LEARNING TO GENERALIZE FROM THE VICINITY: KNOWLEDGE EXPANSION FROM A SINGLE TEACHER

Anonymous authors
Paper under double-blind review

ABSTRACT

When empirical data distributions poorly represent the true data manifold, models exposed to local continuities around data tend to generalize better than those that tightly fit individual data. Similarly, in Knowledge Distillation (KD), when teacher representations poorly reflect the true representation manifold, can exposing a student to local neighborhoods provide greater benefits than fitting to individual teacher targets? Put simply, *Can vicinal distributions around teacher targets help students learn better?* We investigate this by introducing a novel distillation objective where we admit a generative vicinal distribution for each individual teacher target; *Generative Vicinity Distillation (GVD)*. GVD learns a generative prior over the teacher representation space to enable sampling from vicinities and derives guidance functions to impose desired constraints on the generated samples (class consistency, locality). Departing from conventional KD thinking that depends on multi-teacher supervision for robustness, diversity, and fidelity, we instead exploit the ambiguity of a generative model's solution space to derive such properties from a single teacher. Extensive experiments across datasets and architectures demonstrate the effectiveness of GVD in improving student generalization and robustness.

1 INTRODUCTION

Knowledge Distillation (KD) (Buciluǎ et al., 2006; Hinton et al., 2015) is a training paradigm that improves the generalization of a classifier (student) using supervision derived from another model (teacher). KD uses teacher predictive distributions, intermediate representations (Romero et al., 2015), or induced structures such as input relationships (Park et al., 2019) as supervisory targets, which have been shown to restore granularity lost in hard-label supervision (Menon et al., 2021). Nevertheless, supervision in KD remains fundamentally *point-wise*: for each input, the student is supervised by an individual teacher target that remains fixed throughout training. This setting raises three key concerns: *i) teacher reliability*, as individual targets may inherit teacher biases, overconfidence, and limited diversity; *ii) student absorption*, as fixed targets may exceed the student's capacity in early learning; and *iii) teacher–student alignment*, as point-wise alignment may be insufficient for faithful transfer of teacher behavior. We revisit each challenge below and derive a corresponding design question.

Deriving robust supervision from imperfect teachers: Prior works posit supervision from a single teacher parameterization can be unreliable: overconfident teachers may offer no improvement over hard labels (Dong et al., 2019); a single model may be unable to capture all views of multi-view data (Allen-Zhu & Li, 2020). As a result, supervision from a single teacher has been extended to multiple teachers (You et al., 2017; Allen-Zhu & Li, 2020), ensembles of teacher assistants (Son et al., 2021; Mirzadeh et al., 2020), multiple teacher tails (Yu et al., 2025), and mutually learning student shelves (Furlanello et al., 2018; Zhang et al., 2018). These methods share the common intuition that aggregating complementary views from distinct model parameterizations is more robust than relying on a single model parameterization. Since DNN ensembles often differ only in random initialization, multi-teacher supervision may offer limited improvement that does not justify its implementation cost, due to shared inductive biases and limited diversity (Ortega et al., 2022; Allen-Zhu & Li, 2020; Yu et al., 2025). A technique that generates diverse views from a single teacher would be advantageous, providing the robustness of multi-teacher supervision while avoiding the cost of training multiple models. Recent work (Hossain et al., 2025) generates multiple augmented targets by perturbing single teacher outputs with Gaussian random noise. However, noise perturbations neglect the geometry of the teacher representation manifold, potentially producing label-conflicting targets that degrade

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

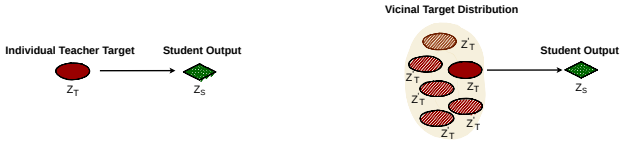


Figure 1: Left: For each input, conventional KD defines supervision at an individual teacher target. Right: We extend supervision to a vicinal distribution around each teacher target.

supervision. **1) Can a single teacher generate multiple plausible targets that remain consistent with its representation distribution and provide reliable supervision?**

Efficient student learning from derived teacher supervision: Even with improved teacher supervision, the gains from KD remain limited, as the student fails to effectively incorporate supervisory signals during training (Cho & Hariharan, 2019; Mirzadeh et al., 2020; Qiu et al., 2022; Stanton et al., 2021). DNNs often fit target functions from low to high frequencies, comprising an initial fast nascent phase that fits low-frequency components, followed by a fine-tuning phase (Xu et al., 2019; Shamir et al., 2021; Ji & Zhu, 2020). During the nascent phase of student learning, fully optimized teacher targets from the fine-tuned phase may be misaligned with the student's capacity, leading to suboptimal convergence (Ji & Zhu, 2020; Dong et al., 2019). Simplified teacher targets from early-stopped teachers (Dong et al., 2019; Cho & Hariharan, 2019), teacher-assistant models (Mirzadeh et al., 2020; Son et al., 2021), and student-friendly teacher training (Park et al., 2021; Rao et al., 2023) can provide more comprehensible supervision, but may sacrifice the expressive power of fully optimized teacher targets. Moreover, these methods define the progression of teacher targets in a few discrete stages that remain fixed throughout student training. **2) Can teacher targets follow a controllable progression that gradually evolves from simplified to expressive targets with the student learning?**

Distilled students inherit robustness of teacher: KD is widely motivated as a means to deploy powerful teacher models in low-latency environments using parameter-efficient students (Cho & Hariharan, 2019; Romero et al., 2015; Ahn et al., 2019). While large models can often discover structure in data, smaller models lack the inductive biases to learn such representations independently. Transferring well-established properties of large models, such as calibrated uncertainties and robustness, is therefore beneficial and can be achieved by maintaining high *teacher–student fidelity*, i.e., the ability of the student to match the teacher predictions (Stanton et al., 2021). Although existing KD improve predictive accuracy, achieving high *fidelity* remains extremely challenging (Stanton et al., 2021). It is well known that different architectures and model complexities can learn decision boundaries that align on the data manifold but diverge significantly in regions without training data (Shamir et al., 2021). This suggests that matching the teacher only on observed data may insufficient for high *teacher–student fidelity*. Instead, fidelity requires transferring the teacher behavior beyond the training distribution. **3) Can distillation loss extend supervision beyond the distillation dataset?**

To address these design questions, we revisit knowledge distillation by extending individual teacher targets into vicinal distributions around each target (Figure 1). Specifically, we augment single-teacher supervision by generating multiple augmented teacher targets in the vicinity of each target. When designing such augmentation, we choose the following design criteria to address the above design questions in distillation: **1. Acknowledging teacher imperfections:** We model each teacher target as a set of vicinal targets generated through meaningful perturbations that (i) preserve the class prediction and (ii) maintain locality to the original target. **2. Progressive supervision:** We regulate the level of supervision, from coarse to refined, by controlling the strength of coupling between each vicinal target and the original teacher target. **3. Distillation set expansion:** We expand each discrete teacher target into a local distribution within a generated neighborhood. With these, we depart from conventional feature augmentation and propose *Generative Vicinity*, which learns a score-based generative prior over teacher target distributions to sample teacher-like vicinal targets around each target. The student is then trained by distilling from the generated vicinities; *Generative Vicinity Distillation* (GVD).

Our contributions: 1) Reformulate KD from point-wise supervision to learning from a vicinal distribution over teacher representations. We derive a novel distillation objective that enforces consistency within this teacher-defined neighborhood, encouraging student representations that better reflect the underlying manifold. 2) Introduce a novel vicinal sampling framework that learns vicinal geometry, enabling controllable sampling under constraints difficult to achieve with predefined geometries. 3) Demonstrate that vicinal supervision improves student generalization, calibration, and teacher–student fidelity, and enables counterfactual teacher supervision that corrects teacher errors.

2 METHODS

Diffusion Models. Real-world high-dimensional data often lie on a low-dimensional manifold $\mathcal{M}$ within the ambient space. Diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020) learn complex data distributions without prior knowledge of the manifold by transforming noise $\mathbf{z}_T \sim \mathcal{N}(0, \mathbf{I})$ into data samples $\mathbf{z}_0 \sim \mathcal{P}_\mathbf{z}$. The forward process $q(\mathbf{z}_{1:T} \mid \mathbf{z}_0)$ progressively adds noise over T transitions, with marginal $q(\mathbf{z}_t \mid \mathbf{z}_0) = \mathcal{N}\big(\sqrt{\alpha_t}\,\mathbf{z}_0,\ (1-\alpha_t)\mathbf{I}\big)$, where the noise schedule $\{\alpha_t\}_{t=1}^T$ transitions the data ($\alpha_1 \approx 1$) toward Gaussian noise ($\alpha_T \approx 0$). The reverse process $p_{\boldsymbol{\phi}}$ progressively denoises $\mathbf{z}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ through transitions $p_{\boldsymbol{\phi}}(\mathbf{z}_{t-1} \mid \mathbf{z}_t) = q\big(\mathbf{z}_{t-1} \mid \mathbf{z}_t, \mathbf{z}_0 = \hat{\mathbf{z}}_0(\mathbf{z}_t, t)\big)$, where $q(\mathbf{z}_{t-1} \mid \mathbf{z}_t, \mathbf{z}_0)$ is a posterior derived from the forward process and $\hat{\mathbf{z}}_0(\mathbf{z}_t, t)$ estimates the clean sample $\mathbf{z}_0$. Instead of directly predicting $\hat{\mathbf{z}}_0$, diffusion models (Ho et al., 2020) commonly learn a noise predictor $\epsilon_{\boldsymbol{\phi}}(\mathbf{z}_t, t)$ to estimate the noise in $\mathbf{z}_t$ and recover the clean sample:

$$\hat{\mathbf{z}}_0 = \frac{\mathbf{z}_t - \sqrt{1-\alpha_t}\,\epsilon_{\boldsymbol{\phi}}(\mathbf{z}_t, t)}{\sqrt{\alpha_t}} \tag{1}$$

The predicted noise can be related to a predicted score function for the smoothed density $\nabla_{\mathbf{z}_t} \log p(\mathbf{z}_t)$ through (Robbins, 1992): $\epsilon_\phi(\mathbf{z}_t, t) = -\sqrt{1-\alpha_t} s_\phi(\mathbf{z}_t, t)$. Thus, diffusion models learn a neural approximation of the score field, which implicitly captures the data manifold $\mathcal{M}$ by directing samples toward high-density regions (Stanczuk et al., 2022). **Training** minimizes the ℓ_2 loss between the true forward noise $\boldsymbol{\epsilon}$ and the predicted noise $\boldsymbol{\epsilon}_\phi(\mathbf{z}_t, t)$:

$$\mathcal{L}_{\text{Diff}}(\boldsymbol{\phi}) = \mathbb{E}_{\mathbf{z}_0 \sim \mathcal{P}_z,\, t \sim \text{Uniform}\{1,\ldots,T\},\, \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0},\mathbf{I})} \left[\left\| \boldsymbol{\epsilon}_{\boldsymbol{\phi}}\big(\sqrt{\alpha_t}\mathbf{z}_0 + \sqrt{1-\alpha_t}\,\boldsymbol{\epsilon},\, t\big) - \boldsymbol{\epsilon} \right\|_2^2 \right]. \tag{2}$$

Sampling/Guided Sampling. Samples are generated by running the learned reverse transitions starting from Gaussian noise. The generated samples naturally form vicinal distributions around the learned $\mathcal{P}_z$ (Wen & Yang, 2026). Guided sampling augments the sampling with external signals (e.g., labels or features) to steer generation toward desired conditions (Dhariwal & Nichol, 2021).

