datasets. Similarly, on VERA220 (Table I), MixUp boosts TAR from 51.00% to 92.27%. These results demonstrate that rigorous threshold-dependent metrics like TAR@FAR underscore the necessity of advanced mixing strategies for high-security real-world deployments. Figure 5 shows the ROC curves of 5 vein datasets using the ResNet18 model, and the results are shown in Appendix VI.

### *C. Results of Robustness*

Beyond recognition performance, we evaluate robustness across four complementary dimensions: calibration (Table XI), corruption (Table XIII, Table XIV), adversarial attacking (Table VIII, Table XII), and occlusion classification.

**Calibration:** Following the experiments of a lot of mixup methods [10], [11], [55], [56], we evaluate some well-performing vein classifiers with the calibration experiments, which aim to identify the consistency between the predictive confidence and the actual empirical accuracy of models.

TABLE V
TOP-1 ACCURACY (%)↑, EER (%)↓, AND TAR@FAR (T@F)=0.0001 (%)↑ OF VARIOUS AUGMENTATIONS ACROSS DIFFERENT MODELS ON SDUMLA-HMT.

| SDUMLA-HMT | R18 | | | Mobv2 | | | FVN | | | APN | | | SLK-S | | | ViT-S | | | Swin-T | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F | Acc | EER | T@F |
| Vanilla | 84.51 | 1.90 | 80.03 | 82.94 | 2.20 | 78.77 | 86.71 | 1.49 | 83.96 | 93.55 | 0.56 | 93.79 | 82.15 | 1.80 | 77.52 | 72.17 | 7.49 | 66.12 | 82.39 | 2.37 | 76.49 |
| Flip | 79.09 | 2.56 | 74.84 | 77.52 | 2.60 | 70.83 | 84.59 | 1.64 | 80.58 | 87.89 | 1.42 | 84.67 | 81.84 | 2.10 | 76.42 | 66.90 | 9.46 | 58.41 | 80.11 | 3.07 | 74.61 |
| Rotate | 91.19 | 1.00 | 90.25 | 90.88 | 0.87 | 90.17 | 84.98 | 2.05 | 82.08 | 93.40 | 0.64 | 93.47 | 92.69 | 0.94 | 92.53 | 83.49 | 2.47 | 80.58 | 90.33 | 0.95 | 89.23 |
| Translation | 90.17 | 0.85 | 89.23 | 89.94 | 0.96 | 88.68 | 88.44 | 1.25 | 86.32 | 94.10 | 0.55 | 94.18 | 92.69 | 0.60 | 91.90 | 86.16 | 1.98 | 82.47 | 91.75 | 0.85 | 90.64 |
| Noise | 80.82 | 2.38 | 77.20 | 82.15 | 2.19 | 77.04 | 85.93 | 1.66 | 82.47 | 91.59 | 0.77 | 91.04 | 91.43 | 0.95 | 91.04 | 69.89 | 6.31 | 63.68 | 82.78 | 2.99 | 77.28 |
| Cutout [7] | 89.39 | 0.87 | 87.97 | 89.70 | **0.60** | 88.44 | 90.72 | 1.16 | 89.78 | 93.95 | 0.58 | 93.79 | 90.09 | 0.70 | 89.31 | 74.69 | 3.84 | 68.08 | 83.96 | 1.63 | 79.48 |
| GridMask [36] | 86.64 | 1.30 | 83.02 | 89.78 | 0.85 | 88.52 | 87.66 | 1.51 | 84.59 | 92.69 | 0.70 | 92.53 | 90.33 | 0.72 | 88.60 | 79.48 | 2.16 | 72.56 | 81.45 | 2.99 | 76.18 |
| RandomErasing [18] | 85.61 | 1.56 | 82.39 | 87.81 | 1.27 | 85.93 | 87.50 | 1.24 | 85.06 | 94.10 | 0.64 | 94.42 | 92.45 | 0.79 | 92.53 | 72.25 | 4.78 | 64.07 | 84.83 | 1.65 | 81.13 |
| RandomQuant [39] | 77.83 | 3.00 | 71.62 | 75.00 | 3.22 | 69.81 | 82.55 | 1.58 | 78.62 | 89.86 | 1.40 | 88.44 | 88.52 | 1.33 | 85.46 | 59.98 | 8.49 | 54.17 | 75.94 | 3.13 | 70.52 |
| AutoAugment [5] | 79.09 | 2.77 | 74.61 | 90.25 | 0.83 | 89.15 | 92.06 | 0.56 | 91.51 | 96.62 | 0.31 | 97.33 | 87.34 | 0.93 | 83.18 | 81.53 | 1.66 | 75.71 | 87.26 | 0.79 | 84.43 |
| RandAugment [6] | 89.70 | 0.71 | 88.84 | 90.17 | 0.86 | 88.36 | **94.18** | **0.32** | **94.97** | 96.38 | 0.24 | 97.33 | 94.18 | 0.56 | 94.97 | 87.81 | 1.09 | 85.53 | 93.87 | 0.47 | 94.10 |
| KeepAugment [19] | 84.28 | 2.05 | 79.32 | 81.53 | 1.90 | 77.75 | 86.64 | 1.41 | 83.02 | 92.37 | 0.93 | 91.59 | 88.44 | 1.06 | 85.06 | 75.24 | 6.28 | 69.81 | 82.08 | 2.18 | 77.52 |
| TrivialAugment [13] | 89.78 | 1.00 | 87.89 | **93.95** | 0.70 | **94.18** | 93.63 | 0.41 | 93.71 | **97.56** | **0.22** | **98.58** | **95.36** | **0.39** | **95.75** | 87.89 | 0.88 | 85.22 | **94.03** | 0.63 | **94.42** |
| TeachAugment [14] | 32.00 | 11.71 | 21.46 | 38.29 | 7.06 | 22.64 | – | – | – | 53.30 | 5.98 | 46.30 | – | – | – | **93.55** | **0.48** | **93.47** | 93.63 | **0.32** | 94.18 |
| SoftAugment [38] | 91.19 | 0.72 | 90.80 | 92.37 | 0.77 | 92.53 | 87.50 | 1.34 | 85.46 | 95.13 | 0.54 | 95.44 | 92.22 | 0.79 | 92.14 | 87.81 | 2.27 | 85.06 | 90.09 | 1.01 | 88.99 |
| YOCO [37] | **92.92** | **0.39** | **92.61** | 91.27 | 1.18 | 76.73 | 91.43 | 1.03 | 87.19 | 95.99 | 0.38 | 95.44 | 90.49 | 1.28 | 75.31 | 89.62 | 1.30 | 87.97 | 92.14 | 0.68 | 92.22 |
| RICAP [21] | **98.66** | **0.10** | **99.53** | **97.56** | **0.11** | **97.88** | 92.77 | 0.81 | 92.06 | 91.51 | 1.18 | 90.72 | – | – | – | **94.26** | **0.70** | **93.71** | **95.91** | **0.40** | **96.70** |
| MixUp [8] | 79.87 | 2.25 | 74.29 | 82.55 | 1.90 | 77.99 | 93.40 | 0.63 | **93.24** | 91.43 | 0.94 | 91.27 | 87.81 | 1.33 | 86.08 | 77.59 | 2.98 | 72.17 | 69.97 | 5.03 | 64.07 |
| CutMix [15] | 93.47 | 0.55 | 91.67 | 92.06 | 0.66 | 92.14 | 91.67 | 0.69 | 91.12 | 95.05 | 0.63 | 94.81 | 93.55 | 0.41 | 92.92 | 84.67 | 1.87 | 78.85 | 88.68 | 1.22 | 86.64 |
| FMix [41] | 90.72 | 1.10 | 87.74 | 88.29 | 1.48 | 85.93 | 89.39 | 1.18 | 87.03 | 94.50 | 0.72 | 93.00 | 90.49 | 0.94 | 88.29 | 79.32 | 2.44 | 74.06 | 85.77 | 1.97 | 80.74 |
| GridMix [20] | 90.02 | 1.12 | 88.68 | 92.61 | 0.86 | 91.59 | 91.90 | 0.88 | 90.72 | 93.95 | 0.85 | 93.71 | 92.85 | 0.41 | 92.77 | 86.87 | 1.72 | 84.04 | 89.15 | 1.41 | 87.74 |
| ResizeMix [40] | 95.83 | 0.23 | 96.70 | 93.87 | 0.41 | 93.79 | 90.72 | 0.80 | 89.86 | **95.75** | 0.48 | **96.46** | **94.65** | **0.31** | **95.28** | 87.19 | 1.64 | 84.98 | 91.75 | 1.10 | 89.70 |
| SaliencyMix [22] | 90.09 | 0.77 | 88.92 | 89.07 | 0.83 | 87.11 | 91.59 | **0.57** | 89.94 | 93.00 | 0.48 | 92.85 | 91.12 | 1.00 | 89.94 | 80.35 | 2.23 | 75.47 | 86.56 | 1.41 | 83.49 |
| PuzzleMix [23] | 88.29 | 1.22 | 85.93 | 93.32 | 0.68 | 91.67 | **93.63** | 0.64 | 93.00 | 94.34 | 0.71 | 93.32 | 93.63 | 0.55 | 92.37 | 84.67 | 1.80 | 79.01 | 87.19 | 1.72 | 84.28 |
| GuidedMixup [12] | 88.29 | 0.88 | 86.56 | 89.31 | 0.85 | 87.81 | 92.85 | 0.60 | 92.14 | 91.19 | 0.95 | 90.88 | 89.78 | 0.76 | 88.76 | 87.42 | 1.32 | 84.98 | 87.81 | 0.91 | 85.77 |
| StarMixup [9] | 79.40 | 1.96 | 75.71 | 81.76 | 1.96 | 77.44 | 91.67 | 0.70 | 91.12 | 89.54 | 1.25 | 88.60 | 84.36 | 2.11 | 80.03 | 78.14 | 2.90 | 73.66 | 67.69 | 5.73 | 61.40 |
| LabelSmoothing [24] | **87.03** | 1.79 | **84.51** | 84.83 | **1.67** | 80.90 | 89.78 | 1.02 | 87.03 | 97.09 | 0.25 | **97.72** | 87.11 | **1.01** | 84.28 | 73.98 | 5.81 | 66.75 | **84.12** | 2.44 | **78.14** |
| OnlineLabelSmooth [26] | 84.75 | **1.72** | 81.76 | 81.13 | 2.44 | 76.57 | 87.26 | 1.25 | 84.28 | 93.40 | 0.46 | 93.08 | 84.36 | 1.88 | 80.50 | 74.14 | 7.24 | 68.24 | 82.70 | **1.63** | 77.36 |
| ConfidencePenalty [25] | 83.57 | 2.05 | 79.25 | 78.22 | 2.65 | 71.93 | 87.03 | 1.33 | 84.12 | 93.87 | 0.91 | 94.10 | **87.89** | 1.58 | **85.14** | 71.54 | 5.96 | 64.70 | 80.50 | 3.28 | 76.57 |
| DirichletLabelSmooth [27] | 86.56 | 1.97 | 83.96 | **85.53** | 1.80 | **81.13** | **90.80** | **0.93** | **87.74** | **97.25** | **0.24** | 97.25 | **87.89** | 1.71 | 84.67 | 73.19 | 5.98 | 65.96 | 82.31 | 2.51 | 78.07 |
| Bootstrapping [42] | 82.31 | 2.13 | 77.04 | 72.64 | 4.03 | 64.86 | 78.46 | 2.35 | 62.74 | 93.95 | 0.54 | 93.95 | – | – | – | **74.69** | **5.74** | **69.50** | 82.39 | 2.66 | 76.97 |

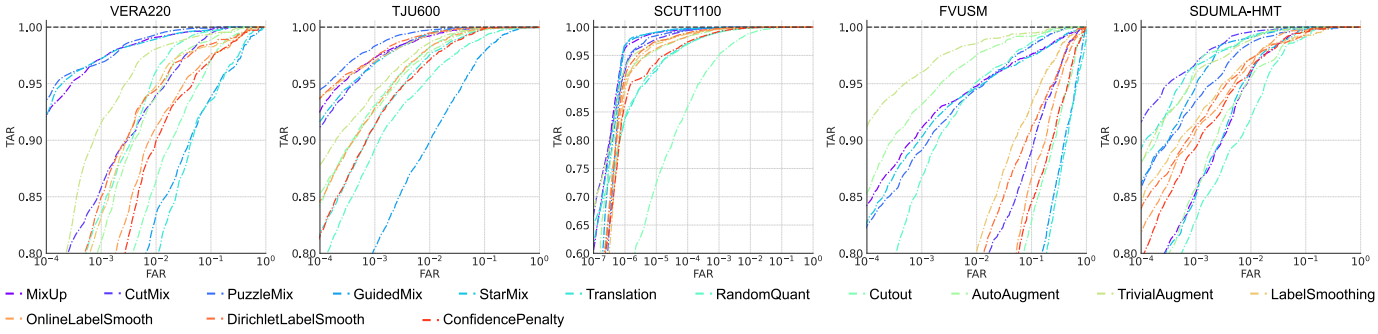


Fig. 5. Receiver Operating Characteristic (ROC) curves of various data augmentation methods across five vein datasets using the ResNet18 backbone.

Figure. 6 shows that compared with multi-image augmentations, single-image augmentations could achieve a lower ECE score than a vanilla classifier. AutoAugment and KeepAugment achieve 1.9% and 2.3%, respectively. However, mixup-based methods are all worse than vanilla. Lots of them show an overconfident state. The full results were reported in Table XI.