2.1 PROBLEM FORMULATION: FROM POINT-WISE TO VICINAL TEACHER TARGETS

Consider a K-class classification setting with instance space $\mathcal{X}$ and label space $\mathcal{Y} = [K] = \{1, \ldots, K\}$, where input–label pairs $(\mathbf{x}, y) \in \mathcal{X} \times \mathcal{Y}$ follow an unknown data distribution $\mathcal{P}$. The goal is to learn a classifier $\mathbf{s} : \mathcal{X} \to \mathbb{R}^K$ that minimizes the expected risk: $R(\mathbf{s}) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{P}} \left[\ell\big(y, \mathbf{s}(\mathbf{x})\big) \right]$.
Standard Classification. In practice, the underlying distribution $\mathcal{P}$ is approximated using a finite dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N \sim \mathcal{P}^N$, over which the empirical task loss (cross-entropy) is minimized:

$$\mathcal{L}_{\text{task}}(\mathbf{s}; \mathcal{D}) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}} \left[\ell\big(y, \mathbf{s}(\mathbf{x})\big) \right]. \tag{3}$$

where each prediction is supervised by a hard label $y \in [K]$, assuming absolute class certainty. **Knowledge Distillation** enriches hard-label supervision with targets $\mathbf{z}_{\boldsymbol{\tau}}$ from a *teacher* $\boldsymbol{\tau}$ to minimize the distillation loss:

$$\mathcal{L}_{\text{KD}}(\mathbf{s}; \mathcal{D}) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}} \left[\ell\big(\mathbf{z}_\tau(\mathbf{x}), \mathbf{z}_s(\mathbf{x})\big) \right]. \tag{4}$$

Here, $\mathbf{z}_\tau$ may represent teacher logits (Hinton et al., 2015), intermediate features (Romero et al., 2015), or other induced representations (Park et al., 2019), with $\mathbf{z}_s$ denoting the corresponding student outputs and ℓ their discrepancy. Teacher targets encode class uncertainty, task-specific representations, and input relationships, leading to improved generalization bounds (Menon et al., 2021).

From Point Targets to Vicinal Supervision. Nevertheless, teacher supervision in KD remains fundamentally *point-wise*: for each input $\mathbf{x}$, supervision is confined to a single observed target $\mathbf{z}_\tau(\mathbf{x})$. However, since the teacher itself learns from finite data and inherits inductive biases and optimization limitations (Dong et al., 2019), a natural question is: *Is a single teacher target sufficiently reliable, and must supervision be confined to individual targets?* In response, we enrich single-teacher supervision by extending each individual target into a vicinal distribution $\nu(\cdot \mid \mathbf{z}_\tau(\mathbf{x}))$ around it:

$$\underbrace{\delta_{\mathbf{z}_\tau(\mathbf{x})}}_{\text{Point Target}} \quad \longrightarrow \quad \underbrace{\nu(\cdot \mid \mathbf{z}_\tau(\mathbf{x}))}_{\text{Vicinal distribution}}$$

See Fig. 1. Accordingly, for each input, the point-wise distillation loss in eq. (4) is replaced by the expected loss over its vicinal target distribution: $\ell\left(\mathbf{z}_\tau(\mathbf{x}), \mathbf{z}_s(\mathbf{x})\right) \longrightarrow \mathbb{E}_{\tilde{\mathbf{z}}_\tau \sim \nu(\cdot \mid \mathbf{z}_\tau(\mathbf{x}))} \left[\ell\left(\tilde{\mathbf{z}}_\tau, \mathbf{z}_s(\mathbf{x})\right) \right]$.

The underlying motivation is that an individual teacher target $\mathbf{z}_\tau$ may be imperfect due to inherited biases, memorization, or overconfidence (Lukasik et al., 2021). Probing its local vicinity $\nu(\cdot \mid \mathbf{z}_\tau)$ provides multiple complementary targets around $\mathbf{z}_\tau$, enabling more robust supervision to be identified (validated on misclassified samples and overfitted teachers in Sec. 3). This, in turn, strengthens teacher–student alignment by matching local regions beyond individual points (validated by improved fidelity in Sec. 3). As we show in Sec. 2.2.1, this shift to vicinal supervision admits a natural interpretation through Vicinal Risk Minimization, enabling the student to learn the local structure of the teacher representation manifold $\mathcal{M}_\tau$. The quality of this supervision consequently depends on how faithfully the chosen vicinal distribution respects the underlying structure of the teacher representation distribution—a property we henceforth refer to as *manifold-consistency*.

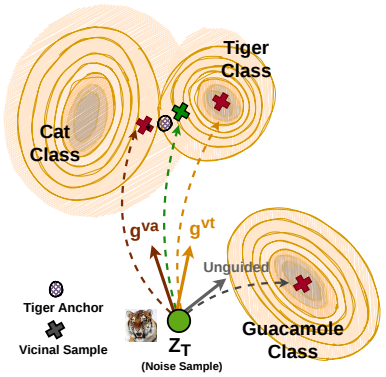
2.2 Generative Vicinity Distillation

The remaining challenge, therefore, is to construct $\nu(\cdot \mid \mathbf{z}_\tau)$ from which manifold-consistent vicinal targets can be sampled. Drawing insights from data augmentation, two common approaches emerge: 1) *Isotropic Gaussian perturbation* (Huang & Wei, 2022) to induce a Gaussian vicinity around each target: $\mathcal{N}(\mathbf{z}_\tau, \sigma^2\mathbf{I})$. 2) *Mixup interpolations* (Zhang et al., 2017) to construct synthetic targets by interpolating pairs $\mathbf{z}_\tau$ and their labels. Neither approach guarantees manifold-consistent targets. In high-dimensional representation spaces, only a small fraction of directions locally follow the manifold, while the remaining majority point away from it (Ansuini et al., 2019). Isotropic perturbations ignore this geometry, assigning equal variance in every direction and thus predominantly perturbing targets away from the manifold. Mixup likewise provides no guarantee that the generated samples remain on the manifold; interpolations may cross into regions of other classes, producing misrepresentative samples with conflicting labels—a phenomenon known as *manifold intrusion* (Baena et al., 2022).

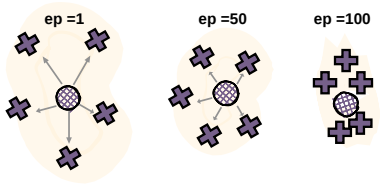
Manifold-Consistent Vicinities using a Score-Based Prior. Rather than using arbitrary perturbations, we seek to construct vicinal samples $\tilde{\mathbf{z}}_\tau \sim \nu(\cdot \mid \mathbf{z}_\tau)$ that remain plausible under the teacher representation distribution $p(\mathbf{z}_\tau)$. We propose *Generative Vicinity*, a novel vicinal learning approach that leverages a score-based generative prior learned over $p(\mathbf{z}_\tau)$ to generate manifold-consistent targets. Given teacher representations $\mathbf{z}_\tau \sim p(\mathbf{z}_\tau)$, we first learn the neural approximation of the score field of their distribution using a diffusion model: $\nabla_{\mathbf{z}_\tau} \log p_\phi(\mathbf{z}_\tau)$. Once learned, the score field assigns vectors throughout the representation space, pointing toward higher-density regions and, near the manifold, approximately toward the manifold itself (Song et al., 2020b; Stanczuk et al., 2022). During sampling, the reverse process $p_{\boldsymbol{\phi}}(\mathbf{z}_{\tau,t-1} \mid \mathbf{z}_{\tau,t})$ follows the learned score field to progressively transform Gaussian noise $\mathbf{z}_{\tau,T} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ into a plausible teacher target $\mathbf{z}_{\tau,0} \in \mathcal{M}_\tau$ through a sequence of small denoising corrections for $t = T, T-1, \ldots, 1$. Each denoising correction uses score vectors to suppress components of $\mathbf{z}_{\tau,t}$ that push it toward low-density regions, while reinforcing those that move it toward high-density regions of $p_\phi(\mathbf{z}_\tau)$. Consequently, when the reverse process is anchored to $\mathbf{z}_\tau$, the learned score prior regularizes the resulting perturbations toward plausible regions of the representation space, reducing off-distribution deviations (Song et al., 2020a). The sequence $\{\mathbf{z}_{\tau,T}, \mathbf{z}_{\tau,T-1}, \ldots, \mathbf{z}_{\tau,0}\}$ forms a reverse diffusion trajectory from Gaussian noise to a plausible representation. However, global plausibility is insufficient to construct a meaningful vicinity around a given anchor $\mathbf{z}_\tau$: an unconstrained reverse diffusion trajectory may converge to any plausible region of the teacher manifold, including regions far from the anchor or corresponding to a different semantic class. For example, in Fig. 2, an unconstrained trajectory from a *tiger* anchor may converge to an unrelated class such as *guacamole*. To prevent manifold intrusion and conflicting supervision, we seek to control the sampling to satisfy two properties: **1) Locality:** $\tilde{\mathbf{z}}_\tau \sim \nu(\cdot \mid \mathbf{z}_\tau)$ remains tied to the vicinity of its corresponding anchor $\mathbf{z}_\tau$; **2) Class consistency:** $\arg\max \boldsymbol{\tau}^{\text{tail}}(\tilde{\mathbf{z}}_\tau) = y$, preserving the target class of the anchor; where $\boldsymbol{\tau}^{\text{tail}}$ denotes the teacher tail, such that $\boldsymbol{\tau} = \boldsymbol{\tau}^{\text{tail}} \circ \boldsymbol{\tau}^{\text{feat}}$.

Controlling the Vicinity using Guidance. Accordingly, we modify the effective score field followed during reverse process using diffusion guidance. We design two guidance functions to provide gradient-based corrections to each transition $p_{\boldsymbol{\phi}}(\mathbf{z}_{\tau,t-1} \mid \mathbf{z}_{\tau,t})$, steering the trajectory toward desirable targets (see Fig. 2). **1) Vicinal anchor guidance $\mathbf{g}_t^{\text{va}}$ for locality:** For each $\mathbf{x}$, we refer to its observed teacher representation as the *anchor*, denoted by $\boldsymbol{a} = \mathbf{z}_\tau(\mathbf{x})$. We treat the predicted clean representation at timestep t, $\hat{\mathbf{z}}_{\tau,0}$ (in 1), as a proxy for the final converged representation $\mathbf{z}_{\tau,0}$, and guide it towards the corresponding $\boldsymbol{a}$: $\mathbf{g}_t^{\text{va}} = (\boldsymbol{a} - \hat{\mathbf{z}}_{\tau,0})$. This guidance is applied with guidance scale λ_t^{va}, as an augmentation to the original denoising prediction $\epsilon_{\boldsymbol{\phi}}(\mathbf{z}_{\tau,t}, t)$ at timestep t:

$$\tilde{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_{\tau,t}, t) = \epsilon_{\boldsymbol{\phi}}(\mathbf{z}_{\tau,t}, t) + \sqrt{1-\alpha_t}\ \lambda_t^{\text{va}}\, \mathbf{g}_t^{\text{va}} \tag{5}$$



(Left) Guided vs. unguided sampling. Unguided samples drift toward an incorrect nearby class (guacamole). Target guidance $\mathbf{g}^{\text{vt}}$ steers trajectories toward the correct class (tiger) but ignores locality. Anchor guidance $\mathbf{g}^{\text{va}}$ preserves locality but fails to correct class-level errors (cat). Joint guidance steers toward locally consistent and class-correct vicinal targets.



(Right) Progressive targets generated by refining vicinities.

Figure 2: Illustration of guided sampling and progressive supervision with refined vicinities.

Since negative predicted noise approximates the score function (up to a timestep-dependent scaling), nudging the predicted noise toward $\boldsymbol{a}$ biases the trajectory toward a local neighborhood of anchor $\boldsymbol{a}$.

2) Vicinal target guidance $\mathbf{g}_t^{\text{vt}}$ for class consistency: To preserve the target class, we incorporate gradients from the teacher tail $\boldsymbol{\tau}^{\text{tail}}$ and penalize the sample with cross-entropy w.r.t. the ground-truth: $\tilde{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_{\tau,t}, t) = \epsilon_{\boldsymbol{\phi}}(\mathbf{z}_{\tau,t}, t) + \sqrt{1-\alpha_t}\ \nabla_{\mathbf{z}_{\tau,t}} \ell_{\text{ce}}\big(y, \boldsymbol{\tau}^{\text{tail}}(\mathbf{z}_{\tau,t})\big)$. However, since $\boldsymbol{\tau}^{\text{tail}}$ is trained on clean teacher representations, it may fail to provide meaningful guidance on noisy representations from reverse steps. Therefore, we guide the clean representation to match the target class based on $\hat{\mathbf{z}}_{\tau,0}$ (in 1). We first estimate the change in clean representation space that enforces the target constraint: $\Delta\mathbf{z}_{\tau,0} = \lambda_t^{\text{vt}} \nabla_{\mathbf{z}_{\tau,0}} \ell_{\text{ce}}\big(y, \boldsymbol{\tau}^{\text{tail}}(\hat{\mathbf{z}}_{\tau,0})\big) = \lambda_t^{\text{vt}}\, \mathbf{g}_t^{\text{vt}}$ where λ_t^{vt} is the guidance scale. Then, we linearly translate $\Delta\mathbf{z}_{\tau,0}$ back to the noisy representation space of $\mathbf{z}_{\tau,t}$ by computing a guided denoising prediction $\tilde{\epsilon}$ that satisfies: $\mathbf{z}_{\tau,t} = \sqrt{\alpha_t}\,(\hat{\mathbf{z}}_{\tau,0} - \Delta\mathbf{z}_{\tau,0}) + \sqrt{1-\alpha_t}\ \tilde{\epsilon}$. Reusing 1, we can express $\tilde{\epsilon}$ as an augmentation of the original denoising prediction $\epsilon_{\boldsymbol{\phi}}(\mathbf{z}_{\tau,t}, t)$:

$$\tilde{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_{\tau,t}, t) = \epsilon_{\boldsymbol{\phi}}(\mathbf{z}_{\tau,t}, t) + \sqrt{\frac{\alpha_t}{1-\alpha_t}}\,\Delta\mathbf{z}_{\tau,0} = \epsilon_{\boldsymbol{\phi}}(\mathbf{z}_{\tau,t}, t) + \sqrt{\frac{\alpha_t}{1-\alpha_t}}\,\lambda_t^{\text{vt}}\,\mathbf{g}_t^{\text{vt}}. \tag{6}$$

Together these two guidance impose our desired target behavior: maximizing the ability to represent target-class while remaining within the local vicinity of the original representation. **Counterfactual Supervision When the Teacher Misclassifies:** Our target guidance conflicts with the teacher when its prediction disagrees with the ground truth, $\boldsymbol{\tau}^{\text{tail}}(\mathbf{z}_\tau) \neq y$, since guidance enforces preservation of y. Such errors can nevertheless be informative, as mispredictions typically correspond to semantically related classes (e.g., *tiger* misclassified as *cat*). In this case, $\mathbf{z}_\tau$ lies near the feature mode of the incorrect class. By generating vicinal targets around $\mathbf{z}_\tau$ and guiding them toward y, we construct *counterfactual teacher views*: features originating from the "cat" class that are steered to align with the semantics of the target class (e.g., "tiger"). These views provide vicinal targets that correct teacher errors, which we empirically validate in Sec. 3. We illustrate the joint effect of both guidance on trajectories in Fig. 2.

Progressive Teacher Supervision with Vicinal Dispersion. In KD, the target $\mathbf{z}_\tau$ remains fixed throughout student learning. Our ability to control the generated targets enables a further extension from fixed to progressive supervision, by regulating vicinal dispersion. We construct epoch-dependent vicinal distributions $\nu_{\boldsymbol{\phi}}^{(ep)}(\cdot \mid \mathbf{z}_\tau)$ that progressively transition from a broader to a narrower vicinity around the anchor to provide increasingly fine-grained supervision (see Fig. 2). Accordingly, we modulate the guidance strength: weaker guidance permits greater variation, producing a broader vicinity, while stronger guidance concentrates targets around the anchor. We define: (i) a cosine decay schedule of the guidance strengths λ_t^{va} and λ_t^{vt} over epochs:

$$\lambda(ep) = \lambda_{\text{end}} + \frac{1}{2}\,(\lambda_{\text{start}} - \lambda_{\text{end}})\left(1 + \cos\left(\frac{\pi(ep-1)}{E-1}\right)\right), \tag{7}$$

and (ii) an epoch-dependent schedule of $t_{\text{init}} \in [T]$:

$$t_{\text{init}}^{(ep)} = t_{\text{end}} + \frac{1}{2}\,(t_{\text{start}} - t_{\text{end}})\left(1 + \cos\left(\frac{\pi(ep-1)}{E-1}\right)\right), \tag{8}$$

to control the amount of information retained from anchor to initialize the reverse process: $\mathbf{z}_{\tau,t_{\text{init}}} \sim q(\mathbf{z}_{\tau,t_{\text{init}}} \mid \mathbf{z}_{\tau,0}) = \sqrt{\alpha_{t_{\text{init}}}}\,\mathbf{z}_\tau + \sqrt{1-\alpha_{t_{\text{init}}}}\,\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Higher t_{init} moves the initialization farther away from $\mathbf{z}_\tau$ and thus increases ambiguity. This way, early epochs impose weaker supervision constraints, allowing more flexible and student-friendly teacher targets.

Training. We propose to train the student by minimizing the *Generative Vicinal Distillation* loss:

$$\mathcal{L}_{\text{GVD}}(\mathbf{s}; \mathcal{D}) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\, \mathbb{E}_{\tilde{\mathbf{z}}_\tau \sim \nu_{\boldsymbol{\phi}}^{(ep)}(\cdot|\mathbf{z}_\tau)} \left[\ell(\tilde{\mathbf{z}}_\tau,\, \mathbf{z}_s)\right] \tag{9}$$

Practical implementation. In practice, we realize progressive vicinities $\nu_{\boldsymbol{\phi}}^{(ep)}(\cdot \mid \mathbf{z}_\tau)$ by sampling from the diffusion prior $p_{\boldsymbol{\phi}}(\cdot \mid \mathbf{c}^{(ep)})$ under epoch-dependent conditioning (eqs. (7) and (8)). The inner expectation is approximated using M vicinal samples per instance in a mini-batch $\mathcal{B}$:

$$\widehat{\mathcal{L}}_{\text{GVD}}(\mathbf{s}; \mathcal{B}) = \frac{1}{|\mathcal{B}|} \sum_{(\mathbf{x},y)\in\mathcal{B}} \frac{1}{M} \sum_{m=1}^{M} \ell\Big(\tilde{\mathbf{z}_\tau}^{(m)},\, \mathbf{z}_s\Big); \quad \tilde{\mathbf{z}_\tau}^{(m)} \sim \nu_{\boldsymbol{\phi}}^{(ep)}(. \mid \mathbf{z}_\tau) \;\approx\; p_{\boldsymbol{\phi}}(\cdot \mid \mathbf{c}^{(ep)}), \tag{10}$$

We jointly leverage supervision from vicinal features and their corresponding teacher-tail logits, using MSE for ℓ_{feat} and KL divergence for ℓ_{logit}:

$$\ell\Big(\tilde{\mathbf{z}_\tau}^{(m)},\, \mathbf{z}_s\Big) = \Big[\ell_{\text{feat}}\Big(\tilde{\mathbf{z}_\tau}^{(m)},\, \mathbf{z}_s\Big) + \ell_{\text{logit}}\Big(\boldsymbol{\tau}^{\text{tail}}(\tilde{\mathbf{z}_\tau}^{(m)}),\, \mathbf{s}^{\text{tail}}(\mathbf{z}_s)\Big)\Big] \tag{11}$$

2.2.1 Vicinal Teacher Supervision as Implicit Vicinal Risk Minimization

Under Empirical Risk Minimization (ERM), the task loss in eq. (3) can be viewed as an expectation over the empirical distribution $\mathcal{P}_{\text{emp}}(\mathbf{x}, y) = \frac{1}{|\mathcal{D}|}\sum_{(\mathbf{x}_i,y_i)\in\mathcal{D}} \delta_{\mathbf{x}_i}(\mathbf{x})\,\delta_{y_i}(y)$; a collection of delta functions centered at each data. Consequently, ERM constrains the model primarily at the observed data points, leaving it sensitive to small perturbations, particularly when the data are sparse or noisy. Vicinal Risk Minimization (VRM) (Chapelle et al., 2000) improves the empirical approximation by replacing each delta at $\mathbf{x}_i$ with a local density estimate $p_{\mathbf{x}_i}(\mathbf{x})$, that models its vicinity: $\mathcal{P}_{\text{est}}(\mathbf{x}, y) = \frac{1}{|\mathcal{D}|}\sum_{(\mathbf{x}_i,y_i)\in\mathcal{D}} p_{\mathbf{x}_i}(\mathbf{x})\,\delta_{y_i}(y)$. The resulting vicinal risk is: $\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{P}_{\text{est}}}\left[\ell\big(y, \mathbf{s}(\mathbf{x})\big)\right] = \frac{1}{|\mathcal{D}|}\sum_{(\mathbf{x}_i,y_i)\in\mathcal{D}} \int \ell\big(y_i, \mathbf{s}(\mathbf{x})\big) p_{\mathbf{x}_i}(\mathbf{x}) d\mathbf{x}$. The more faithfully $p_{\mathbf{x}_i}(\mathbf{x})$ captures the local structure of $\mathcal{P}$, the more effectively VRM promotes local smoothness and reduces sensitivity to noise and overfitting (Verma et al., 2019). Mixup is the most widely used VRM implementation for DNNs, but suffers from manifold intrusion. Manifold Mixup (Verma et al., 2019) partly mitigates this by interpolating in learned representation spaces, where semantically similar samples are closer. Nevertheless, linear interpolation does not guarantee manifold-preserving samples, leaving intrusion possible Baena et al. (2022). This work addresses this limitation by moving from predefined to learned vicinal geometry. We rely on an expert model (teacher) to parameterize the data manifold through its learned representation space. We then learn this representation distribution using a score-based generative model and leverage its score field to construct manifold-consistent vicinal distributions. Guidance is introduced to prevent manifold intrusion during sampling. Using generated vicinal distributions as proxies for local data vicinities, we train the student model. Under our KD objective (eq. (9)), the student head is enforced to learn a representation $\mathbf{z}_s(\mathbf{x})$ that remains consistent

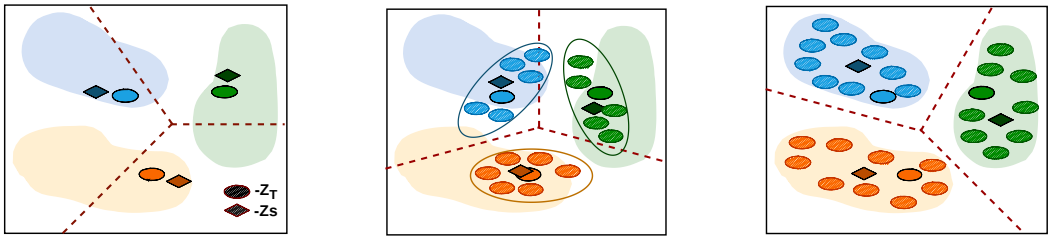


Figure 3: Let each colored blob represent the true local distribution around an instance. (left): KD aligns $\mathbf{z}_s$ with a single teacher target $\mathbf{z}_\tau$. When $\mathbf{z}_\tau$ fails to capture the true local distribution, the student overfits to these imperfect targets, resulting in poor decision boundaries. (center): Samples from a simple parametric vicinal distribution slightly improve the placement of $\mathbf{z}_s$. (right): Vicinal targets that adequately represent the local structure guide $\mathbf{z}_s$ toward optimal placement, improving boundaries.

Table 1: Predictive performance on CIFAR-100. We report Top-1, Top-5 accuracy, and F1-score.

	Homogeneous Architecture Distillation									Heterogeneous Architecture Distillation								
Teacher	WRN-40-2			ResNet-32x4			ResNet-56			ResNet-50			ResNet-32x4			ResNet-32x4		
Student	WRN-40-1			ResNet-8x4			ResNet-20			MobileNetV2			ShuffleNetV1			ShuffleNetV2		
	Top-1	Top-5	F1	Top-1	Top-5	F1	Top-1	Top-5	F1	Top-1	Top-5	F1	Top-1	Top-5	F1	Top-1	Top-5	F1
Base	72.84	92.68	72.83	72.52	92.26	72.51	69.87	92.00	69.85	62.00	86.01	61.77	70.90	91.11	70.78	72.88	91.71	72.73
KD	74.40	93.01	74.37	74.02	93.03	74.04	71.43	92.65	71.47	65.24	87.71	65.00	74.26	92.71	74.19	76.50	94.15	77.41
FitNets	74.31	93.54	74.32	73.38	93.29	73.36	70.77	92.73	70.73	63.45	87.27	63.20	71.73	91.51	71.61	73.19	91.90	73.08
DIST	73.23	92.72	73.26	73.26	92.37	73.23	70.86	92.14	70.85	61.37	85.71	61.19	71.99	91.74	71.95	73.44	92.36	73.37
DiffKD	72.99	92.77	72.95	72.79	92.20	72.76	70.37	91.98	70.34	62.33	86.40	62.06	71.10	91.02	71.01	73.24	92.16	73.12
TeKAP	74.52	93.63	74.51	73.57	93.14	73.59	71.07	92.79	71.02	62.75	86.85	62.55	71.04	91.17	70.92	72.90	91.71	72.81
GVD	**75.74**	**94.03**	**75.70**	**74.65**	**93.36**	**74.65**	**71.52**	**92.92**	**71.50**	**67.22**	**89.47**	**66.91**	**74.38**	**92.92**	**74.29**	**77.59**	**94.22**	**77.51**

across all vicinal targets corresponding to $\mathbf{x}$. This encourages consistent predictions across local neighborhoods, promoting local smoothness in line with VRM. From this perspective, our formulation can be seen as a novel, teacher-induced form of VRM: expert teacher transfers knowledge of the local distributions surrounding individual data, through its representation space to a less capable student. Such local structure cannot be transferred using an individual teacher representation, as in conventional KD, which uses a degenerate vicinal distribution that collapses to a Dirac delta at observed target: $\delta(\cdot \mid \mathbf{z}_\tau(\mathbf{x}))$. In Fig. 3, we provide a mental image of how GVD benefits from matching a vicinity.