While mixup has been reported to improve calibration in general object recognition by smoothing label distributions, its impact on the fine-grained topological features of vein images remains to be explored. Our experiments aim to reveal whether these augmentations help the vein classifier produce more "honest" confidence scores, thereby enhancing the reliability of the system in high-security, real-world deployment scenarios.

**Corruption:** To further evaluate the robustness, we conduct corruption experiments following the setting of AugMix [49]. Table VI and Table VII show the C1 level results that mixup augmentations better than the single augmentations and label enhancement methods on both 2 datasets. Table XIII and Table XIV confirm that all model augmentation combinations suffer severe accuracy degradation as corruption severity escalates from C1 to C3, with vanilla models collapsing to below 30% on TJU600 at C3. Mixup-based methods provide the most consistent robustness, e.g., MixUp and PuzzleMix achieve 85.51%, 75.05%, 57.19% and 87.05%, 77.10%, 58.75% at C1, C2, and C3 on VERA220 under ResNet18, substantially above the Vanilla baseline of 69.59%, 61.99%, 46.70%. Policy-based methods, particularly TrivialAugment, achieve 83.98%, 73.18%, 55.45%, which represent the strongest single-image option for corruption robustness, while basic geometric augmentations consistently degrade below baseline even at mild corruption

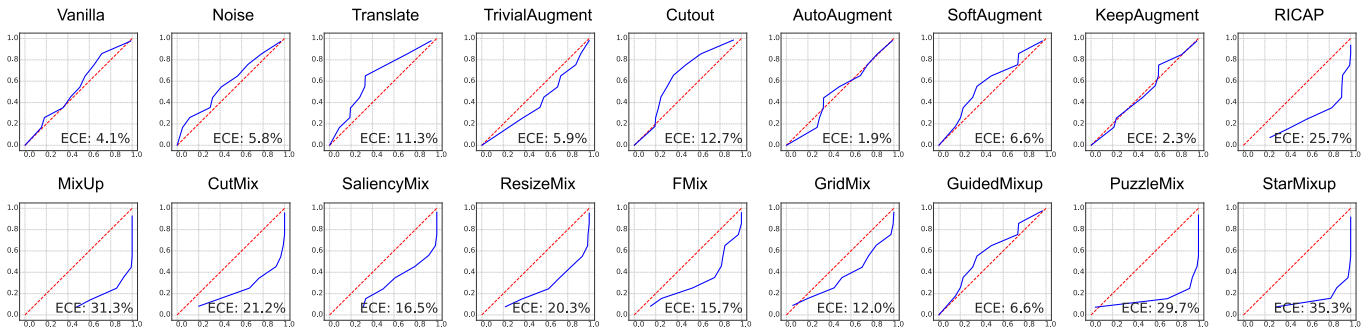


Fig. 6. The confidence plots of different augmentations on the VERA220 dataset using ResNet18. The red line indicates the expected prediction tendency.

TABLE VI
TOP-1 ACCURACY (%) ↑ OF CORRUPTION OF VARIOUS AUGMENTATIONS ACROSS DIFFERENT MODELS ON VEAR220-C1 DATASET.

| **Corruption** | R18 | Mobv2 | FVN | APN | SLK-S |
|---|---|---|---|---|---|
| Vanilla | 69.59 | 69.19 | 39.09 | 73.18 | 65.10 |
| Flip | 63.49 | 63.35 | 30.21 | 65.29 | 60.29 |
| Rotate | 62.75 | 59.66 | 32.18 | 62.58 | 58.33 |
| Translation | 60.67 | 64.14 | 37.06 | 63.54 | 60.93 |
| Noise | 67.32 | 68.49 | 37.18 | 69.02 | 66.00 |
| Cutout [7] | 68.06 | 64.88 | 33.95 | 68.28 | 62.82 |
| GridMask [36] | 64.93 | 66.53 | 24.81 | 54.12 | 54.71 |
| RandomErasing [18] | 61.22 | 66.89 | 36.36 | 72.54 | 65.86 |
| RandomQuant [39] | 79.59 | **81.27** | 22.27 | 80.96 | 77.27 |
| AutoAugment [5] | 75.43 | 71.10 | 35.12 | 76.56 | 72.82 |
| RandAugment [6] | 69.43 | 78.50 | 21.72 | 75.60 | **82.47** |
| KeepAugment [19] | 70.52 | 69.21 | 41.97 | 71.49 | 63.63 |
| TrivialAugment [13] | **83.98** | 78.78 | 41.33 | **85.29** | 81.99 |
| TeachAugment [14] | 68.06 | 57.19 | 38.16 | 56.84 | 65.31 |
| SoftAugment [38] | 64.94 | 68.21 | **44.97** | 73.91 | 66.81 |
| YOCO [37] | 62.06 | 64.95 | 29.35 | 71.74 | 58.83 |
| RICAP [21] | 73.15 | 67.37 | 28.27 | 67.72 | 61.02 |
| MixUp [8] | 85.51 | **89.54** | **66.89** | 88.75 | **85.81** |
| CutMix [15] | 76.31 | 88.08 | 35.65 | 77.01 | 70.47 |
| FMix [41] | 75.76 | 73.87 | 41.74 | 75.32 | 66.60 |
| GridMix [20] | 72.63 | 69.17 | 28.52 | 67.33 | 62.59 |
| ResizeMix [40] | 77.06 | 71.55 | 34.02 | 77.72 | 64.21 |
| SaliencyMix [22] | 77.42 | 75.05 | 42.47 | 78.55 | 68.96 |
| PuzzleMix [23] | **87.05** | 88.47 | 60.10 | 84.77 | 84.97 |
| GuidedMixup [12] | 61.85 | 63.15 | 22.61 | 54.85 | 61.98 |
| StarMixup [9] | 87.03 | 84.81 | 66.49 | **89.47** | 84.66 |
| LabelSmoothing [24] | 76.51 | 70.48 | **51.84** | 83.18 | 66.38 |
| OnlineLabelSmooth [26] | 70.83 | 67.14 | 48.56 | 76.01 | **68.23** |
| ConfidencePenalty [25] | 66.52 | 65.17 | 45.54 | 72.54 | 60.91 |
| DirichletLabelSmooth [27] | **77.53** | **71.53** | 51.48 | **84.94** | 64.24 |
| Bootstrapping [42] | 61.85 | 50.15 | 48.06 | 71.49 | 38.16 |

TABLE VII
TOP-1 ACCURACY (%) ↑ OF CORRUPTION OF VARIOUS AUGMENTATIONS ACROSS DIFFERENT MODELS ON TJU600-C1 DATASET.

| **Corruption** | R18 | Mobv2 | FVN | APN | SLK-S | ViT-S | Swin-T |
|---|---|---|---|---|---|---|---|
| Vanilla | 76.99 | 77.64 | 45.82 | 70.11 | 60.90 | 57.05 | 63.47 |
| Flip | 67.48 | 71.81 | 39.58 | 65.62 | 52.43 | 52.26 | 59.35 |
| Rotate | 72.36 | 73.76 | 39.74 | 64.62 | 55.62 | 58.75 | 64.13 |
| Translation | 76.57 | 76.25 | 44.71 | 68.70 | 61.87 | 59.65 | 62.51 |
| Noise | 79.48 | 78.58 | 47.59 | 63.63 | 76.83 | 57.52 | 64.32 |
| Cutout [7] | 75.82 | 74.17 | 46.94 | 64.47 | 68.08 | 58.92 | 64.38 |
| GridMask [36] | 71.11 | 73.86 | 39.40 | 55.95 | 70.14 | 57.47 | 63.33 |
| RandomErasing [18] | 79.53 | **80.35** | 47.08 | **75.83** | 67.57 | 58.20 | **66.89** |
| RandomQuant [39] | 69.27 | 76.97 | 47.66 | 67.39 | 72.37 | **65.68** | 47.00 |
| AutoAugment [5] | 77.11 | 76.68 | 42.08 | 70.22 | **77.72** | 52.97 | 64.53 |
| RandAugment [6] | 80.89 | 77.96 | 45.45 | 73.39 | 69.45 | 65.25 | 65.64 |
| KeepAugment [19] | 76.08 | 79.27 | 44.26 | 70.83 | 74.81 | 56.97 | 64.33 |
| TrivialAugment [13] | **82.67** | 79.61 | **48.62** | 73.22 | 74.72 | 64.40 | 66.57 |
| TeachAugment [14] | 71.21 | 62.24 | 33.52 | 49.28 | 58.22 | 64.16 | 62.38 |
| SoftAugment [38] | 77.96 | 78.23 | 46.75 | 71.33 | 71.01 | 56.73 | 65.58 |
| YOCO [37] | 63.75 | 67.72 | 39.80 | 68.13 | 65.21 | 54.05 | 51.64 |
| RICAP [21] | 74.39 | 71.76 | 48.40 | 53.07 | 77.04 | 46.15 | 48.43 |
| MixUp [8] | 88.72 | 88.63 | 61.52 | 75.82 | 89.97 | **75.80** | 80.19 |
| CutMix [15] | 83.12 | 83.50 | 54.30 | 68.67 | 83.01 | 61.20 | 63.47 |
| FMix [41] | 82.50 | 79.51 | 53.00 | 67.92 | 78.58 | 59.09 | 69.64 |
| GridMix [20] | 78.48 | 79.46 | 30.89 | 63.98 | 79.89 | 60.21 | 66.17 |
| ResizeMix [40] | 78.16 | 76.65 | 48.23 | 66.78 | 79.91 | 57.56 | 61.24 |
| SaliencyMix [22] | 82.61 | 82.52 | 54.02 | 71.92 | 83.33 | 62.70 | 63.88 |
| PuzzleMix [23] | **89.49** | 88.92 | **63.74** | 75.04 | 89.51 | 74.24 | 73.81 |
| GuidedMixup [12] | 63.44 | 60.42 | 23.50 | 54.54 | 63.09 | 56.00 | 40.75 |
| StarMixup [9] | 88.64 | **89.99** | 62.22 | **78.95** | **91.39** | 73.73 | **80.34** |
| LabelSmoothing [24] | 89.53 | 88.47 | **59.15** | **82.44** | 86.39 | **60.95** | 76.80 |
| OnlineLabelSmooth [26] | 79.33 | 80.50 | 47.70 | 75.32 | 81.60 | 59.62 | 69.21 |
| ConfidencePenalty [25] | 77.00 | 77.98 | 45.66 | 73.55 | 74.36 | 58.12 | 64.06 |
| DirichletLabelSmooth [27] | **89.73** | **88.95** | 58.38 | 81.26 | **87.47** | 60.81 | **77.56** |
| Bootstrapping [42] | 61.58 | 50.43 | 19.64 | 72.39 | 55.50 | 57.64 | 64.58 |

levels. Label enhancement methods show moderate corruption stability, with LabelSmoothing reaching 89.53%, 72.22%, 40.13% on TJU600 under ResNet18.

**Adversarial Attack:** To further explore adversarial attack resilience, we evaluate the Fast Gradient Sign Method (FGSM) and Projected Gradient Descent (PGD). FGSM uses an $\ell_\infty$ constraint with $\epsilon = 0.2/255$. PGD is configured with $\epsilon = 0.2/255$, $\alpha = 0.05/255$, 10 iterations and the $\ell_\infty$ norm.

As shown in Table VIII and Table XII, our results indicate a striking dissociation between clean accuracy and adversarial robustness. Label enhancement methods, LabelSmoothing and DirichletLabelSmooth, exhibit the strongest adversarial robustness, with LabelSmoothing achieving 75.18% and 70.37% by FGSM and PGD on TJU600 under ResNet18, nearly doubling the Vanilla baseline of 40.93% and 30.45%. Conversely, mixup-based methods suffer counterintuitive adversarial fragility: MixUp drops to 18.80% and 4.87% on TJU600 under ResNet18, well below Vanilla, because soft-label training blurs inter-class boundaries and expands the gradient-accessible attack surface.

**Occlusion Classification:** To evaluate the robustness of augmentation methods against spatially continuous information loss, we conduct an occlusion classification experiment on the VERA220 and TJU600 datasets using ResNet18, by randomly masking square regions with increasing ratios (0% to 50%). As shown in Figure 7, cutting-based methods, including Cutout, CutMix, and PuzzleMix, maintain stable performance, while other methods degrade rapidly.