3 Experiments

We conduct our experiments on image benchmarks (CIFAR-100 (Krizhevsky et al., 2009), CIFAR-10 (Krizhevsky et al., 2009), TinyImageNet (Deng et al., 2009), and STL-10 (Coates et al., 2011)) and time-series benchmarks (UCR archive (Dau et al., 2019)). All students follow the same training protocol, with results averaged over five runs. Full experimental details are provided in Sec. 5.2.

Performance on Image Classification. We compare GVD against logit-based KD (Hinton et al., 2015), feature distillation (FitNets) (Romero et al., 2015), multi-teacher distillation (TeKAP) (Hossain et al., 2025), distillation from stronger teacher (DIST) (Huang et al., 2022), and diffusion-based distillation (DiffKD) (Huang et al., 2023), along with a student (Base) trained without distillation. On CIFAR-100 (Tab. 1), GVD achieves the best performance across both homogeneous and heterogeneous teacher–student pairs and metrics, with the largest Top-1 gain for ResNet-50→MobileNetV2 (+1.98 over the strongest baseline), and consistently outperforms noise-augmented multi-teacher distillation (TeKAP). Similar gains are observed on TinyImageNet (Tab. 2) and CIFAR-10 (Tab. 10).

Performance on Time Series Classification (TSC). We evaluate GVD on 20 UCR datasets (Dau et al., 2019) in Tab. 4 (full results in Sec. 5.3). GVD achieves the highest average accuracy and lowest average rank across all settings, outperforming each baseline on at least 75% of the datasets.

GVD Improves Teacher–Student Fidelity. Rather than aligning with a single teacher target, GVD reinforces teacher–student alignment with multiple generated targets, which we expect to improve fidelity—how well the student replicates the teacher. We evaluate fidelity using Top-1/Top-5 prediction agreement, and predictive KL divergence (Stanton et al., 2021). On TinyImageNet (Tab. 3), GVD achieves the highest Top-1 and Top-5 agreement for both teacher–student pairs, while achieving the lowest or competitive KLD. Similar improvements are observed on CIFAR-100 (Tab. 13) and CIFAR-10 (Tab. 14), demonstrating more faithful transfer of teacher behavior.

GVD Improves Calibration. Training on multiple nearby teacher targets encourages locally smooth student outputs that are less sensitive to perturbations. We hypothesize that this reduced sensitivity may mitigate overconfidence and consequently improve calibration. Reliability diagrams in Fig. 4 support this, with GVD exhibiting the closest alignment to the diagonal, indicating improved confidence calibration compared to other students. We further assess this using Expected Calibration Error (ECE), Brier score, and test NLL (Guo et al., 2017) on CIFAR-100 (Table 15), where GVD achieves the lowest ECE and Brier in all settings and the lowest NLL in 5/6 settings. We visualize how GVD improves PCA feature organization and reduces overconfidence in ambiguous regions (see Fig. 5).

Counterfactual Supervision Helps. Unlike conventional feature augmentation, Generative Vicinity enables guided generation with desired properties, allowing generated targets to even surpass the original teacher target on specific measures. For example, for teacher-misclassified instances, $\{\boldsymbol{\tau}^{\text{tail}}(\mathbf{z}_\tau) \neq y\}$, we can refine vicinal features by guiding them toward the ground-truth. To evaluate this, we

Table 2: Evaluation results on TinyImageNet.

	Metric	Base	KD	FitNet	DIST	DiffKD	TeKAP	GVD
ResNet-18 (ResNet-34)	Top-1	59.94	63.47	62.54	61.56	60.82	62.64	**64.07**
	Top-5	80.76	83.75	83.23	81.75	81.38	83.17	**84.60**
	F1-score	59.80	63.25	62.39	61.45	60.68	62.55	**63.89**
	AUC-PRC	64.37	68.56	67.39	66.05	65.22	67.34	**68.88**
MobileNetV1 (ResNet-50)	Top-1	58.43	60.79	58.50	59.20	59.06	58.38	**61.24**
	Top-5	80.03	81.74	80.25	80.54	80.64	80.25	**82.59**
	F1-score	58.17	60.04	58.29	58.95	58.86	58.19	**60.96**
	AUC-PRC	62.64	65.33	62.91	63.55	63.49	62.69	**65.73**

Table 3: Student fidelity on TinyImageNet.

	Metric	Base	KD	FitNet	DIST	DiffKD	TeKAP	GVD
ResNet-18 (ResNet-34)	Top1A	63.04	65.67	65.56	64.41	63.69	65.68	**66.53**
	Top5A	85.20	88.10	87.52	86.00	85.58	87.41	**88.31**
	KLD	0.66	0.51	0.58	0.66	0.64	0.58	**0.49**
MobileNetV1 (ResNet-50)	Top1A	60.33	63.99	60.99	61.81	61.13	60.97	**64.46**
	Top5A	83.39	87.18	83.74	84.26	84.04	83.64	**87.37**
	KLD	0.84	0.54	0.82	0.80	0.81	0.83	**0.57**

Table 4: Summary evaluation results on 20 UCR datasets under three distillation settings.

	LSTM-100-3 → LSTM-8-1					ResNet-64-3 → ResNet-8-3					Inception-32-6 → Inception-16-3				
	Base	KD	Fits	TeKAP	GVD	Base	KD	Fits	TeKAP	GVD	Base	KD	Fits	TeKAP	GVD
Avg. Top-1 Acc.	54.58	54.99	58.99	58.92	**59.31**	89.84	90.69	90.96	90.90	**91.37**	85.92	86.95	87.03	86.80	**87.60**
Avg. Rank	4.20	3.45	2.70	2.60	**1.45**	4.35	2.95	2.75	2.65	**1.35**	3.45	2.95	3.10	3.15	**1.95**
Top-1 Wins	1	4	3	2	**14**	1	4	3	3	**14**	2	3	3	3	**11**
Wins/Draws	19	17	18	17	-	20	17	18	18	-	16	15	15	15	-
Losses	1	2	2	3	-	0	3	1	2	-	4	5	5	5	-

Table 5: Guided feature improvements for teacher-misclassified samples on CIFAR-100, under mild and strong target guidance. Each cell reports: mean gain (percentage of samples improved).

	Mild Guidance ($\lambda_t^{\text{vt}} = 1.0$)			**Strong Guidance** ($\lambda_t^{\text{vt}} = 5.0$)		
	Δp	ΔKL	$\Delta\cos$	Δp	ΔKL	$\Delta\cos$
w.r.t. Original Teacher Feature	**0.073** (91.36%)	**0.590** (91.36%)	**0.025** (100%)	**0.263** (100%)	**2.180** (100%)	**0.022** (100%)
w.r.t. Unguided Vicinal Feature	**0.074** (100%)	**0.929** (100%)	**0.009** (100%)	**0.263** (100%)	**2.319** (100%)	**0.017** (100%)

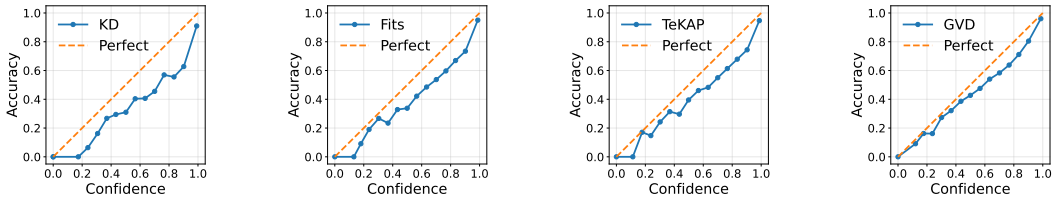


Figure 4: Reliability diagrams for CIFAR-100 students: (a) KD, (b) FitNets, (c) TeKAP, (d) GVD.

study generated targets on CIFAR-100 using a WRN-40-2 teacher and its learned prior. We compare guided features, $\mathbf{z}_{\text{gd}}$, with reference features, $\mathbf{z}_{\text{ref}}$, representing either the original teacher features or unguided vicinal features. For each sample, we report: **i)** target-class probability gain, $\Delta p = p_\tau(y \mid \mathbf{z}_{\text{gd}}) - p_\tau(y \mid \mathbf{z}_{\text{ref}})$, where $\Delta p > 0$ indicates improved alignment with the ground-truth; **ii)** hard-label KL reduction, $\Delta\text{KL} = \text{KL}\big(\mathbf{y} \,\|\, \mathbf{p}_\tau(\mathbf{z}_{\text{ref}})\big) - \text{KL}\big(\mathbf{y} \,\|\, \mathbf{p}_\tau(\mathbf{z}_{\text{gd}})\big)$, where positive values indicate increased confidence in the correct class; and **iii)** cosine similarity gain to the class prototype, $\Delta\cos = \cos(\mathbf{z}_{\text{gd}}, \boldsymbol{\mu}_y) - \cos(\mathbf{z}_{\text{ref}}, \boldsymbol{\mu}_y)$, where positive values indicate movement toward the ground-truth prototype $\boldsymbol{\mu}_y$, defined as the mean feature of class y. Table 5 reports the mean gain and percentage of samples improved for each metric. As shown, mild guidance improves target-class confidence for over 91% of teacher-misclassified samples and moves all generated features closer to the ground-truth class prototype. Stronger guidance extends these gains to all samples, demonstrating that erroneous teacher targets can be refined into more class-consistent supervision, through improved feature placement.

Students on Teacher-Misclassified Samples. To assess the benefit of counterfactual targets that correct teacher errors, we evaluate distilled students on the teacher-misclassified subset. Table 6 compares GVD with reference students using hard-label KL divergence, $\text{KL}(\mathbf{y}|\mathbf{p})$. We compute the per-sample improvement as $\Delta\text{KL} = \text{KL}(\mathbf{y} \,\|\, \mathbf{p}_{\text{ref}}) - \text{KL}(\mathbf{y} \,\|\, \mathbf{p}_{\text{GVD}})$, where $\Delta\text{KL} > 0$ indicates better GVD predictions than the reference student. We report the mean ΔKL and the percentage of improved samples. GVD improves predictions over all reference models, including the teacher, with positive KL gains for 60.66–68.80% of misclassified samples. These improvements translate to higher Top-1 accuracy, with gains of 1.96–3.00% over baseline KD students on the teacher-misclassified set.

Additional Robustness Experiments. We further evaluate GVD under imperfect teacher supervision, limited distillation data, and cross-dataset transfer (Sec. 5.3.1). GVD is robust to overfitted teacher

targets and limited data, with larger gains in low-data regimes. GVD also transfers better from CIFAR-100 to STL-10 under both linear probing and fine-tuning, indicating generalizable representations.

Table 6: GVD performance on teacher-misclassified CIFAR-100 samples: mean hard-label KL reduction (ΔKL), percentage of samples improved (%), and Top-1 accuracy gain over each method.

	Teacher	**KD**	**Fits**	**TeKAP**
ΔKL (mean)	**1.135**	**1.168**	**0.386**	**0.371**
% (KL improved)	**68.80**	**60.66**	**65.89**	**63.93**
Top-1 Accuracy gain (%)	**30.15**	**2.18**	**1.96**	**3.00**

3.1 Ablations and Hyperparameter Analysis.

Vicinal Targets vs. Individual Teacher Targets. We conduct our ablation studies on CIFAR-100 by distilling WRN-40-2 into WRN-40-1. Replacing vicinal features and logits in eq. (11) with their fixed teacher counterparts reduces accuracy from 75.74 to 74.45. This combination of fixed feature and logit targets provides little improvement over either alone (74.31 for FitNets and 74.40 for KD). In contrast, vicinal supervision improves over the combined fixed-target setting, demonstrating that the gains stem from vicinal supervision rather than simply combining multiple teacher signals.

GVD vs. Mixup and Gaussian Vicinals. To test whether the gains are specific to GVD vicinals, we replace them with alternative vicinal constructions. Gaussian perturbations ($\sigma = 0.1$) and Mixup ($\alpha = 0.4$) achieve 74.43 and 73.22, respectively, both below GVD (75.74), highlighting the importance of class-preserving, manifold-consistent vicinal targets.

GVD Guidance Terms. Removing both guidance constraints ($t_{\text{init}} = T,\ \lambda = 0$) reduces accuracy to 64.54, highlighting the importance of controlled target generation. Target guidance alone ($t_{\text{init}} = T,\ \lambda : 0.3 \rightarrow 0.5$) provides only a marginal recovery to 65.61, whereas anchor guidance alone achieves 75.17, indicating anchor guidance is critical for preserving useful local supervision.

Effect of Progressive Targets. Fixing both guidance strengths and reverse step initialization ($\lambda = 0.3,\ t_{\text{init}} = 30$) reduces accuracy from 75.74 to 75.48. Scheduling only $t_{\text{init}} : 50 \rightarrow 30$ achieves 75.68, while scheduling only $\lambda : 0.3 \rightarrow 0.5$ achieves 75.51. These results show that progressive supervision improves performance, with t_{init} scheduling providing the larger contribution.

Effect of Progressive Schedule Strengths. We ablate the schedule strength of λ and t_{init}. For λ, aggressive (0.1–10), moderate (0.3–0.5), and mild (0.01–0.1) schedules achieve 73.22, 75.51, and 74.11, respectively, with $t_{\text{init}} = 30$. For t_{init}, aggressive (600–30), moderate (50–30), and mild (10–1) schedules achieve 74.01, 75.68, and 74.38, respectively, with $\lambda = 0.3$. Moderate schedules perform best for both parameters, indicating that overly weak or aggressive progression degrades supervision.

Effect of loss terms: Task loss (72.84) and GVD loss (72.63) alone underperform their combination (75.74). With task loss retained, adding only vicinal feature loss (in eq. (11)) improves performance to 75.56, and adding only vicinal logit loss yields 74.61, both below the combination (75.74), showing that the two vicinal signals contribute complementarily. We further conduct **additional ablations** to study the cost–performance trade-off of **number of vicinal samples** M (eq. (10)) and **diffusion parameters**, with full results and discussion provided in Sec. 5.3.2.

Computational Cost Analysis. GVD introduces a moderate training overhead due to diffusion-based vicinal sampling ($\approx$ 0.05 s/step over FitNets), while remaining competitive with diffusion-based KD methods (e.g., DiffKD) and using less memory. The training cost increases with the number of inference steps and vicinal samples, but performance gains quickly saturate, leading to a favorable accuracy–efficiency trade-off. Importantly, all additional computation is confined to training, leaving inference unchanged (0.004 s per batch). Detailed stage-wise cost analyses are provided in Sec. 5.3.2.

4 Conclusion

This work investigates whether vicinal distributions around teacher targets can improve student learning beyond point-wise supervision. We answer this affirmatively by proposing a novel distillation objective and demonstrating through extensive experiments across datasets and architectures that generative vicinal supervision improves student generalization and robustness. In seeking to extend single-teacher supervision, this work introduces a principled framework for controllable augmentation using score-based generative priors and guidance functions with applications beyond KD.

REPRODUCIBILITY STATEMENT

The source code for all models, training scripts, and experiments is available at `https://github.com/anonymKD/AnonymKD2026.git`. Details of datasets, preprocessing steps, model configurations, and training procedures are provided in the main text (Section 3) and Appendix (Section 5.2). Additional results are included in the supplementary materials to further support reproducibility.

REFERENCES

Sungsoo Ahn, Shell Xu Hu, Andreas Damianou, Neil D Lawrence, and Zhenwen Dai. Variational information distillation for knowledge transfer. In *CVPR*, pp. 9163–9171, 2019.

Zeyuan Allen-Zhu and Yuanzhi Li. Towards understanding ensemble, knowledge distillation and self-distillation in deep learning. *arXiv preprint arXiv:2012.09816*, 2020.

Alessio Ansuini, Alessandro Laio, Jakob H Macke, and Davide Zoccolan. Intrinsic dimension of data representations in deep neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

Raphael Baena, Lucas Drumetz, and Vincent Gripon. Preventing manifold intrusion with locality: Local mixup. *arXiv preprint arXiv:2201.04368*, 2022.

Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *SIGKDD*, pp. 535–541, 2006.

Olivier Chapelle, Jason Weston, Léon Bottou, and Vladimir Vapnik. Vicinal risk minimization. *Advances in neural information processing systems*, 13, 2000.

Jang Hyun Cho and Bharath Hariharan. On the efficacy of knowledge distillation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4794–4802, 2019.

Hongjun Choi, Eun Som Jeon, Ankita Shukla, and Pavan Turaga. Understanding the role of mixup in knowledge distillation: An empirical study. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 2318–2327. IEEE, 2023.

Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 215–223. JMLR Workshop and Conference Proceedings, 2011.

Hoang Anh Dau, Anthony Bagnall, Kaveh Kamgar, Chin-Chia Michael Yeh, Yan Zhu, Shaghayegh Gharghabi, Chotirat Ann Ratanamahatana, and Eamonn Keogh. The ucr time series archive. *IEEE/CAA Journal of Automatica Sinica*, 6(6):1293–1305, 2019.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Bin Dong, Jikai Hou, Yiping Lu, and Zhihua Zhang. Distillation $\approx$ early stopping? harvesting dark knowledge utilizing anisotropic information retrieval for overparameterized neural network. *arXiv preprint arXiv:1910.01255*, 2019.

Tommaso Furlanello, Zachary Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born again neural networks. In *ICML*, pp. 1607–1616. PMLR, 2018.

W Brier Glenn et al. Verification of forecasts expressed in terms of probability. *Monthly weather review*, 78(1):1–3, 1950.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pp. 1321–1330. PMLR, 2017.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Md Imtiaz Hossain, Sharmen Akhter, Choong Seon Hong, and Eui-Nam Huh. Single teacher, multiple perspectives: Teacher knowledge augmentation for enhanced knowledge distillation. In *The Thirteenth International Conference on Learning Representations*, 2025.

Long-Kai Huang and Ying Wei. Improving task-specific generalization in few-shot learning via adaptive vicinal risk minimization. *Advances in Neural Information Processing Systems*, 35: 3329–3342, 2022.

Tao Huang, Shan You, Fei Wang, Chen Qian, and Chang Xu. Knowledge distillation from a stronger teacher. *Advances in neural information processing systems*, 35:33716–33727, 2022.

Tao Huang, Yuan Zhang, Mingkai Zheng, Shan You, Fei Wang, Chen Qian, and Chang Xu. Knowledge diffusion for distillation. *Advances in Neural Information Processing Systems*, 36:65299–65316, 2023.

Guangda Ji and Zhanxing Zhu. Knowledge distillation in wide neural networks: Risk bound, data efficiency and imperfect teacher. *Advances in Neural Information Processing Systems*, 33: 20823–20833, 2020.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Tom A Lamb, Rudy Brunel, Krishnamurthy DJ Dvijotham, M Pawan Kumar, Philip HS Torr, and Francisco Eiras. Faithful knowledge distillation. *arXiv preprint arXiv:2306.04431*, 2023.

Michal Lukasik, Srinadh Bhojanapalli, Aditya Krishna Menon, and Sanjiv Kumar. Teacher's pet: understanding and mitigating biases in distillation. *arXiv preprint arXiv:2106.10494*, 2021.

Aditya K Menon, Ankit Singh Rawat, Sashank Reddi, Seungyeon Kim, and Sanjiv Kumar. A statistical perspective on distillation. In *International Conference on Machine Learning*, pp. 7632–7642. PMLR, 2021.

Seyed Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. Improved knowledge distillation via teacher assistant. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 5191–5198, 2020.

Luis A Ortega, Rafael Cabañas, and Andres Masegosa. Diversity and generalization in neural network ensembles. In *International Conference on Artificial Intelligence and Statistics*, pp. 11720–11743. PMLR, 2022.

Amin Parchami-Araghi, Moritz Böhle, Sukrut Rao, and Bernt Schiele. Good teachers explain: Explanation-enhanced knowledge distillation. In *European Conference on Computer Vision*, pp. 293–310. Springer, 2024.

Dae Young Park, Moon-Hyun Cha, Daesin Kim, Bohyung Han, et al. Learning student-friendly teacher networks for knowledge distillation. *Advances in neural information processing systems*, 34:13292–13303, 2021.

Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *CVPR*, pp. 3967–3976, 2019.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *NeurIPS*, 32, 2019.

Zengyu Qiu, Xinzhu Ma, Kunlin Yang, Chunya Liu, Jun Hou, Shuai Yi, and Wanli Ouyang. Better teacher better student: Dynamic prior knowledge for knowledge distillation. *arXiv preprint arXiv:2206.06067*, 2022.

Jun Rao, Xv Meng, Liang Ding, Shuhan Qi, Xuebo Liu, Min Zhang, and Dacheng Tao. Parameter-efficient and student-friendly knowledge distillation. *IEEE Transactions on Multimedia*, 26: 4230–4241, 2023.

Herbert E Robbins. An empirical bayes approach to statistics. In *Breakthroughs in Statistics: Foundations and basic theory*, pp. 388–394. Springer, 1992.

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets, 2015.

Adi Shamir, Odelia Melamed, and Oriel BenShmuel. The dimpled manifold model of adversarial examples in machine learning. *arXiv preprint arXiv:2106.10151*, 2021.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pp. 2256–2265. pmlr, 2015.

Wonchul Son, Jaemin Na, Junyong Choi, and Wonjun Hwang. Densely guided knowledge distillation using multiple teacher assistants. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9395–9404, 2021.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020a.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020b.

Jan Stanczuk, Georgios Batzolis, Teo Deveney, and Carola-Bibiane Schönlieb. Your diffusion model secretly knows the dimension of the data manifold. *arXiv preprint arXiv:2212.12611*, 2022.

Samuel Stanton, Pavel Izmailov, Polina Kirichenko, Alexander A Alemi, and Andrew G Wilson. Does knowledge distillation really work? *NeurIPS*, 34:6906–6919, 2021.

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*, pp. 6438–6447. PMLR, 2019.

Jiaqi Wen and Jianyi Yang. Distributionally robust optimization via generative ambiguity modeling. *arXiv preprint arXiv:2602.08976*, 2026.

Zhi-Qin John Xu, Yaoyu Zhang, Tao Luo, Yanyang Xiao, and Zheng Ma. Frequency principle: Fourier analysis sheds light on deep neural networks. *arXiv preprint arXiv:1901.06523*, 2019.

Shan You, Chang Xu, Chao Xu, and Dacheng Tao. Learning from multiple teacher networks. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1285–1294, 2017.

Seonghoon Yu, Dongjun Nam, Dina Katabi, and Jeany Son. Single-teacher view augmentation: Boosting knowledge distillation via angular diversity. *arXiv preprint arXiv:2510.22480*, 2025.

Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *arXiv preprint arXiv:1612.03928*, 2016.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

Ying Zhang, Tao Xiang, Timothy M Hospedales, and Huchuan Lu. Deep mutual learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4320–4328, 2018.

5 APPENDIX

5.1 RELATED WORK

Knowledge Distillation (KD), first introduced in Buciluǎ et al. (2006); Hinton et al. (2015), shows that transferring knowledge from larger models can help smaller models achieve comparable or even superior predictive performance. Generally, KD is implemented by matching the teacher's softened logits with those of the student, using a temperature hyperparameter to amplify the contribution of negative logits. Subsequent works extend teacher targets beyond logits to include intermediate features Romero et al. (2015); Zagoruyko & Komodakis (2016); Ahn et al. (2019) and correlations among outputs Park et al. (2019); Huang et al. (2022), establishing state-of-the-art results.