TABLE VIII
TOP-1 ACCURACY (%) ↑ OF VARIOUS AUGMENTATIONS ACROSS DIFFERENT MODELS UNDER FGSM AND PGD ATTACKS ON TJU600 DATASET.

| TJU600 | R18 | | Mobv2 | | FVN | | APN | | SLK-S | | ViT-S | | Swin-T | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | FGSM | PGD | FGSM | PGD | FGSM | PGD | FGSM | PGD | FGSM | PGD | FGSM | PGD | FGSM | PGD |
| Vanilla | 40.93 | 30.45 | 47.27 | 39.68 | 0.23 | 0.00 | 22.73 | 15.93 | 13.82 | 6.73 | 2.75 | 0.48 | 0.08 | 0.00 |
| Flip | 19.42 | 9.85 | 32.30 | 22.65 | 0.07 | 0.00 | 17.85 | 11.30 | 6.80 | 1.08 | 2.30 | 0.58 | 0.08 | 0.00 |
| Rotate | 30.22 | 18.77 | 28.27 | 15.47 | 0.12 | 0.00 | 2.35 | 0.30 | 2.47 | 0.17 | 2.40 | 0.38 | 0.03 | 0.00 |
| Translation | 40.25 | 30.23 | 28.63 | 12.98 | 0.17 | 0.00 | 4.38 | 0.32 | 9.95 | 3.57 | 1.52 | 0.20 | 0.02 | 0.00 |
| Noise | **53.98** | **49.55** | **53.90** | **48.07** | **0.40** | **0.07** | 16.73 | 10.47 | **54.02** | **49.52** | **14.63** | **9.77** | **8.52** | **4.23** |
| Cutout [7] | 32.80 | 16.05 | 28.67 | 14.42 | 0.17 | 0.02 | 17.65 | 9.03 | 25.82 | 10.58 | 4.58 | 1.25 | 0.05 | 0.00 |
| GridMask [36] | 45.83 | 37.10 | 49.73 | 43.23 | 0.12 | 0.00 | 15.15 | 9.58 | 39.35 | 18.90 | 8.07 | 3.47 | 2.75 | 0.30 |
| RandomErasing [18] | 50.07 | 40.82 | 46.17 | 36.22 | 0.25 | 0.03 | **31.20** | **24.05** | 25.52 | 13.77 | 3.17 | 0.55 | 0.13 | 0.00 |
| RandomQuant [39] | 30.53 | 16.60 | 46.50 | 34.92 | 0.03 | 0.00 | 27.45 | 20.73 | 38.95 | 15.92 | 13.68 | 6.80 | 0.58 | 0.07 |
| AutoAugment [5] | 37.23 | 22.45 | 25.48 | 6.43 | 0.00 | 0.00 | 22.53 | 12.70 | 42.93 | 23.28 | 0.67 | 0.05 | 0.02 | 0.00 |
| RandAugment [6] | 49.43 | 38.32 | 22.10 | 6.98 | 0.00 | 0.00 | 23.67 | 14.08 | 11.27 | 1.37 | 3.20 | 0.58 | 0.07 | 0.00 |
| KeepAugment [19] | 37.48 | 26.60 | 45.75 | 36.47 | 0.17 | 0.00 | 20.97 | 14.00 | 43.08 | 31.45 | 2.75 | 0.75 | 0.05 | 0.00 |
| TrivialAugment [13] | 53.20 | 42.12 | 25.48 | 4.55 | 0.08 | 0.00 | 15.55 | 5.32 | 18.93 | 5.57 | 1.30 | 0.08 | 0.02 | 0.00 |
| TeachAugment [14] | 1.50 | 0.00 | 0.15 | 0.02 | 0.00 | 0.00 | 6.93 | 0.00 | 3.80 | 0.87 | 1.92 | 0.32 | 0.00 | 0.00 |
| SoftAugment [38] | 46.68 | 37.65 | 40.90 | 30.43 | 0.32 | 0.00 | 28.12 | 20.43 | 37.38 | 24.57 | 2.27 | 0.42 | 0.05 | 0.00 |
| YOCO [37] | 13.47 | 4.85 | 4.40 | 0.05 | 0.05 | 0.02 | 20.00 | 8.50 | 12.50 | 1.25 | 2.85 | 0.50 | 0.02 | 0.00 |
| RICAP [21] | 39.70 | 16.43 | 31.08 | 6.70 | 0.57 | 0.02 | 8.00 | 4.05 | 30.80 | 9.95 | 0.95 | 0.20 | 0.05 | 0.00 |
| MixUp [8] | 18.80 | 4.87 | 34.13 | 12.33 | 0.48 | 0.12 | 12.40 | 3.45 | 24.82 | 7.22 | 15.47 | 5.23 | 1.07 | 0.02 |
| CutMix [15] | **51.53** | 28.13 | 46.02 | 17.73 | 0.60 | 0.13 | **22.90** | 13.08 | 41.27 | 14.37 | 6.23 | 0.95 | 0.27 | 0.00 |
| FMix [41] | 42.58 | 16.30 | 32.75 | 4.42 | 0.32 | 0.02 | 20.87 | 12.78 | 25.62 | 4.68 | 7.80 | 2.63 | 0.18 | 0.00 |
| GridMix [20] | 42.80 | 26.32 | 42.88 | 26.85 | 0.10 | 0.00 | 8.30 | 0.22 | 31.60 | 14.93 | 10.58 | 3.03 | 0.52 | 0.00 |
| ResizeMix [40] | 48.25 | 22.13 | 38.68 | 4.28 | 0.05 | 0.00 | 18.02 | 10.13 | 24.42 | 2.58 | 7.03 | 1.60 | 0.17 | 0.00 |
| SaliencyMix [22] | 51.47 | **29.93** | 42.47 | 15.02 | 0.57 | 0.03 | 22.88 | **14.67** | 45.65 | 19.90 | 10.17 | 2.83 | 0.43 | 0.00 |
| PuzzleMix [23] | 30.75 | 6.70 | 37.95 | 8.32 | 0.35 | 0.02 | 16.92 | 9.35 | 30.35 | 1.12 | 5.32 | 0.95 | 0.55 | 0.02 |
| GuidedMixup [12] | 2.95 | 0.00 | 0.47 | 0.00 | 0.00 | 0.00 | 10.38 | 4.38 | 0.92 | 0.00 | 1.08 | 0.10 | 0.00 | 0.00 |
| StarMixup [9] | 31.67 | 14.08 | **56.12** | **39.07** | **0.72** | **0.15** | 21.48 | 13.18 | **50.68** | **27.17** | **15.95** | **5.37** | **1.88** | **0.17** |
| LabelSmoothing [24] | **75.18** | **70.37** | **73.35** | **69.43** | **2.00** | **0.30** | **55.60** | **45.73** | 67.15 | 58.73 | 13.17 | 6.67 | **3.87** | **0.73** |
| OnlineLabelSmooth [26] | 46.18 | 35.68 | 52.78 | 45.37 | 0.25 | 0.02 | 26.65 | 18.08 | 55.30 | 47.13 | 4.52 | 1.48 | 0.23 | 0.00 |
| ConfidencePenalty [25] | 42.43 | 33.35 | 48.77 | 41.87 | 0.30 | 0.00 | 26.58 | 19.40 | 41.33 | 30.12 | 2.92 | 0.52 | 0.10 | 0.00 |
| DirichletLabelSmooth [27] | 74.30 | 67.98 | 70.55 | 65.67 | 1.73 | 0.25 | 52.48 | 42.63 | **70.80** | **63.48** | **13.28** | **7.00** | 3.52 | 0.63 |
| Bootstrapping [42] | 23.20 | 12.58 | 6.35 | 0.12 | 0.30 | 0.02 | 25.15 | 17.87 | 3.42 | 0.02 | 3.00 | 0.72 | 0.07 | 0.00 |

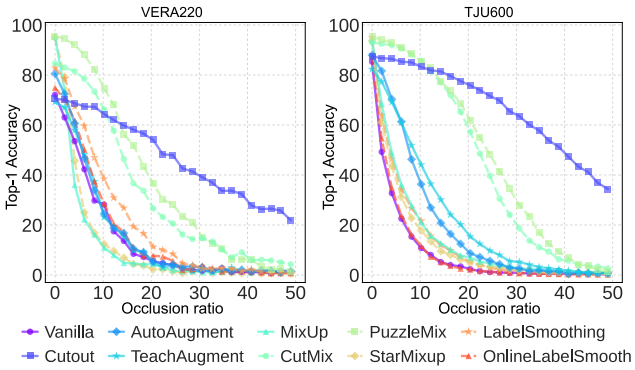


Fig. 7. Top-1 Accuracy under varying occlusion ratios (0% → 50%) on the VERA220 and TJU600 datasets using ResNet18.

## V. ANALYSIS AND DISCUSSIONS

### A. Orthogonality of Augmentations

We study whether augmentations from different categories interact constructively by composing single-image, multi-image, and label-enhancement methods on ResNet18 with VERA220 and TJU600, as shown in Table IX.

Cross-category compositions consistently improve performance, indicating that different augmentation families provide complementary benefits. On VERA220, AutoAugment + LabelSmoothing improves accuracy from 80.82% to 89.73% and TAR@FAR from 65.09% to 78.64%. Label regularization also remains useful at high accuracy: MixUp + LabelSmoothing increases accuracy from 95.27% to 97.18%, while PuzzleMix + LabelSmoothing reduces EER from 0.83% to 0.65%. The full three-level composition (AutoAugment + PuzzleMix + LabelSmoothing) achieves the best overall results, reaching 98.00%, 0.56%, and 95.27% on VERA220 and 96.50%, 0.45%, and 96.12% on TJU600.

In contrast, intra-category combinations deliver marginal improvements. On VERA220, AutoAugment + MixUp only slightly outperforms vanilla MixUp (97.00% *vs.* 95.27%), demonstrating that cross-category fusion works better.

### B. Efficiency Analysis

Our analysis is conducted on a lightweight backbone using the VERA220 dataset, with a standardized input resolution of $224 \times 224$ to ensure a fair comparison of overhead. Table X summarizes the efficiency–accuracy trade-off on MobileNetv2. Most methods keep the same inference cost as Vanilla (0.65 GFLOPs and 1.00M augmentation parameters), so their overhead mainly appears in training time and peak memory. MixUp offers the best balance, improving accuracy from 71.64% to 95.55% with nearly unchanged training time (5.04 *vs.* 5.02 s/epoch). StarMixup is similarly efficient, reaching 92.55% accuracy with near-Vanilla cost. PuzzleMix achieves the highest accuracy (95.91%) but requires more computation and training time. In contrast, TeachAugment and SoftAugment incur high overhead or lower-than-Vanilla accuracy, indicating that lightweight mixing or quantization-based strategies are more practical for deployment-oriented vein recognition.

TABLE IX
TOP-1 ACCURACY (%)↑, EER (%)↓, AND TAR@FAR=0.0001 (%)↑ OF VARIOUS COMPOSED AUGMENTATIONS ON VERA220 AND TJU600 DATASETS IN RESNET18.

| Methods | VERA220 | | | TJU600 | | |
|---|---|---|---|---|---|---|
| | Acc. | EER | TAR@FAR | Acc. | EER | TAR@FAR |
| Vanilla | 71.45 | 5.20 | 51.00 | 85.55 | 1.72 | 81.23 |
| ***Single Image Augmentation with Label Enhancement*** | | | | | | |
| AutoAugmentation | 80.82 | 2.55 | 65.09 | 88.28 | 1.59 | 85.23 |
| + LabelSmoothing | 89.73 | 2.44 | 78.64 | 94.97 | 0.77 | 93.83 |
| ***Multi Image Augmentation with Label Enhancement*** | | | | | | |
| MixUp | 95.27 | 0.91 | 92.27 | 93.90 | 0.84 | 92.51 |
| + LabelSmoothing | 97.18 | 0.63 | 96.36 | 96.37 | 0.51 | 95.33 |
| PuzzleMix | 95.55 | 0.83 | 93.36 | 95.25 | 0.46 | 94.45 |
| + LabelSmoothing | 97.27 | 0.65 | 96.09 | 96.58 | 0.38 | 96.05 |
| ***Composed Augmentations*** | | | | | | |
| AutoAugmentation | 80.82 | 2.55 | 65.09 | 88.28 | 1.59 | 85.23 |
| + MixUp | 97.00 | 0.70 | 92.73 | 95.55 | 0.65 | 93.73 |
| + LabelSmoothing | 97.55 | 0.44 | 94.00 | 95.83 | 0.65 | 94.48 |
| AutoAugmentation | 80.82 | 2.55 | 65.09 | 88.28 | 1.59 | 85.23 |
| + PuzzleMix | 97.18 | 0.80 | 94.55 | 96.25 | 0.67 | 95.17 |
| + LabelSmoothing | 98.00 | 0.56 | 95.27 | 96.50 | 0.45 | 96.12 |

TABLE X
GFLOPS, TOP-1 ACCURACY (%), TRAINING TIME (S/EPOCH), AUGMENTATION PARAMETERS (M), GPU MEMORY (MB), AND APEX RANK OF AUGMENTATIONS (MOBILENETV2 ON THE VEAR220 DATASET).

| Methods | GFLOPs | Acc. ↑ | $T_{\text{train}}$ ↓ | $P_{\text{aug}}$ | $M_{\text{peak}}$ ↓ | APEX Rank |
|---|---|---|---|---|---|---|
| Vanilla | 0.65 | 71.64 | 5.02 | 1.00 | 2507.10 | – |
| Flip | 0.65 | 67.32 | 5.03 | 1.00 | 2524.83 | 3 |
| Rotate | 0.65 | 63.05 | 5.28 | 1.00 | 2507.10 | 4 |
| Translation | 0.65 | 65.45 | 5.23 | 1.00 | 2507.10 | 3 |
| Noise | 0.65 | 69.73 | 5.04 | 1.00 | 2566.45 | 3 |
| Cutout [7] | 0.65 | 67.55 | 6.57 | 1.00 | 2524.83 | 4 |
| GridMask [36] | 0.65 | 68.50 | 6.82 | 1.00 | 2524.83 | 4 |
| RandomErasing [18] | 0.65 | 69.86 | 5.06 | 1.00 | 2488.38 | 2 |
| RandomQuant [39] | 0.65 | 82.23 | 5.13 | 1.00 | **2483.50** | **1** |
| AutoAugment [5] | 0.65 | 76.91 | 7.81 | 1.00 | 2507.72 | 4 |
| RandAugment [6] | 0.65 | 82.09 | 5.49 | 1.00 | 2485.71 | 2 |
| KeepAugment [19] | 1.31 | 70.91 | 5.45 | 1.00 | 2549.48 | 4 |
| TrivialAugment [13] | 0.65 | 82.45 | 5.19 | 1.00 | 2507.10 | 2 |
| TeachAugment [14] | 3.82 | 59.86 | 9.92 | 2.37 | 12156.80 | 5 |
| SoftAugment [38] | 0.65 | 69.82 | 11.79 | 1.00 | 2652.35 | 5 |
| YOCO [37] | 0.65 | 66.14 | 6.68 | 1.00 | 2506.12 | 3 |
| RICAP [21] | 0.65 | 70.82 | 5.09 | 1.00 | 2508.17 | 3 |
| MixUp [8] | 0.65 | 95.55 | 5.04 | 1.00 | 2545.46 | **1** |
| CutMix [15] | 0.65 | 77.41 | 5.05 | 1.00 | 2507.72 | **1** |
| FMix [41] | 0.65 | 75.55 | 5.05 | 1.00 | 2524.83 | 2 |
| GridMix [20] | 0.65 | 75.14 | 5.07 | 1.00 | 2524.83 | 3 |
| ResizeMix [40] | 0.65 | 75.86 | 5.04 | 1.00 | 2488.38 | **1** |
| SaliencyMix [22] | 0.65 | 79.14 | 5.24 | 1.00 | 2507.10 | 3 |
| PuzzleMix [23] | 1.31 | **95.91** | 8.66 | 1.00 | 2578.32 | **1** |
| GuidedMixup [12] | 1.31 | 69.68 | 7.33 | 1.00 | 2538.22 | 4 |
| StarMixup [9] | 0.65 | 92.55 | 5.09 | 1.00 | 2507.10 | **1** |
| LabelSmoothing [24] | 0.65 | 75.95 | 5.03 | 1.00 | 2547.46 | 2 |
| OnlineLabelSmooth [26] | 0.65 | 70.18 | 5.08 | 1.00 | 2566.63 | 4 |
| ConfidencePenalty [25] | 0.65 | 70.86 | **5.02** | 1.00 | 2507.72 | 2 |
| DirichletLabelSmooth [27] | 0.65 | 77.59 | 5.03 | 1.00 | 2524.83 | **1** |
| Bootstrapping [42] | 0.65 | 52.59 | 5.03 | 1.00 | 2507.72 | 3 |

## C. Discussion

***When Do Augmentations Help?*** Augmentation efficacy is strongly conditioned on dataset scale and evaluation granularity. In data-scarce regimes (e.g., the VERA220 dataset with 2,200 images and the FV-USM dataset with 1,476 images), augmentation yields the largest absolute gains in accuracy. On larger datasets (e.g., SCUT1100 dataset with 11,000 images), vanilla baselines are already strong. Yet, augmentation continues to produce meaningful improvements at the harder metrics: MixUp reduces EER from 0.30% to 0.07% and boosts TAR@FAR=0.0001 from 97.30% to 99.63% on SCUT1100, revealing improvements that are invisible to accuracy-centric analysis. This observation confirms that stringent criteria like TAR@FAR=0.0001 and EER are the most sensitive and operationally relevant metrics for evaluating biometric augmentation strategies.

***Limitations of Current Augmentation Methods.*** Three systematic limitations characterize current augmentation strategies in the vein domain. First, geometric augmentations (Flip, Rotate, Translation, YOCO) consistently degrade performance across all datasets and architectures, as arbitrary spatial transformations violate the topological integrity of vein patterns that is essential for identity discrimination. Second, the highest accuracy methods (mixup-based) are simultaneously the worst calibrated (ECE >30%) and the most adversarially fragile, e.g., MixUp drops to 4.87% under PGD on TJU600, which reveals a multi-dimensional performance and robustness trade-off that no currently available single method resolves. Third, augmentation benefits are dataset-dependent: on SDUMLA-HMT, MixUp marginally underperforms the Vanilla baseline (79.87% *vs.* 84.51% on ResNet18) while RICAP achieves the highest accuracy of 98.66%, indicating that dataset-specific structural characteristics modulate which augmentation family is optimal and that blanket application of a single strategy across all vein datasets remains inadvisable.

***Trade-off between Accuracy & Calibration.*** Strong recognition performance and robustness are rarely co-realized by the same augmentation method. MixUp dominates on Accuracy, EER, and TAR@FAR yet ranks among the worst in calibration. For example, ECE score >30% on TJU600 and adversarial robustness 4.87% under PGD; LabelSmoothing provides the strongest adversarial defense but is simultaneously the most miscalibrated (ECE score up to 47.88%). No single method currently offers an adequate all-in-one solution, and designing augmentations that jointly optimize across recognition accuracy and robustness dimensions remains an open research direction.

***Data-Agnostic Augmentation Methods and Cross-Domain Generalization.*** TrivialAugment and RandomQuant consistently rank among the top-performing single-image methods regardless of anatomical site (palm or finger vein) or imaging sensor, suggesting that their respective mechanisms, which include tuning-free operation sampling and quantization that induced spectral invariance, are genuinely domain-agnostic. In contrast, mixup-based methods are highly effective on palm vein datasets but exhibit degraded or reversed benefits on finger vein configurations (SDUMLA-HMT), confirming that no single augmentation strategy achieves universally optimal generalization across vein modalities. This finding motivates modality-aware augmentation selection and multi-method composition as standard practices in future vein recognition system design.

***Extensibility to Broader Biometric Modalities.*** Beyond vein recognition, the highly modular architecture of the AGVBench framework provides a clear pathway for extension to other biometric modalities, such as fingerprint, iris, or periocular recognition. Because our open-source codebase is deeply decoupled based on the MMCV ecosystem, researchers can seamlessly integrate new biometric tasks by simply customizing the data preprocessing pipelines (`.agvbench.datasets`), while fully reusing the comprehensive suite of standardized augmentation modules and evaluation protocols. This potential extensibility establishes AGVBench not merely as a vein-specific benchmark but as a unified, foundational testbed for exploring representation learning and data augmentation across the broader biometric security community.

## VI. Conclusion

We have presented AGVBench, a large-scale benchmark evaluating 30 augmentation strategies for vein recognition across five datasets, seven deep learning architectures, and six evaluation dimensions. Our study shows that multi-image augmentations consistently dominate recognition performance, while geometric augmentations are broadly detrimental. Critically, top-performing methods exhibit poor calibration and adversarial fragility, exposing an accuracy-robustness trade-off unresolved by any existing method. We hope AGVBench motivates future augmentation designs that jointly optimize recognition accuracy and real-world reliability.

## References


[1] Huafeng Qin, Yuming Fu, Jing Chen, Qun Song, Yantao Li, Mounim A. El-Yacoubi, and Dexing Zhong. Wtxgrn: Wavelet transform-based extended gated recurrent network for palm vein recognition. *IEEE Transactions on Information Forensics and Security*, 20:7911–7926, 2025.

[2] Dacan Luo, Yitao Qiao, Di Xie, Shifeng Zhang, and Wenxiong Kang. Palm vein recognition under unconstrained and weak-cooperative conditions. *IEEE Transactions on Information Forensics and Security*, 19:4601–4614, 2024.

[3] Huafeng Qin, Mounim A El-Yacoubi, Yantao Li, and Chongwen Liu. Multi-scale and multi-direction gan for cnn-based single palm-vein identification. *IEEE Transactions on Information Forensics and Security*, 16:2652–2666, 2021.

[4] Nurul Maisarah Kamaruddin and Bakhtiar Affendi Rosdi. A new filter generation method in pcanet for finger vein recognition. *IEEE Access*, 7:132966–132978, 2019.

[5] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 113–123, 2019.

[6] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 702–703, 2020.

[7] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

[8] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018.

[9] Xin Jin, Hongyu Zhu, Simon Fong, João Alexandre Lobo Marques, Huafeng Qin, and Yun Jiang. Starmixup: A more suitable mixup method for palm-vein identification. In *2025 7th International Symposium on Computational and Business Intelligence (ISCBI)*, pages 83–87. IEEE, 2025.

[10] Zicheng Liu, Siyuan Li, Di Wu, Zihan Liu, Zhiyuan Chen, Lirong Wu, and Stan Z Li. Automix: Unveiling the power of mixup for stronger classifiers. In *European Conference on Computer Vision*, pages 441–458. Springer, 2022.

[11] Huafeng Qin, Xin Jin, Yun Jiang, Mounîm El-Yacoubi, and Xinbo Gao. Adversarial automixup. In *The Twelfth International Conference on Learning Representations*, 2024.

[12] Minsoo Kang and Suhyun Kim. Guidedmixup: an efficient mixup strategy guided by saliency maps. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 1096–1104, 2023.

[13] Samuel G Müller and Frank Hutter. Trivialaugment: Tuning-free yet state-of-the-art data augmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 774–782, 2021.

[14] Teppei Suzuki. Teachaugment: Data augmentation optimization using teacher knowledge. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10904–10914, 2022.

[15] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *International Conference on Computer Vision (ICCV)*, pages 6023–6032, 2019.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[17] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021.

[18] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 13001–13008, 2020.

[19] Chengyue Gong, Dilin Wang, Meng Li, Vikas Chandra, and Qiang Liu. Keepaugment: A simple information-preserving data augmentation approach. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1055–1064, 2021.

[20] Kyungjune Baek, Duhyeon Bang, and Hyunjung Shim. Gridmix: Strong regularization through local context mapping. *Pattern Recognition*, 109:107594, 2021.

[21] Ryo Takahashi, Takashi Matsubara, and Kuniaki Uehara. Data augmentation using random image cropping and patching for deep cnns. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(9):2917–2931, 2019.

[22] AFM Shahab Uddin, Mst Sirazam Monira, Wheemyung Shin, TaeChoong Chung, and Sung-Ho Bae. Saliencymix: A saliency guided data augmentation strategy for better regularization. In *International Conference on Learning Representations*, 2020.

[23] Jang-Hyun Kim, Wonho Choo, and Hyun Oh Song. Puzzle mix: Exploiting saliency and local statistics for optimal mixup. In *International Conference on Machine Learning*, pages 5275–5285. PMLR, 2020.

[24] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[25] Gabriel Pereyra, George Tucker, Jan Chorowski, Łukasz Kaiser, and Geoffrey Hinton. Regularizing neural networks by penalizing confident output distributions. In *International Conference on Learning Representations*, 2017.

[26] Chang-Bin Zhang, Peng-Tao Jiang, Qibin Hou, Yunchao Wei, Qi Han, Zhen Li, and Ming-Ming Cheng. Delving deep into label smoothing. *IEEE Transactions on Image Processing*, 30:5984–5996, 2021.

[27] Minhao Cheng, Pin-Yu Chen, Sijia Liu, Shiyu Chang, Cho-Jui Hsieh, and Payel Das. Self-progressing robust training. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 7107–7115, 2021.

[28] So Ra Cho, Young Ho Park, Gi Pyo Nam, Kwang Youg Shin, Hyeon Chang Lee, Kang Ryoung Park, Sung Min Kim, and Ho Chul Kim. Enhancement of finger-vein image by vein line tracking and adaptive gabor filtering for finger-vein recognition. *Applied Mechanics and Materials*, 145:219–223, 2012.

[29] Wenxiong Kang and Qiuxia Wu. Contactless palm vein recognition using a mutual foreground-based local binary pattern. *IEEE transactions on Information Forensics and Security*, 9(11):1974–1985, 2014.

[30] Munalih Ahmad Syarif, Thian Song Ong, Andrew BJ Teoh, and Connie Tee. Enhanced maximum curvature descriptors for finger vein verification. *Multimedia Tools and Applications*, 76(5):6859–6887, 2017.