However, recent works observe that KD often fails to live up to this conventional understanding: **1) When the supervision provided by a single teacher is limited:** the teacher can be overfitted, providing no better supervision than hard labels Dong et al. (2019), and the knowledge from a single teacher is not diverse enough to capture multiple views of the data Allen-Zhu & Li (2020); Hossain et al. (2025). To address this, existing works propose teacher ensembles (Allen-Zhu & Li, 2020; You et al., 2017) or mutual supervision from student ensembles (Zhang et al., 2018; Furlanello et al., 2018). More recent work Hossain et al. (2025) proposes to augment single-teacher supervision by generating multiple augmented teacher targets through perturbing features with random noise, thereby increasing the robustness of supervision while avoiding the cost of retraining multiple models. **2) When teacher targets from the fine-tuned phase do not match the capacity of the student in its initial, nascent phase:** To address this, recent studies have proposed incorporating targets from intermediate "teacher–assistant" models Mirzadeh et al. (2020); Son et al. (2021), student-friendly teachers (Park et al., 2021; Rao et al., 2023; Cho & Hariharan, 2019), and early checkpoints Cho & Hariharan (2019). However, these methods provide simpler teacher targets at the cost of losing the expressive power of a large teacher. Moreover, the progression of teacher targets is defined in a few discrete stages that remain fixed throughout student training. **3) When the student fails to inherit the robustness of the teacher:** Recent works have observed that KD often transfers limited knowledge, leading students to learn predictive distributions that differ significantly from those of the teacher, thereby hindering their safe substitution (Stanton et al., 2021; Lamb et al., 2023). To address this, prior studies have proposed transferring more informative supervisory signals beyond logits or features to improve teacher–student fidelity (Parchami-Araghi et al., 2024; Lamb et al., 2023). However, these approaches remain grounded in empirical, point-wise teacher targets. In contrast, we extend supervision beyond individual teacher targets by modeling teacher knowledge as a generative distribution, enforcing teacher–student fidelity across a neighborhood of plausible teacher behaviors.

Diffusion Models for Knowledge Distillation. The use of diffusion models in knowledge distillation remains limited. A recent approach, Diffusion-KD Huang et al. (2023), employs a diffusion model to denoise and refine student features before aligning them with teacher representations, thereby focusing on improving the student representation prior to distillation. In contrast, the generative capability of diffusion models for representing and augmenting *teacher knowledge* remains largely unexplored. GVD takes this direction by learning a diffusion-based generative prior over the teacher representation space and sampling controlled vicinal targets for student supervision. Thus, rather than using diffusion to refine student features, we exploit its generative capacity to expand teacher supervision from individual targets to distributions around them.

Data Augmentation in Knowledge Distillation. Data augmentation techniques such as Mixup have been studied in the context of KD, by modifying the data used to train the teacher. For example, Choi et al. (2023) study how training a teacher with Mixup affects subsequent distillation, showing that Mixup-trained teachers produce smoother predictions and that temperature scaling during KD may further over-smooth them. To the best of our knowledge, we are the first to reformulate teacher supervision from individual targets to teacher-induced vicinal distributions around each target. This formulation introduces a teacher-induced form of Vicinal Risk Minimization, where the student learns from local distributions of teacher targets rather than individual point-wise supervision.

5.2 EXPERIMENTAL SETUP

Datasets. For image classification, we evaluate on standard KD benchmarks: **(1) CIFAR-100** Krizhevsky et al. (2009): 100 classes, 50K training and 10K test images of size 32×32.

(2) CIFAR-10 Krizhevsky et al. (2009): 10 classes, 50K training and 10K test images of size 32×32. **(3) STL-10** Coates et al. (2011): 10 classes, 5K training and 8K test images of size 96×96. **(4) TinyImageNet** Deng et al. (2009): 200 classes, each with 500 training, 50 validation, and 50 test images of size 64×64. For time-series classification, we evaluate on 20 datasets from the **UCR-2015 archive** (Dau et al., 2019) (see Tab. 7). All series are resampled to length 100 via linear interpolation, z-normalized, and evaluated using the original train/test split, with 20% of the training set used for validation.

Table 7: Summary of univariate UCR benchmark datasets used in our experiments.

Dataset	Type	Train	Test	Variables	Seq. Length	Classes
Computers	Device	250	250	1	720	2
UWaveGestureLibraryAll	Motion	896	3582	1	945	8
Strawberry	Spectro	613	370	1	235	2
BeetleFly	Image	20	20	1	512	2
Wafer	Sensor	1000	6164	1	152	2
Lightning2	Sensor	60	61	1	637	2
ItalyPowerDemand	Sensor	67	1029	1	24	2
Yoga	Image	300	3000	1	426	2
Trace	Sensor	100	100	1	275	4
ShapesAll	Image	600	600	1	512	60
MiddlePhalanxOutlineCorrect	Image	600	291	1	80	2
SwedishLeaf	Image	500	625	1	128	15
FaceAll	Image	560	1690	1	131	14
StarLightCurves	Sensor	1000	8236	1	1024	3
ECG200	ECG	100	100	1	96	2
MoteStrain	Sensor	20	1252	1	84	2
SonyAIBORobotSurfaceII	Sensor	27	953	1	65	2
Ham	Spectro	109	105	1	431	2
NonInvasiveFetalECGThorax1	ECG	1800	1965	1	750	42
NonInvasiveFetalECGThorax2	ECG	1800	1965	1	750	42

Evaluation Metrics. Model performance is primarily evaluated using the following metrics (reported in tables using the abbreviations in bold):

- **Top-1:** Accuracy (top prediction is correct).
- **Top-5:** Fraction of samples where the true label is among the top-5 predictions).
- **F1:** F1-score (harmonic mean of precision and recall, $\frac{2PR}{P+R}$).
- **AUC-PRC:** Area under the precision–recall curve, averaged across classes.

Model calibration is evaluated using:

- **ECE:** Expected Calibration Error Guo et al. (2017); Stanton et al. (2021), $\mathrm{ECE} = \sum_{b=1}^{B} \frac{|S_b|}{N} \left|\mathrm{acc}(S_b) - \mathrm{conf}(S_b)\right|$, where S_b denotes samples in bin b, and acc and conf are accuracy and confidence.
- **Brier:** Brier Score Glenn et al. (1950). Mean squared error between predicted probabilities $\mathbf{p}_i$ and one-hot labels $\mathbf{y}_i$, $\mathrm{Brier} = \frac{1}{N} \sum_{i=1}^{N} \|\mathbf{p}_i - \mathbf{y}_i\|_2^2$.
- **NLL:** Negative log-likelihood Guo et al. (2017); Stanton et al. (2021) (equivalently, cross-entropy loss) computed on the test set, $\mathrm{NLL} = -\frac{1}{N} \sum_{i=1}^{N} \log p(y_i \mid \mathbf{x}_i)$.

Teacher–student fidelity is evaluated using:

- **Top1A:** Top1 Agreement Stanton et al. (2021). $\frac{1}{N} \sum_{i=1}^{N} \mathbf{1}\{y_{\tau,i} = y_{s,i}\}$.
- **Top5A:** Top-5 Agreement. Fraction of samples where the teacher's top prediction lies within the student's top-5 predictions.
- **KLD:** Average predictive KL divergence Stanton et al. (2021), $\frac{1}{N} \sum_{i=1}^{N} KL\big(\boldsymbol{p}_\tau(\mathbf{x}_i) \,\|\, \boldsymbol{p}_s(\mathbf{x}_i)\big)$.

For time-series classification across multiple datasets, we employ a win/draw/loss evaluation. A method is considered to *win* on a dataset if it achieves the highest Top-1 accuracy.

- **Avg. Top-1 Acc.:** Mean Top-1 accuracy across all datasets.

- **Avg. Rank:** Average rank of each method across datasets (lower is better).
- **Top-1 Wins:** Number of datasets where the method achieves the best Top-1 accuracy.
- **Wins/Draws:** Number of datasets where GVD achieves equal or better performance than all baselines.
- **Losses:** Number of datasets where GVD underperforms relative to baselines.

Implementation Details. The teacher is selected from five random initializations based on Top-1 accuracy. Students are trained with a combined task and distillation loss:

$$\mathcal{L}_{\text{train}}(\boldsymbol{s}, \mathcal{D}) = \alpha_{\text{task}}\, \mathcal{L}_{\text{task}}(\boldsymbol{s}, \mathcal{D}) + \alpha_{\text{KD}}\, \mathcal{L}_{\text{KD}}(\boldsymbol{s}, \mathcal{D}). \tag{12}$$

Here, α_{task} and α_{KD} determine the contributions of the task loss $\mathcal{L}_{\text{task}}$ (cross-entropy) and the distillation loss $\mathcal{L}_{\text{KD}}$, respectively. We fix $\alpha_{\text{task}} = 1$ in all experiments. For image classification, $\alpha_{\text{KD}} = 1$, while for time-series classification it is selected via grid search over $\{0.1, 1, 10\}$. Models are implemented in PyTorch Paszke et al. (2019) and trained using the configurations in Tab. 8. All student results are averaged over five runs with different random initializations. The source code can be found in `https://github.com/anonymKD/AnonymKD2026.git`.

Table 8: Training configurations for different datasets. BS: batch size; LR: learning rate; WD: weight decay; LS: label smoothing.

Dataset	Epochs	BS	LR	Optimizer	Momentum	WD	LS	Scheduler	Augmentation
CIFAR-10	250	128	0.1	SGD	0.9	1×10^{-4}	0.0	Cosine	cutout
CIFAR-100	250	128	0.1	SGD	0.9	1×10^{-4}	0.0	Cosine	cutout
TinyImageNet	100	256	0.1	SGD	0.9	1×10^{-4}	0.1	Cosine	crop + flip + cutout
UCR	150	32	0.001	AdamW	–	1×10^{-4}	0.0	Cosine	none

Implementation of GVD. After training the teacher, the diffusion prior is learned using $\mathcal{L}_{\text{Diff}}$ (eq. (2)). We adopt a lightweight DDIM implementation (Huang et al., 2023) for diffusion prior training with $T = 1000$ diffusion steps. The diffusion prior is trained with the same training configurations as the teacher and student models on the datasets listed in Tab. 7. All GVD hyperparameters are listed in Tab. 9.

Table 9: Default hyperparameters for GVD.

Parameter	**Value**
Diffusion steps (T)	1000
Sampling steps (NFEs)	5
t_{start}–t_{end} (eq. (8))	50–30
λ_t^{va} (λ_{start}–λ_{end}) (eq. (7))	0.3–0.5
λ_t^{vt} (λ_{start}–λ_{end}) (eq. (7))	0.3–0.5
KD loss weight (α_{KD})	1 (images); best of $\{0.1, 1, 10\}$ (time series)
Task loss weight (α_{task})	1
Number of teacher views (M)	1

5.3 Additional Results

Performance on Image Classification Benchmarks The complete evaluation results on CIFAR-10 are summarized in Tab. 10. As shown, GVD achieves the best or tied-best performance across all metrics for both teacher–student pairs, with the highest Top-1, F1-score, and AUC-PRC in both settings, demonstrating consistent gains across architectures.

Evaluation Results on Time-Series Classification. To evaluate time-series classification across multiple datasets, we employ a win/draw/loss evaluation. A method is considered to *win* on a dataset if it achieves the highest Top-1 accuracy, while methods are ranked by their Top-1 accuracy on each dataset. The complete evaluation results on time-series classification are summarized in Tab. 11, under three teacher-student distillation settings. Across 20 UCR datasets, GVD achieves the highest average accuracy and lowest average rank for all three architectures, with 14/20, 14/20, and 11/20

Table 10: Evaluation of predictive performance on CIFAR-10. We report Top-1/Top-5 (%), F1-score, and AUC-PRC.

Student (Teacher)	Metric	Base	KD	FitNets	DIST	DiffKD	TeKAP	GVD
ResNet-18 (ResNet-34)	Top-1	93.45	94.11	94.08	93.52	93.32	94.06	**94.32**
	Top-5	99.84	**99.88**	**99.88**	99.85	99.81	**99.88**	**99.88**
	F1-score	93.45	94.10	94.07	93.52	93.32	94.07	**94.32**
	AUC-PRC	98.02	98.23	98.30	98.09	98.03	98.24	**98.43**
MobileNetV1 (ResNet-50)	Top-1	89.96	89.01	90.02	89.64	89.72	89.73	**90.38**
	Top-5	99.72	99.65	99.69	99.69	99.70	99.66	**99.73**
	F1-score	89.95	89.00	90.02	89.63	89.72	89.73	**90.27**
	AUC-PRC	95.74	95.02	95.28	95.64	95.62	95.66	**95.80**

Top-1 wins for LSTM, ResNet, and Inception, respectively. Notably, GVD records wins on at least 15/20 datasets against each individual baseline across all three settings, demonstrating consistent gains across diverse datasets and architectures.