[31] Vijayakumar Ponnusamy, Abhijit Sridhar, Arun Baalaaji, and M Sangeetha. A palm vein recognition system based on a support

vector machine. *IEIE Transactions on Smart Processing & Computing*, 8(1):1–7, 2019.
[32] Shazeeda Shazeeda and Bakhtiar Affendi Rosdi. Finger vein recognition using mutual sparse representation classification. *IET biometrics*, 8(1):49–58, 2019.
[33] Weili Yang, Wei Luo, Wenxiong Kang, Zhixing Huang, and Qiuxia Wu. Fvras-net: An embedded finger-vein recognition and antispoofing system using a unified cnn. *IEEE Transactions on Instrumentation and Measurement*, 69(11):8690–8701, 2020.
[34] Jiaquan Shen, Ningzhong Liu, Chenglu Xu, Han Sun, Yushun Xiao, Deguang Li, and Yongxin Zhang. Finger vein recognition algorithm based on lightweight deep convolutional neural network. *IEEE Transactions on Instrumentation and Measurement*, 71:1–13, 2021.
[35] Dacan Luo, Junduan Huang, Weili Yang, M. Saad Shakeel, and Wenxiong Kang. Rsnet: Region-specific network for contactless palm vein authentication. *IEEE Transactions on Information Forensics and Security*, 20:2734–2747, 2025.
[36] Pengguang Chen, Shu Liu, Hengshuang Zhao, Xingquan Wang, and Jiaya Jia. Gridmask data augmentation. *arXiv preprint arXiv:2001.04086*, 2020.
[37] Junlin Han, Pengfei Fang, Weihao Li, Jie Hong, Mohammad Ali Armin, Ian Reid, Lars Petersson, and Hongdong Li. You only cut once: Boosting data augmentation with a single cut. In *International Conference on Machine Learning*, pages 8196–8212. PMLR, 2022.
[38] Yang Liu, Shen Yan, Laura Leal-Taixé, James Hays, and Deva Ramanan. Soft augmentation for image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16241–16250, 2023.
[39] Huimin Wu, Chenyang Lei, Xiao Sun, Peng-Shuai Wang, Qifeng Chen, Kwang-Ting Cheng, Stephen Lin, and Zhirong Wu. Randomized quantization: A generic augmentation for data agnostic self-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16305–16316, 2023.
[40] Jie Qin, Jiemin Fang, Qian Zhang, Wenyu Liu, Xingang Wang, and Xinggang Wang. Resizemix: Mixing data with preserved object information and true labels. *arXiv preprint arXiv:2012.11101*, 2020.
[41] Ethan Harris, Antonia Marcu, Matthew Painter, Mahesan Niranjan, and Adam Prügel-Bennett Jonathon Hare. Fmix: Enhancing mixed sample data augmentation. *arXiv preprint arXiv:2002.12047*, 2020.
[42] Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596*, 2014.
[43] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4510–4520, 2018.
[44] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *International Conference on Computer Vision (ICCV)*, 2021.
[45] Lin Zhang, Zaixi Cheng, Ying Shen, and Dongqing Wang. Palmprint and palmvein recognition based on dcnn and a new large-scale contactless palmvein dataset. *Symmetry*, 10(4):78, 2018.
[46] Pedro Tome and Sébastien Marcel. On the vulnerability of palm vein recognition to spoofing attacks. In *2015 International Conference on Biometrics (ICB)*, pages 319–325. IEEE, 2015.
[47] Mohd Shahrimie Mohd Asaari, Shahrel A Suandi, and Bakhtiar Affendi Rosdi. Fusion of band limited phase only correlation and width centroid contour distance for finger based biometrics. *Expert Systems with Applications*, 41(7):3367–3382, 2014.
[48] Yilong Yin, Lili Liu, and Xiwei Sun. Sdumla-hmt: A multimodal biometric database. In *Chinese conference on biometric recognition*, pages 260–268. Springer, 2011.
[49] Dan Hendrycks, Norman Mu, Ekin D. Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. AugMix: A simple data processing method to improve robustness and uncertainty. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.
[50] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019.
[51] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.
[52] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.
[53] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.
[54] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.
[55] Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. Manifold mixup: Better representations by interpolating hidden states. In *International Conference on Machine Learning*, pages 6438–6447, 2019.
[56] Khawar Islam, Muhammad Zaigham Zaheer, Arif Mahmood, and Karthik Nandakumar. Diffusemix: Label-preserving data augmentation with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27621–27630, 2024.

## APPENDIX

### A. Additional Full Results

This appendix reports the complete experimental tables that are summarized in the main paper. These tables are intended to provide a model-wise and dataset-wise view of the robustness, calibration, and efficiency behavior of each augmentation method. Unless otherwise specified, the notation follows the main text: the best results are highlighted in bold with a blue background, the second-best results are underlined, and gray values indicate performance worse than the corresponding Vanilla baseline under the same model and dataset setting.

- **Full Calibration Results.** Table XI reports the full calibration results measured by ECE across VERA220, TJU600, and SCUT1100. Lower ECE indicates better confidence calibration. For VERA220, results are reported for ResNet18, MobileNetv2, FVRASNet, AMPVNet, and StarLKNet-S. ViT-S and Swin-T are evaluated only on TJU600 and SCUT1100, where the larger-scale settings provide sufficient data for transformer-based backbones. These results expand the calibration discussion in the main paper and show that methods with high recognition accuracy are not necessarily well calibrated.
- **Full Adversarial Robustness Results.** Table XII reports the full SCUT1100 adversarial robustness results under FGSM and PGD attacks. Together with Table VIII in the main text, this table provides a complete view of adversarial behavior on the two vein datasets. The results further support the observation that recognition accuracy, calibration, corruption robustness, and adversarial robustness can favor different augmentation families.
- **Full Corruption Robustness Results.** Tables XIII and XIV provide the complete corruption robustness results under three corruption severity levels. The main paper reports the C1 results to keep the presentation compact, whereas the appendix includes C1, C2, and C3 for each backbone. These full results show how augmentation methods behave as the corruption level increases, which is important for distinguishing methods that only improve mild robustness from those that remain stable under severe degradation.
- **More ROC Curve Results.** Figures 8, 9 and 10 show more compressive results with different models.

TABLE XI
RESULTS OF CALIBRATION OF VARIOUS AUGMENTATIONS ACROSS DIFFERENT MODELS ON VEAR220, TJU600, AND SCUT1100 DATASETS. VIT-S AND SWIN-T ONLY WITH TJU600 AND SCUT1100.

| Calibration | VEAR220 ECE Score (%) ↓ | | | | | TJU600 ECE Score (%) ↓ | | | | | | | SCUT1100 ECE Score (%) ↓ | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | R18 | Mobv2 | FVN | APN | SLK-S | R18 | Mobv2 | FVN | APN | SLK-S | ViT-S | Swin-T | R18 | Mobv2 | FVN | APN | SLK-S | ViT-S | Swin-T |
| Vanilla | 4.10 | 3.17 | 10.38 | 10.07 | 8.17 | 7.96 | 6.73 | 2.17 | 8.42 | 5.58 | 16.78 | 8.77 | 3.66 | 2.90 | 1.95 | 1.39 | 0.78 | 8.83 | 1.18 |
| Flip | 5.23 | 6.02 | 10.62 | 13.48 | 10.38 | 7.07 | 6.24 | 3.93 | 10.35 | 6.00 | 16.27 | 10.05 | 3.48 | 2.78 | 1.69 | 2.17 | 0.95 | 12.87 | 0.66 |
| Rotate | 8.70 | 9.91 | 16.59 | 15.77 | 12.40 | 7.03 | 6.95 | 5.84 | 9.84 | 9.46 | 14.19 | 9.37 | 3.74 | 3.34 | 1.27 | 2.05 | 0.65 | 10.03 | 1.01 |
| Translation | 11.33 | 9.40 | 18.85 | 17.57 | 12.18 | 7.72 | 5.44 | 6.15 | 8.96 | 5.26 | 14.00 | 9.06 | 4.09 | 2.84 | 1.15 | 1.83 | 0.66 | 9.10 | 0.49 |
| Noise | 5.80 | 4.44 | 15.54 | 12.47 | 9.82 | 7.70 | 7.29 | 3.75 | 9.42 | 1.55 | 16.86 | 12.51 | 6.40 | 2.94 | 2.52 | 2.08 | 0.63 | 10.51 | 1.77 |
| Cutout [7] | 12.71 | 15.44 | 18.72 | 14.29 | 12.82 | **0.94** | 3.27 | 10.98 | 8.76 | 4.45 | 12.93 | 7.07 | **1.13** | 3.25 | 3.41 | 4.17 | 3.87 | 5.69 | 0.70 |
| GridMask [36] | 11.15 | 9.60 | 22.00 | 18.46 | 16.33 | 3.97 | **1.19** | 17.31 | 14.29 | 5.64 | 16.46 | 9.96 | 3.45 | **2.26** | 6.99 | 12.30 | 4.67 | 11.96 | 2.54 |
| RandomErasing [18] | 9.45 | 7.11 | 16.03 | 12.47 | 9.90 | 8.47 | 6.78 | 5.21 | 8.27 | 5.57 | 16.11 | 8.78 | 3.51 | 2.80 | **0.98** | 2.00 | 0.90 | 10.20 | 1.07 |
| RandomQuant [39] | **1.62** | **1.49** | 11.73 | 5.70 | 5.44 | 9.75 | 3.73 | 11.33 | 13.99 | 1.68 | 11.68 | 16.79 | 4.50 | 3.73 | 4.12 | 3.54 | 2.03 | 4.27 | 1.13 |
| AutoAugment [5] | 1.91 | 3.44 | 4.05 | 3.61 | 2.78 | 10.00 | 6.13 | 5.98 | 7.43 | 1.71 | 14.92 | 6.16 | 3.11 | 2.55 | 2.17 | 1.32 | 0.56 | 3.71 | 0.54 |
| RandAugment [6] | 3.14 | 2.78 | 5.53 | 4.98 | 3.16 | 10.64 | 6.06 | 4.54 | 6.13 | 1.11 | **9.84** | **5.00** | 3.69 | 3.08 | 1.56 | **1.27** | 1.28 | **2.60** | 0.55 |
| KeepAugment [19] | 2.26 | 4.55 | 9.56 | 3.51 | 11.53 | 8.11 | 7.01 | 2.99 | 8.09 | 2.83 | 16.13 | 9.07 | 3.23 | 2.54 | 1.95 | 1.31 | 0.65 | 8.45 | 0.67 |
| TrivialAugment [13] | 5.88 | 3.48 | **2.96** | **1.74** | **1.34** | 9.92 | 7.16 | 2.22 | **5.58** | **0.54** | 11.42 | 5.77 | 2.86 | 3.01 | 2.02 | 1.36 | 1.02 | 3.46 | **0.41** |
| TeachAugment [14] | 2.19 | 8.10 | 4.58 | 11.47 | 9.89 | 5.58 | 3.50 | 17.99 | 10.89 | 4.92 | 10.14 | 6.77 | - | 11.56 | - | 9.90 | - | 2.61 | 0.57 |
| SoftAugment [38] | 6.65 | 5.67 | 10.23 | 10.24 | 9.45 | 4.65 | 4.36 | 3.04 | 9.23 | 4.13 | 15.07 | 8.63 | 2.95 | 2.46 | 1.77 | 1.71 | **0.47** | 9.32 | 1.00 |
| YOCO [37] | 15.98 | 18.60 | 22.95 | 14.51 | 14.17 | 5.61 | 1.88 | 12.01 | 9.31 | 1.08 | 11.01 | 10.56 | 4.59 | 3.02 | 23.29 | 42.93 | 71.75 | 6.78 | 0.99 |
| RICAP [21] | 25.71 | 18.99 | 7.19 | 18.84 | 18.08 | 33.16 | 35.17 | 35.41 | 36.57 | 37.79 | 26.46 | 33.86 | 9.57 | 11.23 | 17.25 | 15.09 | 9.77 | 15.80 | 8.22 |
| MixUp [8] | 31.26 | 31.45 | 37.74 | 30.88 | 35.60 | 33.77 | 36.09 | 35.83 | 35.59 | 38.65 | 26.56 | 33.86 | 8.93 | 11.13 | 17.03 | 14.74 | 9.14 | 15.81 | 8.22 |
| CutMix [15] | 21.20 | 15.50 | 5.43 | 13.15 | 11.89 | 24.99 | 23.40 | 18.85 | 20.65 | 13.76 | 8.87 | 9.78 | 13.91 | 15.54 | 11.33 | 7.45 | 7.84 | 15.12 | 6.29 |
| FMix [41] | 15.73 | 12.83 | 5.04 | 12.82 | 7.27 | 26.47 | 29.02 | 20.61 | 21.67 | 20.87 | 7.50 | 10.01 | 10.47 | 11.15 | 9.46 | 9.60 | 7.61 | 12.60 | **5.15** |
| GridMix [20] | 11.99 | 12.23 | 3.02 | 3.47 | **6.84** | 17.30 | 22.10 | **2.09** | 11.73 | 18.58 | **2.79** | 8.86 | 11.44 | 11.02 | 12.45 | 7.37 | 8.47 | 10.22 | 6.62 |
| ResizeMix [40] | 20.33 | 14.66 | **2.92** | 14.42 | 8.98 | 21.65 | 25.10 | 15.69 | 21.58 | 14.76 | 13.07 | 11.18 | 9.68 | 12.40 | 11.49 | 9.50 | 7.04 | 18.37 | 6.95 |
| SaliencyMix [22] | 16.53 | 12.11 | 4.10 | 12.30 | 8.46 | 22.83 | 25.08 | 16.62 | 17.44 | **13.49** | 9.65 | 9.61 | 8.91 | 11.05 | 9.34 | 6.26 | 4.64 | 13.65 | 5.89 |
| PuzzleMix [23] | 29.65 | 28.93 | 25.36 | 25.76 | 28.80 | 31.13 | 31.49 | 23.27 | 26.50 | 40.65 | 24.97 | 24.91 | 26.45 | 25.82 | 22.95 | 19.28 | 22.45 | 26.54 | 24.87 |
| GuidedMixup [12] | 9.32 | 10.57 | 8.41 | - | 13.60 | 33.40 | 36.16 | 36.05 | 35.69 | 39.87 | 27.21 | 33.86 | 9.20 | 11.09 | 16.93 | 12.86 | 10.67 | 15.93 | 8.22 |
| StarMixup [9] | 35.30 | 33.69 | 39.81 | 37.06 | 34.21 | 38.34 | 39.74 | 38.80 | 41.22 | 45.50 | 27.24 | 35.75 | 9.59 | 12.42 | 19.85 | 15.29 | 8.99 | 15.89 | 8.58 |
| LabelSmoothing [24] | 38.42 | 30.23 | 25.16 | 43.49 | 24.21 | 47.88 | 43.12 | 36.33 | 46.64 | 33.45 | 33.08 | 47.57 | 26.18 | 21.07 | 25.65 | 25.24 | 18.14 | 36.40 | 30.25 |
| OnlineLabelSmooth [26] | 4.41 | **2.68** | **3.88** | 4.22 | 7.56 | 11.74 | 11.26 | 8.21 | **4.22** | 14.52 | **1.41** | 7.34 | 5.74 | 4.82 | 6.77 | 3.29 | 6.32 | 5.21 | 5.44 |
| ConfidencePenalty [25] | 3.57 | 3.28 | 6.20 | 8.49 | 5.75 | 16.99 | 15.84 | 5.96 | 4.91 | **1.67** | 10.81 | **5.62** | 7.03 | 6.58 | 4.22 | 0.92 | 1.55 | **4.90** | **0.57** |
| DirichletLabelSmooth [27] | 39.10 | 31.32 | 25.35 | 43.35 | 22.96 | 47.80 | 44.01 | 38.53 | 47.18 | 34.69 | 33.52 | 47.47 | 26.01 | 22.38 | 25.79 | 25.36 | 19.88 | 36.62 | 29.76 |
| Bootstrapping [42] | **3.31** | 4.14 | 8.56 | **3.32** | 17.05 | 15.92 | 10.05 | 4.16 | 7.90 | 2.95 | 15.80 | 8.05 | 9.38 | 11.90 | **1.77** | **0.76** | 7.69 | 8.45 | 0.73 |