Table 11: Evaluations Results on UCR datasets under three teacher–student distillation settings. Best results are shown in bold.

Dataset	LSTM-100-3 → LSTM-8-1					ResNet-64-3 → ResNet-8-3					Inception-32-6 → Inception-16-3				
	Base	KD	Fits	TeKAP	GVD	Base	KD	Fits	TeKAP	GVD	Base	KD	Fits	TeKAP	GVD
Computers	61.87	**64.53**	63.07	62.80	63.87	76.40	76.40	**78.53**	76.93	77.47	70.67	71.73	70.67	69.73	**71.90**
UWaveGestureLibraryAll	57.53	56.99	67.54	67.81	**68.09**	91.44	92.12	94.47	**94.58**	94.54	94.13	94.82	93.59	93.36	**94.87**
Strawberry	72.70	72.00	81.57	81.29	**82.22**	98.42	98.53	98.37	98.21	**98.59**	**98.04**	97.82	97.55	97.50	**98.04**
BeetleFly	70.00	**73.33**	70.00	70.00	**73.33**	88.33	93.33	**95.00**	93.33	93.33	83.33	88.33	83.33	83.33	**90.00**
Wafer	92.25	91.02	98.30	94.48	**98.31**	99.71	99.78	99.78	99.77	**99.85**	**99.82**	99.81	99.74	99.72	99.69
Lighting2	56.83	**57.38**	**57.38**	**57.38**	**57.38**	78.69	**81.42**	79.23	80.87	79.78	77.60	78.69	76.50	77.05	**78.79**
ItalyPowerDemand	76.77	75.90	78.85	78.75	**79.20**	96.60	96.86	96.92	96.86	**97.12**	96.99	96.99	97.08	**97.15**	**97.15**
Yoga	54.99	54.87	62.19	62.12	**63.09**	88.84	88.89	89.16	89.17	**89.28**	84.89	**86.71**	85.54	84.92	86.69
Trace	26.33	29.33	**41.67**	41.33	41.00	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	88.33	90.33	**99.33**	96.33	96.33
ShapesAll	17.72	17.67	20.61	21.39	**21.89**	77.94	77.94	78.78	78.39	**79.00**	62.11	60.67	61.17	**62.33**	61.22
MiddlePhalanxOutlineCorrect	53.61	48.50	53.94	53.89	**55.17**	81.94	**84.22**	82.78	83.44	84.00	75.83	77.67	77.89	76.83	**78.00**
SwedishLeaf	28.21	28.91	**39.25**	36.53	38.45	93.23	92.59	93.87	93.60	**94.67**	88.05	**89.33**	88.00	87.73	89.23
FaceAll	28.58	29.23	34.73	**38.48**	36.41	83.02	**86.82**	82.05	81.42	86.41	75.25	75.92	74.52	75.38	**76.06**
StarLightCurves	84.58	85.13	85.07	85.36	**85.73**	97.97	98.11	98.09	98.09	**98.12**	97.81	98.06	98.09	98.08	**98.11**
ECG200	68.33	70.00	68.00	68.33	**70.33**	90.67	91.67	91.33	91.67	**92.67**	90.33	89.67	**92.67**	91.67	92.33
MoteStrain	74.92	75.32	75.00	75.00	**75.43**	92.15	91.69	92.47	93.00	**93.16**	90.23	**91.08**	89.94	90.12	89.96
SonyAIBORobotSurfaceII	**64.85**	64.01	63.55	63.62	63.59	97.17	97.17	97.20	97.38	**97.41**	94.26	94.02	94.58	94.37	**94.82**
Ham	58.41	**62.54**	59.37	59.05	61.59	78.10	79.68	84.13	**84.44**	**84.44**	74.92	79.68	**82.86**	81.59	80.95
NonInvasiveFatalECG_Thorax1	16.62	17.27	27.21	28.02	**30.57**	92.57	92.69	93.01	92.82	**93.38**	85.09	86.99	86.90	87.60	**87.74**
NonInvasiveFatalECG_Thorax2	26.40	25.78	32.59	32.74	**32.99**	93.69	93.84	93.96	94.08	**94.20**	90.81	90.69	90.74	**91.13**	90.14
Avg. Top-1 Acc.	54.58	54.99	58.99	58.92	**59.31**	89.84	90.69	90.96	90.90	**91.37**	85.92	86.95	87.03	86.80	**87.60**
Avg. Rank	4.20	3.45	2.70	2.60	**1.45**	4.35	2.95	2.75	2.65	**1.35**	3.45	2.95	3.10	3.15	**1.95**
Top-1 Wins	1	4	3	2	**14**	1	4	3	3	**14**	2	3	3	3	**11**
Wins/Draws	19	17	18	17	**20**	20	17	18	18	**20**	16	15	15	15	**20**
Losses	1	2	2	3	**0**	0	3	1	2	**0**	4	5	5	5	**0**

Comparison with Multi-Teacher Distillation. We additionally evaluate two representative multi-teacher baselines on CIFAR-100 under the WRN-40-2→WRN-16-2 distillation setting. We compare GVD with Ensemble Distillation (ED) (Allen-Zhu & Li, 2020), using four teacher networks, and Teacher Assistant Knowledge Distillation (TAKD) (Mirzadeh et al., 2020), using the sequence of teacher assistants as WRN-40-2→WRN-34-2→WRN-28-2→WRN-22-2. As shown in Tab. 12, GVD achieves higher Top-1 accuracy (74.86) than ED (73.82) and TAKD (72.77), while requiring fewer auxiliary parameters (4.16M vs. 9.02M and 6.69M, respectively). Unlike multi-teacher methods, GVD generate multiple generative vicinal targets from a single teacher, eliminating the need for multiple models. Since the generative prior is used only during training, inference remains as efficient as the baseline classifier (without KD).

Table 12: Comparison with multi-teacher distillation methods on CIFAR-100 using WRN-40-2→WRN-16-2. Best results are shown in bold.

Metric	ED	TAKD	GVD
Top-1 Accuracy (%) ↑	73.82	72.77	**74.86**
Parameters (M) ↓	9.02	6.69	**4.16**

GVD Improves Teacher–Student Fidelity. Fidelity results for CIFAR-100, TinyImageNet, and CIFAR-10 are reported in Tab. 13, Tab. 3, and Tab. 14, respectively. On CIFAR-100 (Tab. 13), GVD achieves the highest Top-1 and Top-5 agreement in 5/6 teacher–student pairs and the lowest predictive KLD in 5/6 pairs. These consistent fidelity improvements across homogeneous and heterogeneous architectures indicate more faithful transfer of teacher behavior. Similar behavior is observed on CIFAR-10 (Tab. 14), where GVD achieves the highest Top-1 agreement for both teacher–student pairs and the best or tied-best Top-5 agreement. GVD also maintains low predictive KLD relative to other students, indicating improved alignment with the teacher.

Table 13: Evaluation of teacher–student fidelity on CIFAR-100. We report Top-1 agreement (↑), Top-5 agreement (↑), and average predictive KL divergence (↓) between teacher and student predictions.

	Homogeneous Architecture Distillation									**Heterogeneous Architecture Distillation**								
Teacher	WRN-40-2			ResNet-32x4			ResNet-56			ResNet-50			ResNet-32x4			ResNet-32x4		
Student	WRN-40-1			ResNet-8x4			ResNet-20			MobileNetV2			ShuffleNetV1			ShuffleNetV2		
	Top1A.	Top5A.	KLD	Top1A.	Top5A.	KLD	Top1A.	Top5A.	KLD	Top1A.	Top5A.	KLD	Top1A.	Top5A.	KLD	Top1A.	Top5A.	KLD
Base	73.94	92.80	1.02	73.89	93.13	1.02	71.33	92.44	0.95	62.92	86.72	1.40	72.48	92.01	1.19	74.37	92.63	0.97
KD	75.49	93.34	1.10	75.96	93.72	0.93	**72.88**	92.53	1.03	66.64	88.69	1.85	75.75	93.41	0.99	78.97	94.75	0.74
FitNets	75.57	93.83	0.90	75.25	94.04	0.80	71.90	93.23	0.86	64.50	87.81	**1.32**	73.43	92.39	1.10	74.75	92.95	0.96
DIST	74.41	93.30	0.97	74.49	93.26	0.95	72.32	92.51	0.90	62.52	86.63	1.39	73.50	92.60	1.06	75.17	93.46	0.89
DiffKD	74.29	93.02	1.02	74.01	93.24	1.01	71.74	92.42	0.93	63.44	86.73	1.39	72.79	92.02	1.16	74.44	93.19	0.94
TeKAP	75.66	93.68	0.90	75.31	94.19	0.80	72.52	**93.31**	0.85	63.58	87.20	1.37	72.65	92.12	1.12	74.16	92.85	0.97
GVD	**76.50**	**94.05**	**0.76**	**76.57**	**94.68**	**0.70**	72.69	93.20	**0.78**	**68.63**	**90.16**	1.35	**76.60**	**94.30**	**0.75**	**79.51**	**95.48**	**0.59**

Table 14: Teacher-Student fidelity comparison on CIFAR-10. We report Top-1/Top-5 agreement (%) ↑ and predictive KL divergence ↓.

Student (Teacher)	**Metric**	**Base**	**KD**	**FitNets**	**DIST**	**DiffKD**	**TeKAP**	**GVD**
ResNet-18 (ResNet-34)	Agreement (Top1)	94.16	94.64	94.70	94.12	94.05	94.57	**94.88**
	Agreement (Top5)	99.88	99.93	99.92	99.88	99.91	99.90	**99.95**
	KLD	0.18	0.21	**0.15**	0.17	0.18	**0.16**	**0.16**
MobileNetV1 (ResNet-50)	Agreement (Top1)	90.83	89.87	90.76	90.39	90.42	90.50	**90.96**
	Agreement (Top5)	99.77	**99.80**	99.78	99.74	99.78	99.75	**99.80**
	KLD	0.27	0.59	0.27	0.26	0.28	0.27	**0.25**

GVD Improves Student Calibration. Calibration results for CIFAR-100 students are reported in Tab. 15. As shown, GVD achieves the lowest ECE and Brier score across all six teacher–student pairs and the lowest NLL in 5/6 settings. These consistent improvements across homogeneous and heterogeneous architectures indicate better-calibrated student predictions, supporting the hypothesis that supervision from multiple nearby teacher targets can reduce overconfidence and improve calibration. We further visualize feature representations using PCA on 8 selected CIFAR-100 classes for different distilled students in Fig. 5. Points are colored by ground-truth class, with shading indicating prediction confidence. Compared to KD, GVD shows better feature organization and reduced overconfidence in ambiguous regions.

5.3.1 Robustness and Transfer Experiments.

GVD improves transferability. We evaluate the transferability of distilled students by transferring a WRN-40-1 student (distilled from a WRN-40-2 teacher on CIFAR-100) to STL-10. As shown in Tab. 16, GVD achieves the highest performance under both linear probing (53.54) and full fine-tuning (66.44). Notably, GVD improves over the strongest baseline by 0.75 points under linear probing and 0.59 points after full fine-tuning, indicating that vicinal supervision promotes more generalizable representations across datasets.

Robustness to Limited Distillation Data. We evaluate robustness to limited data by stratified subsampling CIFAR-100 with ratios $\{0.1, 0.2, 0.4, 1.0\}$ to distill WRN-40-2 into WRN-40-1. As shown in Tab. 17, GVD consistently outperforms all baselines across data ratios. The advantage is most pronounced with only 10% of the data, where GVD achieves 50.02, outperforming the strongest baseline (KD) by 1.05 points, FitNets by 5.24 points, and TeKAP by 8.00 points, indicating its ability to provide robust supervision under limited distillation data.

Table 15: Evaluation on CIFAR-100. We report Expected Calibration Error (ECE↓) and Brier Score (×100, ↓) to assess calibration, along with Negative Log-Likelihood (NLL↓).