TABLE XII
TOP-1 ACCURACY. (%) ↑ OF VARIOUS AUGMENTATIONS ACROSS DIFFERENT MODELS UNDER FGSM AND PGD ATTACKS ON SCUT1100 DATASET.

| SCUT1100 | R18 | | Mobv2 | | FVN | | APN | | SLK-S | | ViT-S | | Swin-T | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | FGSM | PGD | FGSM | PGD | FGSM | PGD | FGSM | PGD | FGSM | PGD | FGSM | PGD | FGSM | PGD |
| Vanilla | 90.38 | 89.86 | 85.80 | 83.75 | 8.73 | 0.14 | 86.20 | 85.27 | 86.42 | 85.76 | 49.91 | 46.40 | 43.66 | 30.36 |
| Flip | 87.53 | 86.44 | 83.13 | 81.31 | 8.76 | 0.55 | 82.14 | 81.09 | 80.44 | 78.94 | 42.89 | 40.13 | 35.26 | 21.62 |
| Rotate | 85.75 | 84.00 | 62.18 | 48.62 | 15.40 | 2.67 | 26.33 | 9.74 | 59.16 | 45.18 | 46.93 | 42.95 | 31.14 | 16.20 |
| Translation | 87.49 | 86.18 | 78.44 | 71.09 | 21.78 | 2.49 | 52.09 | 33.96 | 69.78 | 61.00 | 42.33 | 37.27 | 39.87 | 25.40 |
| Noise | 77.89 | 77.40 | **88.89** | **88.25** | **61.71** | **57.20** | 85.80 | 85.38 | **88.78** | **88.33** | 60.07 | 59.66 | **74.67** | **73.24** |
| Cutout [7] | 65.44 | 62.66 | 57.76 | 53.96 | 8.38 | 0.24 | 33.07 | 16.34 | 65.86 | 62.96 | 63.44 | 61.00 | 45.60 | 31.64 |
| GridMask [36] | 71.09 | 70.00 | 69.07 | 68.31 | 14.22 | 5.00 | 15.09 | 6.24 | 64.89 | 63.98 | 58.64 | 57.82 | 71.55 | 69.80 |
| RandomErasing [18] | 90.84 | 90.36 | 86.78 | 85.20 | 41.45 | 20.82 | **88.87** | **87.78** | 86.53 | 85.55 | 52.16 | 49.45 | 46.76 | 33.11 |
| RandomQuant [39] | 87.02 | 83.44 | 87.02 | 82.62 | 38.69 | 27.69 | 86.53 | 85.94 | 85.55 | 81.06 | **73.55** | **72.67** | 74.11 | 71.24 |
| AutoAugment [5] | 86.31 | 82.33 | 77.47 | 64.55 | 4.86 | 0.04 | 71.91 | 64.49 | 66.71 | 42.53 | 61.04 | 57.34 | 11.51 | 0.71 |
| RandAugment [6] | 89.11 | 86.36 | 82.56 | 74.89 | 3.64 | 0.00 | 81.42 | 77.93 | 74.40 | 54.67 | 65.93 | 61.76 | 32.18 | 15.09 |
| KeepAugment [19] | 90.82 | 90.20 | 87.24 | 85.71 | 8.54 | 0.04 | 85.27 | 84.14 | 86.16 | 85.09 | 49.76 | 46.85 | 44.87 | 31.27 |
| TrivialAugment [13] | **91.64** | 89.44 | 83.96 | 77.84 | 4.76 | 0.02 | 73.47 | 65.71 | 80.86 | 75.11 | 63.16 | 59.00 | 44.33 | 27.69 |
| TeachAugment [14] | 0.26 | 0.11 | 20.11 | 10.07 | 0.00 | 0.00 | 15.53 | 0.56 | 0.16 | 0.16 | 64.13 | 59.73 | 14.93 | 2.40 |
| SoftAugment [38] | 91.36 | **90.93** | 87.75 | 87.07 | 20.07 | 2.85 | 86.18 | 85.16 | 85.93 | 85.22 | 48.27 | 45.15 | 39.66 | 25.71 |
| YOCO [37] | 24.69 | 21.71 | 39.42 | 33.73 | 1.76 | 0.06 | 10.22 | 8.91 | 0.98 | 0.49 | 48.29 | 44.42 | 25.42 | 13.47 |
| RICAP [21] | 91.71 | 89.24 | 86.27 | 83.27 | 16.31 | 2.44 | 65.07 | 54.13 | 88.36 | 83.53 | 57.02 | 54.47 | 39.80 | 26.05 |
| MixUp [8] | 91.73 | 85.58 | 92.27 | 87.69 | 6.87 | 0.02 | 85.87 | 76.25 | 85.93 | 75.66 | 75.40 | 72.56 | 27.31 | 2.80 |
| CutMix [15] | **95.53** | **94.55** | 91.86 | 90.13 | **41.13** | 9.67 | 90.87 | 89.84 | 91.67 | 89.82 | 72.24 | 70.22 | 58.89 | 41.09 |
| FMix [41] | 93.82 | 92.42 | 89.73 | 87.69 | 27.69 | 3.07 | 90.25 | 89.71 | 82.45 | 73.33 | 64.34 | 62.62 | 61.85 | 42.00 |
| GridMix [20] | 93.71 | 93.00 | 89.18 | 87.96 | 34.34 | 8.96 | 79.16 | 71.76 | 89.29 | 87.38 | 69.76 | 68.25 | 48.93 | 31.22 |
| ResizeMix [40] | 94.84 | 94.11 | **92.45** | **91.36** | 38.80 | **12.31** | **92.09** | **91.64** | **92.47** | **91.09** | 65.84 | 64.36 | 55.51 | 37.64 |
| SaliencyMix [22] | 94.78 | 93.60 | 91.64 | 89.27 | 33.76 | 8.80 | 91.96 | 91.20 | 90.91 | 87.31 | 74.64 | **72.75** | **63.71** | **44.84** |
| PuzzleMix [23] | 93.49 | 89.55 | 88.33 | 76.58 | 6.62 | 0.02 | 85.45 | 80.96 | 84.34 | 61.71 | 70.25 | 67.18 | 61.60 | 39.05 |
| GuidedMixup [12] | 91.91 | 85.62 | 90.09 | 83.20 | 7.18 | 0.06 | 78.62 | 60.38 | 86.75 | 78.20 | **75.42** | 72.64 | | 0.18 |
| StarMixup [9] | 92.64 | 87.11 | 92.02 | 88.56 | 8.93 | 0.06 | 83.75 | 64.07 | 90.06 | 84.02 | 74.38 | 71.31 | 34.53 | 3.54 |
| LabelSmoothing [24] | 96.34 | 96.18 | 93.00 | **92.51** | 47.07 | 11.73 | 95.58 | 95.49 | 92.76 | 92.27 | 61.96 | 61.16 | 77.60 | 73.18 |
| OnlineLabelSmooth [26] | 93.06 | 92.51 | 89.34 | 88.16 | 16.36 | 0.66 | 89.71 | 89.02 | 90.40 | 89.73 | 56.27 | 53.85 | 56.89 | 44.42 |
| ConfidencePenalty [25] | 91.64 | 91.13 | 87.16 | 85.86 | 11.31 | 0.20 | 86.29 | 85.49 | 86.69 | 85.71 | 50.47 | 46.93 | 45.85 | 30.98 |
| DirichletLabelSmooth [27] | **96.53** | **96.44** | **93.04** | 92.47 | **51.87** | **17.42** | **95.73** | **95.64** | **93.36** | **92.96** | **62.73** | **61.87** | **78.51** | **74.09** |
| Bootstrapping [42] | 83.31 | 80.91 | 57.00 | 48.45 | 7.86 | 0.13 | 86.00 | 84.98 | 54.40 | 44.27 | 50.55 | 47.31 | 43.84 | 29.67 |

TABLE XIII
TOP-1 ACCURACY (%) ↑ OF CORRUPTION OF VARIOUS AUGMENTATIONS ACROSS DIFFERENT MODELS ON VEAR220 DATASET.

| Corruption | VERA220-C1 | | | | | VERA220-C2 | | | | | VERA220-C3 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | R18 | Mobv2 | FVN | APN | SLK-S | R18 | Mobv2 | FVN | APN | SLK-S | R18 | Mobv2 | FVN | APN | SLK-S |
| Vanilla | 69.59 | 69.19 | 39.09 | 73.18 | 65.10 | 61.99 | 62.70 | 25.57 | 63.35 | 59.12 | 46.70 | 50.69 | 9.81 | 41.96 | 48.61 |
| Flip | 63.49 | 63.35 | 30.21 | 65.29 | 60.29 | 56.01 | 56.60 | 18.47 | 56.82 | 54.47 | 40.05 | 44.57 | 6.77 | 37.13 | 43.95 |
| Rotate | 62.75 | 59.66 | 32.18 | 62.58 | 58.33 | 53.66 | 53.28 | 20.91 | 51.75 | 52.13 | 38.49 | 40.53 | 8.80 | 35.86 | 37.18 |
| Translation | 60.67 | 64.14 | 37.06 | 63.54 | 60.93 | 53.90 | 56.51 | 25.10 | 52.90 | 55.14 | 38.13 | 43.52 | 10.86 | 36.51 | 42.97 |
| Noise | 67.32 | 68.49 | 37.18 | 69.02 | 66.00 | 61.32 | 61.63 | 28.42 | 61.27 | 60.22 | 48.66 | 50.38 | **16.60** | 41.99 | 48.80 |
| Cutout [7] | 68.06 | 64.88 | 33.95 | 68.28 | 62.82 | 58.73 | 57.77 | 21.51 | 56.70 | 56.05 | 40.50 | 43.64 | 8.56 | 38.28 | 42.32 |
| GridMask [36] | 64.93 | 66.53 | 24.81 | 54.12 | 54.71 | 57.99 | 59.86 | 15.02 | 44.40 | 48.68 | 45.02 | 47.75 | 6.63 | 33.16 | 37.61 |
| RandomErasing [18] | 61.22 | 66.89 | 36.36 | 72.54 | 65.86 | 55.31 | 60.41 | 25.12 | 63.92 | 59.90 | 44.35 | 50.00 | 11.17 | 44.83 | 49.47 |
| RandomQuant [39] | 79.59 | **81.27** | 22.27 | 80.96 | 77.27 | 68.83 | **74.86** | 15.10 | 72.92 | 70.14 | 48.56 | **67.66** | 6.34 | **59.09** | 56.91 |
| AutoAugment [5] | 75.43 | 71.10 | 35.12 | 76.56 | 72.82 | 62.01 | 61.55 | 22.06 | 63.37 | 62.77 | 45.05 | 42.99 | 10.38 | 48.04 | 42.20 |
| RandAugment [6] | 69.43 | 78.50 | 21.72 | 75.60 | **82.47** | 57.73 | 69.41 | 13.88 | 62.09 | **74.96** | 41.42 | 52.28 | 5.49 | 43.89 | 58.33 |
| KeepAugment [19] | 70.52 | 69.21 | 41.97 | 71.49 | 63.63 | 61.43 | 60.77 | 28.60 | 62.34 | 57.54 | 45.24 | 50.32 | 10.97 | 44.60 | 45.21 |
| TrivialAugment [13] | **83.98** | 78.78 | 41.33 | **85.29** | 81.99 | **73.18** | 70.17 | 28.21 | **73.74** | 74.20 | **55.45** | 55.51 | 13.25 | 54.91 | **60.27** |
| TeachAugment [14] | 68.06 | 57.19 | 38.16 | 56.84 | 65.31 | 60.67 | 49.88 | 27.17 | 50.11 | 57.07 | 45.00 | 37.66 | 13.28 | 38.12 | 44.02 |
| SoftAugment [38] | 64.94 | 68.21 | **44.97** | 73.91 | 66.81 | 58.75 | 61.88 | **33.40** | 65.81 | 59.86 | 43.76 | 49.88 | 14.66 | 45.34 | 48.06 |
| YOCO [37] | 62.06 | 64.95 | 29.35 | 71.74 | 58.83 | 53.25 | 56.32 | 21.55 | 61.32 | 52.32 | 39.68 | 43.32 | 8.97 | 45.69 | 40.86 |
| RICAP [21] | 73.15 | 67.37 | 28.27 | 67.72 | 61.02 | 64.29 | 59.57 | 18.23 | 59.20 | 53.86 | 47.88 | 46.16 | 8.23 | 45.26 | 40.91 |
| MixUp [8] | 85.51 | **89.54** | **66.89** | 88.75 | **85.81** | 75.05 | **82.49** | 46.72 | 81.16 | **77.40** | 57.19 | **65.86** | 21.84 | **62.82** | **60.89** |
| CutMix [15] | 76.31 | 88.08 | 35.65 | 77.01 | 70.47 | 68.04 | 80.89 | 23.36 | 69.07 | 63.02 | 51.81 | 62.11 | 10.03 | 51.49 | 47.24 |
| FMix [41] | 75.76 | 73.87 | 41.74 | 75.32 | 66.60 | 66.60 | 63.90 | 29.35 | 66.38 | 58.33 | 49.52 | 48.68 | 13.06 | 49.33 | 44.02 |
| GridMix [20] | 72.63 | 69.17 | 28.52 | 67.33 | 62.59 | 63.31 | 60.10 | 19.34 | 59.39 | 54.77 | 46.68 | 46.54 | 7.87 | 44.40 | 41.77 |
| ResizeMix [40] | 77.06 | 71.55 | 34.02 | 77.72 | 64.21 | 68.99 | 63.37 | 23.11 | 70.16 | 56.75 | 51.10 | 48.89 | 10.23 | 52.21 | 42.92 |
| SaliencyMix [22] | 77.42 | 75.05 | 42.47 | 78.55 | 68.96 | 68.61 | 65.31 | 29.26 | 71.05 | 62.16 | 51.48 | 50.69 | 12.86 | 52.12 | 48.16 |
| PuzzleMix [23] | **87.05** | 88.47 | 60.10 | 84.77 | 84.97 | **77.10** | 81.45 | 41.97 | 76.01 | 75.40 | **58.75** | 65.71 | 19.23 | 57.19 | 59.18 |
| GuidedMixup [12] | 61.85 | 63.15 | 22.61 | 54.85 | 61.98 | 54.34 | 55.19 | 15.68 | 47.88 | 54.37 | 40.67 | 42.66 | 7.56 | 35.80 | 41.14 |
| StarMixup [9] | 87.03 | 84.81 | 66.49 | **89.47** | 84.66 | 76.54 | 75.06 | **47.96** | **82.02** | 76.49 | 57.18 | 56.09 | **24.11** | 61.91 | 59.10 |
| LabelSmoothing [24] | 76.51 | 70.48 | **51.84** | 83.18 | 66.38 | 68.60 | 62.09 | **36.56** | 73.80 | 58.62 | 50.10 | 48.06 | **14.93** | 52.17 | 45.62 |
| OnlineLabelSmooth [26] | 70.83 | 67.14 | 48.56 | 76.01 | **68.23** | 63.34 | 59.54 | 34.42 | 66.52 | **60.59** | 47.96 | 47.47 | 14.88 | 48.04 | **47.66** |
| ConfidencePenalty [25] | 66.52 | 65.17 | 45.54 | 72.54 | 60.91 | 59.04 | 57.82 | 31.42 | 63.78 | 54.39 | 43.90 | 45.28 | 13.44 | 46.51 | 43.62 |
| DirichletLabelSmooth [27] | **77.53** | **71.53** | 51.48 | **84.94** | 64.24 | **69.65** | **63.18** | 35.40 | **75.60** | 56.55 | **51.04** | **49.20** | 14.88 | **53.47** | 43.52 |
| Bootstrapping [42] | 61.85 | 50.15 | 48.06 | 71.49 | 38.16 | 55.24 | 43.20 | 33.56 | 62.38 | 33.45 | 41.51 | 32.74 | 13.88 | 46.33 | 26.15 |

TABLE XIV
TOP-1 ACCURACY (%) ↑ OF CORRUPTION OF VARIOUS AUGMENTATIONS ACROSS DIFFERENT MODELS ON TJU600 DATASET.

| Corruption | TJU600-C1 | | | | | | | TJU600-C2 | | | | | | | TJU600-C3 | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | R18 | Mobv2 | FVN | APN | SLK-S | ViT-S | Swin-T | R18 | Mobv2 | FVN | APN | SLK-S | ViT-S | Swin-T | R18 | Mobv2 | FVN | APN | SLK-S | ViT-S | Swin-T |
| Vanilla | 76.99 | 77.64 | 45.82 | 70.11 | 60.90 | 57.05 | 63.47 | 59.31 | 67.13 | 31.96 | 57.81 | 48.51 | 45.22 | 48.40 | 30.01 | 42.52 | 12.84 | 36.87 | 24.43 | 25.29 | 22.26 |
| Flip | 67.48 | 71.81 | 39.58 | 65.62 | 52.43 | 52.26 | 59.35 | 50.34 | 59.77 | 27.34 | 52.70 | 39.39 | 40.84 | 44.10 | 21.48 | 35.85 | 11.23 | 33.56 | 10.79 | 22.75 | 19.20 |
| Rotate | 72.36 | 73.76 | 39.74 | 64.62 | 55.62 | 58.75 | 64.13 | 55.84 | 61.23 | 26.53 | 51.27 | 39.88 | 46.85 | 50.29 | 28.03 | 39.53 | 11.32 | 38.06 | 16.86 | 28.75 | 25.31 |
| Translation | 76.57 | 76.25 | 44.71 | 68.70 | 61.87 | 59.65 | 62.51 | 58.85 | 62.83 | 31.70 | 54.50 | 46.28 | 46.65 | 47.54 | 29.78 | 40.96 | 12.88 | 39.67 | 26.80 | 24.71 | 22.09 |
| Noise | 79.48 | 78.58 | 47.59 | 63.63 | 76.83 | 57.52 | 64.32 | 66.68 | 69.15 | **34.72** | 50.90 | **67.34** | 50.50 | **54.99** | **44.38** | 47.80 | 13.54 | 36.53 | **45.05** | 34.97 | **35.85** |
| Cutout [7] | 75.82 | 74.17 | 46.94 | 64.47 | 68.08 | 58.92 | 64.38 | 56.95 | 59.36 | 32.53 | 51.28 | 52.38 | 46.67 | 49.19 | 25.25 | 34.86 | 13.07 | 38.05 | 23.48 | 26.96 | 22.15 |
| GridMask [36] | 71.11 | 73.86 | 39.40 | 55.95 | 70.14 | 57.47 | 63.33 | 56.69 | 64.08 | 26.61 | 45.24 | 58.69 | 47.04 | 51.83 | 30.80 | 43.44 | 11.87 | 32.42 | 34.78 | 28.34 | 34.63 |
| RandomErasing [18] | 79.53 | **80.35** | 47.08 | **75.83** | 67.57 | 58.20 | **66.89** | 63.53 | **69.88** | 33.28 | **65.07** | 53.50 | 46.56 | 53.84 | 37.66 | 45.76 | 15.01 | 40.08 | 30.30 | 27.29 | 26.66 |
| RandomQuant [39] | 69.27 | 76.97 | 47.66 | 67.39 | 72.37 | **65.68** | 47.00 | 54.49 | 69.13 | 33.35 | 59.96 | 59.03 | **58.27** | 40.46 | 23.54 | **51.87** | 16.54 | 41.72 | 23.66 | **41.51** | 17.46 |
| AutoAugment [5] | 77.11 | 76.68 | 42.08 | 70.22 | **77.72** | 52.97 | 64.53 | 60.48 | 63.35 | 29.39 | 56.71 | 64.39 | 41.28 | 52.02 | 34.26 | 34.72 | 14.68 | 44.41 | 40.25 | 21.89 | 24.49 |
| RandAugment [6] | 80.89 | 77.96 | 45.45 | 73.39 | 69.45 | 65.25 | 65.64 | **67.05** | 66.13 | 31.86 | 59.71 | 51.65 | 53.46 | 52.71 | 42.70 | 49.28 | 16.36 | **47.46** | 33.03 | 34.50 | 25.92 |
| KeepAugment [19] | 76.08 | 79.27 | 44.26 | 70.83 | 74.81 | 56.97 | 64.33 | 58.23 | 68.84 | 31.07 | 57.29 | 60.87 | 45.39 | 49.33 | 28.86 | 42.87 | 12.24 | 37.30 | 35.95 | 25.00 | 23.32 |
| TrivialAugment [13] | **82.67** | 79.61 | **48.62** | 73.22 | 74.72 | 64.40 | 66.57 | 66.80 | 65.38 | 34.09 | 59.88 | 51.79 | 52.46 | 53.45 | 44.35 | 46.39 | **17.96** | 46.29 | 35.98 | 33.14 | 27.01 |
| TeachAugment [14] | 71.21 | 62.24 | 33.52 | 49.28 | 58.22 | 64.16 | 62.38 | 54.51 | 49.08 | 24.71 | 38.36 | 44.73 | 51.03 | 47.42 | 29.11 | 21.80 | 12.82 | 27.58 | 24.43 | 28.43 | 21.74 |
| SoftAugment [38] | 77.96 | 78.23 | 46.75 | 71.33 | 71.01 | 56.73 | 65.58 | 62.06 | 66.87 | 33.00 | 59.59 | 58.58 | 45.82 | 52.88 | 33.33 | 41.67 | 13.38 | 37.98 | 32.55 | 25.86 | 25.79 |
| YOCO [37] | 63.75 | 67.72 | 39.80 | 68.13 | 65.21 | 54.05 | 51.64 | 47.30 | 49.60 | 26.83 | 55.17 | 47.20 | 41.97 | 36.90 | 18.74 | 19.11 | 11.16 | 37.84 | 16.67 | 22.67 | 15.65 |
| RICAP [21] | 74.39 | 71.76 | 48.40 | 53.07 | 77.04 | 46.15 | 48.43 | 57.25 | 53.63 | 34.20 | 43.05 | 59.08 | 35.22 | 35.38 | 26.70 | 22.96 | 14.14 | 30.18 | 28.54 | 18.66 | 15.03 |
| MixUp [8] | 88.72 | 88.63 | 61.52 | 75.82 | 89.97 | **75.80** | 80.19 | **75.66** | 75.06 | 47.48 | 63.22 | 76.76 | **62.23** | **65.17** | **40.20** | 43.42 | **21.92** | **49.26** | 40.33 | **38.83** | 34.65 |
| CutMix [15] | 83.12 | 83.50 | 54.30 | 68.67 | 83.01 | 61.20 | 63.47 | 64.57 | 64.81 | 38.41 | 56.53 | 66.71 | 47.98 | 48.82 | 31.39 | 36.81 | 16.01 | 42.41 | 37.11 | 27.91 | 21.91 |
| FMix [41] | 82.50 | 79.51 | 53.00 | 67.92 | 78.58 | 59.09 | 69.64 | 64.30 | 63.52 | 37.82 | 54.83 | 61.73 | 47.27 | 55.00 | 34.77 | 40.68 | 17.85 | 42.79 | 36.61 | 28.52 | 26.73 |
| GridMix [20] | 78.48 | 79.46 | 30.89 | 63.98 | 79.89 | 60.21 | 66.17 | 61.33 | 67.89 | 23.62 | 52.13 | 67.09 | 48.46 | 53.11 | 31.84 | 43.14 | 10.83 | 39.97 | 40.22 | 28.73 | 23.61 |
| ResizeMix [40] | 78.16 | 76.65 | 48.23 | 66.78 | 79.91 | 57.56 | 61.24 | 62.69 | 58.66 | 35.29 | 54.80 | 61.53 | 45.12 | 45.59 | 33.65 | 34.08 | 16.80 | 41.75 | 33.02 | 27.67 | 24.81 |
| SaliencyMix [22] | 82.61 | 82.52 | 54.02 | 71.92 | 83.33 | 62.70 | 63.88 | 64.48 | 63.66 | 38.12 | 58.23 | 65.64 | 49.66 | 48.36 | 31.69 | 32.41 | 16.07 | 43.14 | 35.42 | 30.37 | 22.20 |
| PuzzleMix [23] | **89.49** | 88.92 | **63.74** | 75.04 | 89.51 | 74.24 | 73.81 | 73.25 | 74.62 | **47.87** | 61.82 | 76.89 | 60.25 | 56.35 | 37.99 | 39.86 | 21.11 | 47.68 | 40.92 | 37.95 | 27.80 |
| GuidedMixup [12] | 63.44 | 60.42 | 23.50 | 54.54 | 63.09 | 56.00 | 40.75 | 50.74 | 47.41 | 16.48 | 46.42 | 49.53 | 45.11 | 35.49 | 24.36 | 23.64 | 6.63 | 33.54 | 26.15 | 24.86 | 16.68 |
| StarMixup [9] | 88.64 | **89.99** | 62.22 | **78.95** | **91.39** | 73.73 | **80.34** | 74.39 | **77.97** | 47.54 | **65.34** | **79.00** | 60.12 | 64.83 | 40.01 | **49.60** | 20.98 | 48.90 | **47.73** | 37.58 | **35.72** |
| LabelSmoothing [24] | 89.53 | 88.47 | **59.15** | **82.44** | 86.39 | **60.95** | 76.80 | **72.22** | **78.85** | **43.23** | **67.84** | 73.25 | **48.29** | 59.58 | 40.13 | **52.00** | **18.22** | **47.53** | 41.10 | **27.96** | **32.05** |
| OnlineLabelSmooth [26] | 79.33 | 80.50 | 47.70 | 75.32 | 81.60 | 59.62 | 69.21 | 61.26 | 70.15 | 33.66 | 62.08 | 69.68 | 47.44 | 53.16 | 31.74 | 43.45 | 13.57 | 39.87 | 41.06 | 26.80 | 25.35 |
| ConfidencePenalty [25] | 77.00 | 77.98 | 45.66 | 73.55 | 74.36 | 58.12 | 64.06 | 59.33 | 67.84 | 31.93 | 62.00 | 60.00 | 46.27 | 49.25 | 29.95 | 43.11 | 12.69 | 38.48 | 35.45 | 25.90 | 22.95 |
| DirichletLabelSmooth [27] | **89.73** | **88.95** | 58.38 | 81.26 | **87.47** | 60.81 | **77.56** | 72.13 | 78.42 | 42.96 | 66.83 | **75.49** | 48.05 | **60.10** | **40.89** | 50.15 | 18.18 | 47.36 | **43.90** | 27.85 | 31.94 |
| Bootstrapping [42] | 61.58 | 50.43 | 19.64 | 72.39 | 55.50 | 57.64 | 64.58 | 45.15 | 35.79 | 12.46 | 60.06 | 40.70 | 46.01 | 49.26 | 18.70 | 15.26 | 4.79 | 37.65 | 18.18 | 25.58 | 23.18 |