	Homogeneous Architecture Distillation									**Heterogeneous Architecture Distillation**								
Teacher	WRN-40-2			ResNet-32x4			ResNet-56			ResNet-50			ResNet-32x4			ResNet-32x4		
Student	WRN-40-1			ResNet-8x4			ResNet-20			MobileNetV2			ShuffleNetV1			ShuffleNetV2		
	ECE	Brier	NLL	ECE	Brier	NLL	ECE	Brier	NLL	ECE	Brier	NLL	ECE	Brier	NLL	ECE	Brier	NLL
Base	0.13	0.41	1.20	0.14	0.41	1.26	0.11	0.43	1.18	0.16	0.54	1.68	0.16	0.45	1.46	0.13	0.41	1.26
KD	0.15	0.38	1.24	0.14	0.39	1.26	0.14	0.41	1.21	0.23	0.56	2.16	0.16	0.41	1.34	0.13	0.35	1.09
FitNets	0.12	0.38	1.09	0.11	0.39	1.08	0.09	0.41	1.09	0.15	0.52	1.59	0.15	0.43	1.38	0.13	0.41	1.27
DIST	0.12	0.40	1.16	0.12	0.40	1.19	0.10	0.41	1.13	0.15	0.55	1.68	0.14	0.42	1.32	0.12	0.39	1.17
DiffKD	0.13	0.40	1.20	0.13	0.41	1.26	0.10	0.42	1.16	0.16	0.54	1.68	0.16	0.44	1.44	0.13	0.40	1.22
TeKAP	0.12	0.38	1.08	0.11	0.38	1.07	0.09	0.41	1.08	0.15	0.53	**1.63**	0.16	0.44	1.42	0.13	0.41	1.27
GVD	**0.09**	**0.36**	**0.97**	**0.09**	**0.37**	**1.02**	**0.06**	**0.39**	**1.02**	**0.14**	**0.50**	1.64	**0.12**	**0.38**	**1.10**	**0.08**	**0.33**	**0.93**

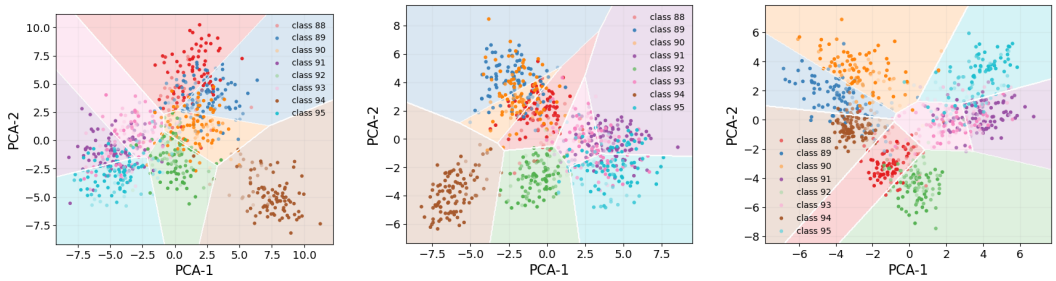


Figure 5: PCA features visualization for 8 selected CIFAR-100 classes: (a) Base, (b) KD, and (c) GVD. Points are colored by ground-truth class, with shading indicating prediction confidence. Background regions show coarse class partitions from a logistic regression model fitted in the 2D PCA space. Compared to KD, GVD shows better feature organization and reduced overconfidence in ambiguous regions.

Robustness to Imperfect Teacher Targets. We construct artificially degraded teacher targets using an overfitted teacher (WRN-40-2) trained on CIFAR-100. We distill students (WRN-40-1) from this degraded supervision and report the resulting performance drop in Tab. 18. GVD exhibits the smallest degradation (1.80 points), compared with 2.51 for KD, 2.25 for FitNets, and 2.04 for TeKAP. GVD also retains the highest accuracy (73.95) under degraded supervision, demonstrating robustness of vicinal supervison to imperfect teacher targets.

Table 16: Transfer learning on STL-10 (% Top-1).

(WRN-40-1)	KD	Fits	TeKAP	GVD
Linear probe	50.50	48.21	52.79	**53.54**
Full fine-tune	55.48	65.85	65.20	**66.44**

Table 17: Limited-data distillation.

Ratio	KD	Fits	TeKAP	GVD
0.1	48.97	44.78	42.02	**50.02**
0.2	60.85	57.48	56.82	**61.29**
0.4	68.94	66.65	65.79	**69.11**
1.0	74.40	74.31	74.52	**75.74**

Table 18: Overfitted supervision.

Supervision	KD	Fits	TeKAP	GVD
General	74.40	74.31	74.52	**75.74**
Overfitted	72.89	72.06	72.48	**73.95**
Drop (↓)	2.51	2.25	2.04	**1.80**

5.3.2 Computational Cost Analysis of GVD

We evaluate the computational overhead of GVD by measuring training cost on CIFAR-100 for distilling WRN-40-2 into WRN-40-1. Experiments are conducted with a batch size of 128 for 250 epochs (390 steps per epoch) on a single NVIDIA RTX 3090 GPU with synchronized timing. We discuss the computational cost of GVD in detail below.

Teacher Training. Training the teacher is identical to any standard KD pipeline and introduces no additional cost in GVD.

Diffusion-prior Training. Although our method introduces an additional diffusion prior alongside the teacher, the associated computational overhead is modest. The prior is trained once and reused across all distillation runs, making it a one-time cost. As shown in Table 19, its training cost is comparable to that of the student and even lower than that of the teacher (0.03 s vs 0.04 s per

step). Unlike conventional pixel-space diffusion models, the proposed prior operates in feature space, resulting in a lightweight and efficient design. Importantly, the diffusion prior enables diverse supervision similar to multi-teacher distillation, while remaining significantly more efficient, avoiding the linear increase in training cost required to train multiple teacher models.

Table 19: Training cost of model components on CIFAR-100 (WRN-40-2 → WRN-40-1). We report parameters, step time, and peak GPU memory.

Component	Params (M)	Step Time (s) ↓	Memory (GB) ↓
Teacher (WRN-40-2)	2.26	0.04	0.79
Student (WRN-40-1)	0.56	0.03	0.44
Diffusion Prior	4.87	0.03	0.61

Student Training. Student training costs are summarized in Table 20, which reports step time, inference time, and peak GPU memory. During the distillation phase, vicinal sampling uses only 5 NFEs and operates entirely in feature space, making each reverse-diffusion step computationally inexpensive. This is reflected in the moderate training overhead shown in Table 20: GVD incurs a step time of 0.12 s compared to 0.07 s for standard feature distillation methods, resulting in an additional $\sim$0.05 s per step, or $\sim$20 s per epoch (390 steps). This modest overhead is well justified by the consistent and significant performance improvements over standard KD and the Base classifier (Table 1). While moderately higher than lightweight KD methods, GVD remains more efficient than diffusion-based KD approaches such as DiffKD (0.15 s/step). In terms of memory, GVD requires 2.07 GB, which is higher than standard KD (0.44–0.47 GB) but lower than DiffKD (2.46 GB). Overall, GVD introduces a moderate training overhead while maintaining competitive efficiency among diffusion based KD.

Table 20: Computational cost comparison on CIFAR-100 (WRN-40-2 → WRN-40-1). We report step time (s), peak GPU memory (GB), and inference time (s).

Method	Step Time (s) ↓	Memory (GB) ↓	Inference Time (s) ↓
Base	0.03	0.44	0.004
KD	0.03	0.45	0.004
MSE	0.07	0.45	0.004
DIST	0.03	0.45	0.004
DiffKD	0.15	2.46	0.004
TeKAP	0.03	0.47	0.004
GVD	0.12	2.07	0.004

Inference Cost. GVD achieves approximately 0.03 ms per sample (0.004 s per batch), identical to the Base model. All additional computation occurs only during training, while the deployed model remains as efficient as the Base classifier.

Effect of Diffusion Parameters and Computational Cost. We ablate the number of inference steps (NFEs),i.e., score function evaluations, and evaluate their impact on GVD performance and computational cost (Table 21). Increasing NFEs introduces additional step time and memory consumption due to the extra reverse-diffusion iterations. However, strong performance is achieved with only a few steps. Since the diffusion prior is implemented using DDIM (Song et al., 2020a), 5 NFEs already achieve 75.74% Top-1, close to the best result of 75.82% obtained with 10 NFEs. We therefore adopt 5 NFEs as the default, offering a favorable trade-off between performance and efficiency. Inference time remains constant across all settings (0.004 s per batch), as diffusion is only applied during training.

Effect of the Number of Vicinal Samples and Computational Cost. We ablate the number of vicinal samples (M in eq. (10)) and analyze the cost–performance trade-off in Tab. 22. Increasing M provides marginal performance improvements, but the gains quickly saturate relative to the additional computational cost. Over E epochs, the student observes $E \times M$ vicinal views; in practice, even $M = 1$ provides sufficient diversity with a favorable accuracy–efficiency trade-off. Based on this

Table 21: Effect of inference steps (NFEs) on performance and computational cost. We report Top-1 accuracy (%), training step time (s), peak GPU memory (GB), and inference time per batch (s).

NFEs	Step Time (s) ↓	Memory (GB) ↓	Top-1	Inference (s) ↓
1	0.14	1.06	73.68	0.004
5	0.12	2.07	75.74	0.004
6	0.13	2.31	75.11	0.004
8	0.16	2.82	75.23	0.004
10	0.20	3.32	75.82	0.004
20	0.55	5.81	74.98	0.004

trade-off, we adopt $M = 1$ in all main experiments. Inference time remains constant across all settings (0.004 s per batch), as vicinal sampling is only applied during training.

Table 22: Effect of the number of diffusion samples on performance and computational cost. We report Top-1 accuracy (%), training step time (s), peak GPU memory (GB), and inference time per batch (s).

#Samples	Step Time (s) ↓	Memory (GB) ↓	Top-1	Inference (s) ↓
1	0.12	2.07	75.74	0.004
2	0.14	3.38	75.76	0.004
3	0.16	3.38	74.48	0.004
4	0.52	3.38	75.92	0.004
10	0.34	3.38	75.62	0.004

These ablations show that GVD provide a flexible cost–performance trade-off with moderate memory overhead and no impact on inference speed.

5.4 Discussion

Limitations and Future Works. Since the score-based generative prior can transforms an initial random noise sample into a teacher-like representation, the proposed formulation provides a source of teacher supervision that extends beyond the information obtainable from a single deterministic forward pass of the teacher. Therefore, a key strength of our approach is its ability to generate robust supervisory signals from a single teacher, reducing the reliance on multi-teacher setups while maintaining computational efficiency. However, since generated vicinal samples are derived from the learned teacher representation space, this may restrict the introduction of entirely novel semantic information. GVD is computationally much cheaper than multi-teacher distillation, although generating multiple vicinal views introduces a modest overhead compared to lightweight KD methods (Fits, KD), which is justified by the resulting performance gains. Future work will explore richer guidance mechanisms to further enhance diversity and semantic expressiveness.

In the search of extending the single teacher supervision to robust, progressive and diverse supervision, this work introduces a principled framework for generating controllable vicinal samples using diffusion-based generative priors with guidance functions. Our current implementation instantiate the controlling criterion as maximizing the ability to represent target-class while remaining within the local vicinity of the original representation. Accordingly, we modify the effective score field followed during reverse diffusion (using two guidance functions) to prioritize vicinal samples with higher target-class confidence. More importantly, this formulation allows this conditioning variable to encode alternative supervisory criterion; one can generate vicinal representations with different desirable properties, such as higher predictive uncertainty/entropy, improved calibration, or other task-specific qualities considered beneficial for knowledge transfer. Beyond KD, this paradigm opens a broader research direction in controllable data augmentation in representation space, where generated samples can be explicitly constrained to satisfy properties such as label consistency, that are difficult to achieve with heuristic augmentation methods.

Broader Impact. The proposed method can improve the efficiency and scalability of training compact models, facilitating deployment in resource-constrained settings such as edge devices and mobile platforms, while potentially reducing computational cost and environmental impact.

However, the use of generative priors introduces challenges, as mis-specified guidance may lead to unintended distortions or reduced interpretability, particularly in high-stakes applications such as finance, healthcare, or surveillance.

THE USE OF LARGE LANGUAGE MODELS (LLMS)

In preparing this manuscript, we used OpenAI's ChatGPT (GPT-4) as a writing assistance tool. Its role was limited to polishing language for improved clarity, grammar, and readability in certain sections of the paper and appendix. The model did not contribute to research ideation, methodological design, experimental execution, data analysis, or interpretation of results. All scientific content, technical contributions, and conclusions are solely the work of the authors. We take full responsibility for the accuracy and integrity of the paper.