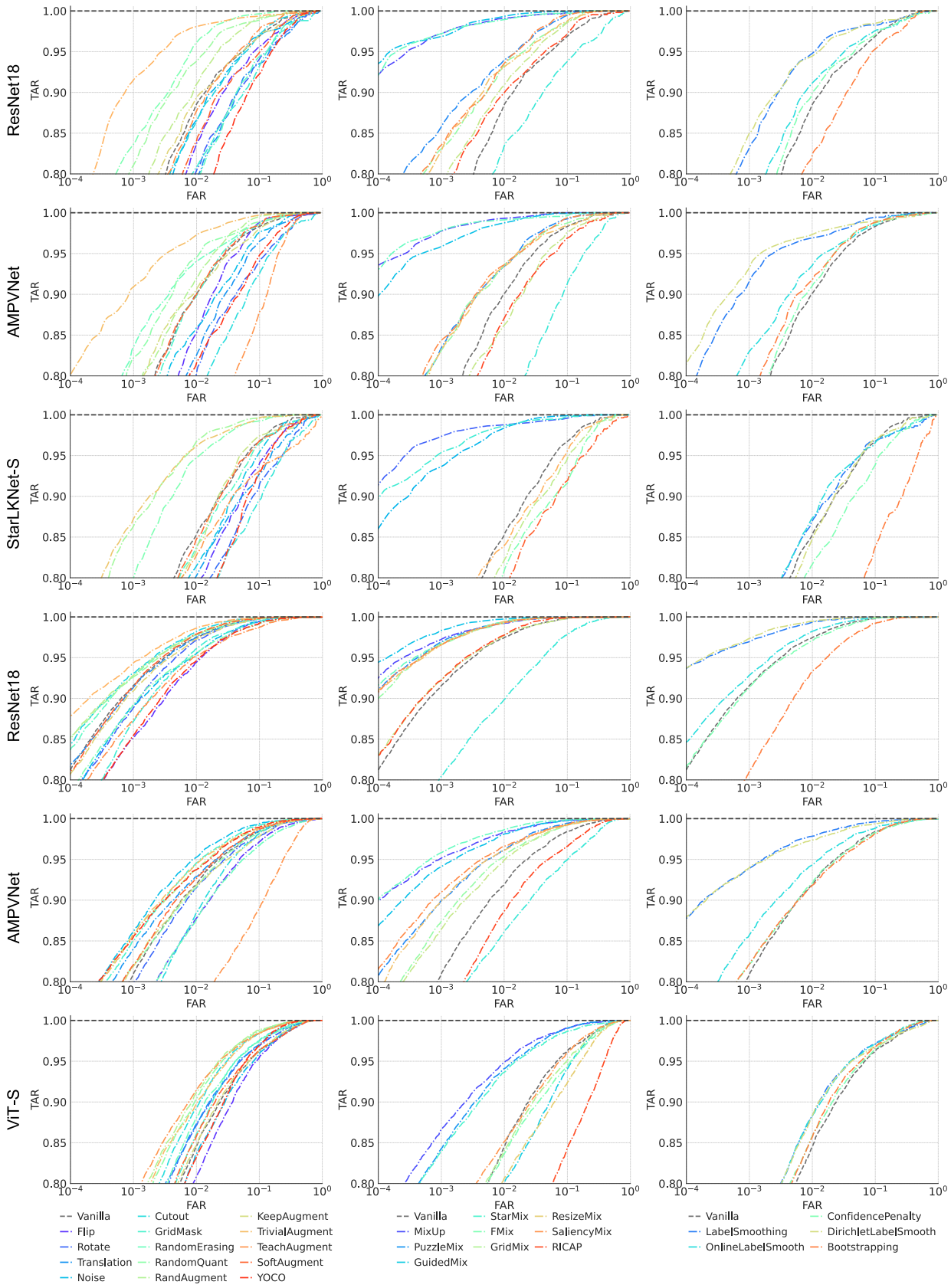


Fig. 8. Receiver Operating Characteristic (ROC) curves of various data augmentation methods across two palm-vein datasets using different backbones. The results for VERA220 are shown in the top three rows, and the results for TJU600 are shown in the bottom three rows.

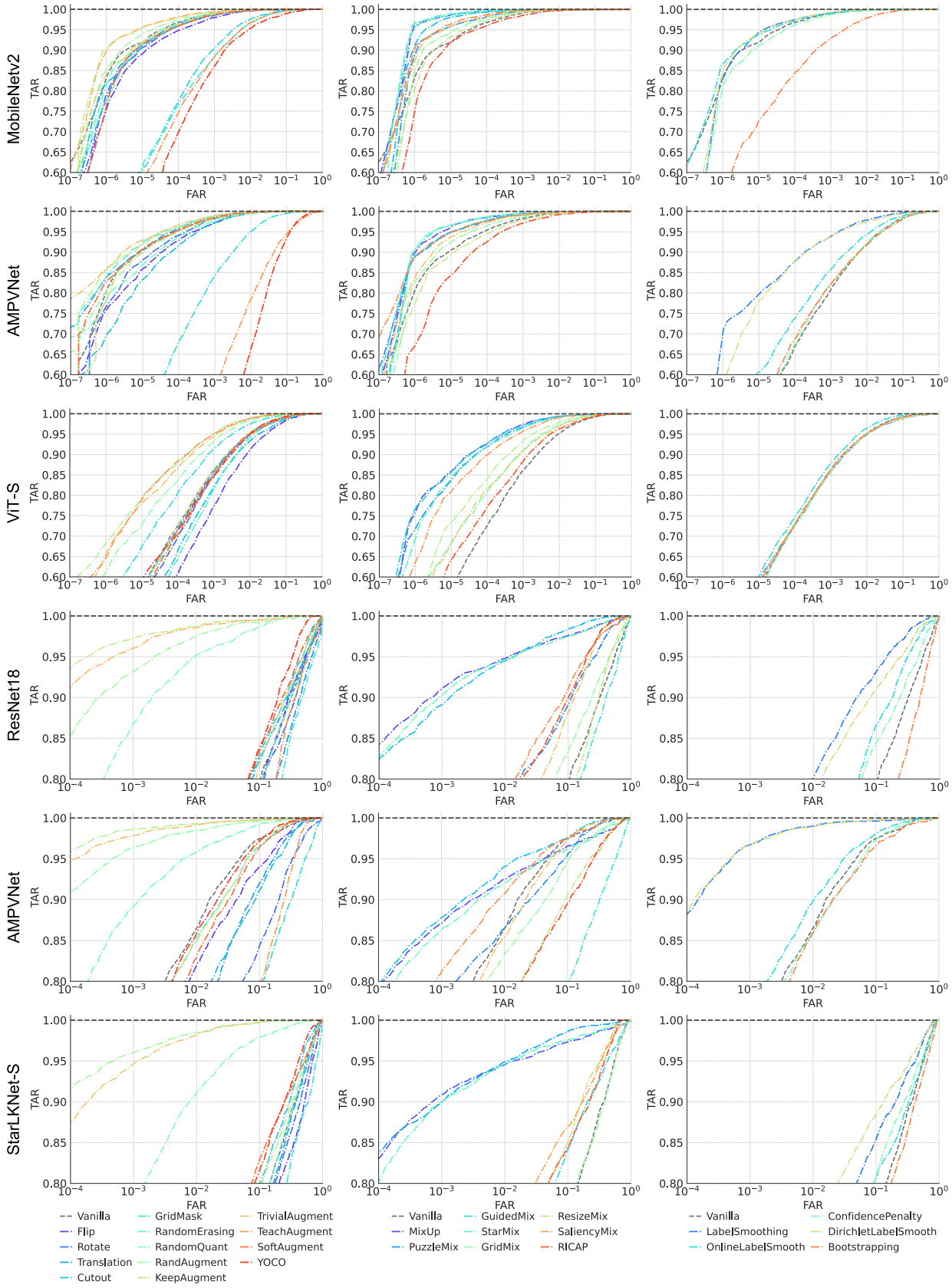


Fig. 9. Receiver Operating Characteristic (ROC) curves of various data augmentation methods across two palm-vein datasets using different backbones. The results for SCTU1100 are shown in the top three rows, and the results for FV-USM are shown in the bottom three rows.

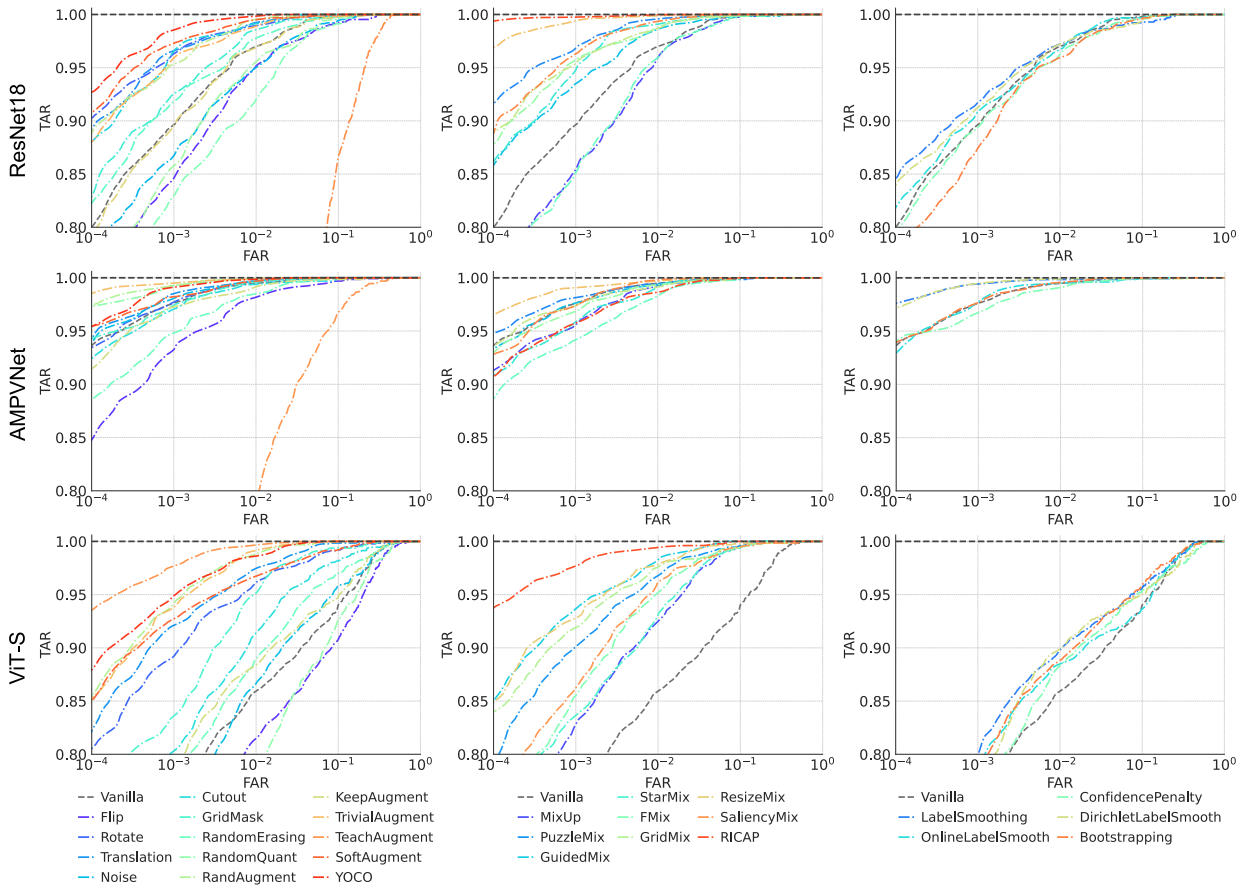


Fig. 10. Receiver Operating Characteristic (ROC) curves of various data augmentation methods across SDUMLA-HMT datasets using different backbones.