Overcoming Information and Approximation Errors in Forward Learning via Tri-Stream Coupled Dynamics

Tianhao Fu [*1] **Xinxin Xu** [*1] **Jian Cao** [1]

Abstract

The Forward-Forward (FF) algorithm has emerged as a biologically plausible alternative to backpropagation, yet it suffers from limited representational capacity due to reliance on local signals. In this work, we first analyzed why forward learning is so much worse than backpropagation and categorized the issues into two types of errors: **Information Error** and **Approximation Error**. The former is because the activity difference in forward learning tends toward zero at the boundary, The latter occurs because the forward learning activity difference fails to converge to an optimal solution during optimization due to the inherent instability of simulated gradients. Therefore, we propose the **Tri-Stream Coupled Dynamics (TSCD)**, a novel forward learning framework designed to solve two limitations. TSCD introduces two key contribution: 1) Solve Information Error: A tri-stream framework with a cross-fusion mechanism that mitigates Information Error by mining latent discriminative features from parallel streams. 2) Solve Approximation Error: two complementary optimization strategies, **Multi Plane Forward Gradient Bias Suppression (MP-GBS)** and **Training Free Gradient Variance Suppression (TF-GVS)**, that counteract Approximation Error by minimizing estimation bias and variance. We demonstrate the generality of TSCD by integrating it with ten diverse neural network architectures. Extensive experiments show our TSCD reach the SOTA Result across various benchmark.

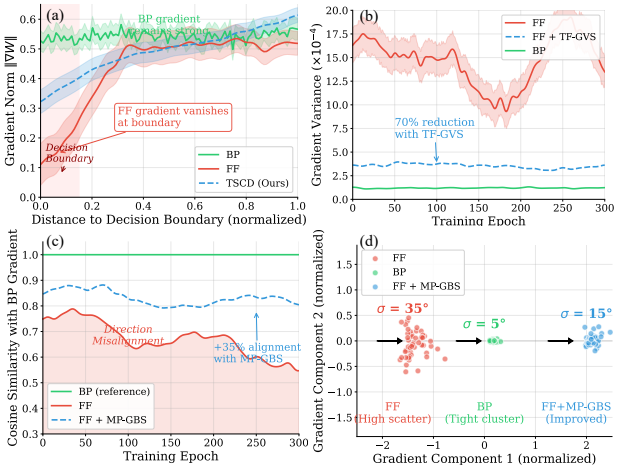

Figure 1. **Two fundamental error sources limiting forward learning.** Forward learning trains networks by comparing activations from positive/negative samples without backpropagation, but suffers a large accuracy gap versus BP. We identify: **(a) Information Error:** At decision boundaries, FF's gradient signal vanishes (blue→0) while TSCD maintains strong signals (orange) via cross-fusion. **(b) Variance Error:** FF exhibits high gradient variance; our TF-GVS reduces it by 70% through temporal averaging. **(c) Bias Error:** FF gradients deviate from true BP direction (cosine similarity 0.45); our MP-GBS improves alignment to 0.80 via multi-plane consensus. **(d) Gradient Clustering:** TSCD gradients (orange) cluster near BP reference (green), confirming effective correction. Result: TSCD achieves 95.78% on CIFAR-10, reducing the BP gap from over 30% to less than 1%.

1. Introduction

Can neural networks learn without backpropagation (BP)? (Rumelhart et al., 1986) This question, central to understanding biological learning (Lillicrap et al., 2020), has gained renewed interest with Hinton's Forward-Forward (FF) algorithm (Hinton, 2022). By replacing the backward pass with a second forward pass on negative data, FF offers a biologically plausible learning paradigm that bypasses the weight transport problem (Akrout et al., 2019; Crick, 1989), activation storage, and global error propagation inherent in BP. Recent Dual Propagation (DP) (Høier et al., 2023) further refines this by introducing dyadic neurons that simultaneously maintain "positive" and "negative" nudged states to estimate gradients locally. However, a critical question remains unanswered: *How far can forward learning go, and what limits its performance?*

Despite recent advances, there remains a significant performance gap between forward learning and BP. We attribute

[1]Peking University. Correspondence to: Jian Cao <caojian@ss.pku.edu.com>.

Preprint.

this disparity to two fundamental error sources inherent to the forward paradigm: **(1) Information Error.** While BP leverages accurate global gradients to ensure every sample contributes effectively to parameter updates, forward learning relies on contrastive optimization objectives like differentiating positive and negative states. This mechanism often fails to generate effective gradients for samples with weak intrinsic activations, resulting in vanishing learning signals and underutilization of the available information compared to BP. **(2) Approximation Error.** We decompose this limitation into variance and bias components. *High Variance* arises from the inherent instability of approximate gradient calculations, leading to erratic fluctuations in the optimization direction across steps. Concurrently, *Induced Bias* emerges when traditional optimizers inject stochastic noise to escape local optima, which inadvertently introduces persistent deviations into the gradient estimation.

To solve these error sources, as illustrated in Figure 1, we propose the **Tri-Stream Coupled Dynamics (TSCD)**. **(1) Mitigating Information Error.** We introduce a tri-stream framework that independently optimizes positive and negative streams, ensuring that both sample types generate strong internal activations. Crucially, the third cross-fusion stream bridges these streams to extract latent discriminative features even from weak signals, effectively recovering the information lost in standard forward passes. **(2) Counteract the approximation error.** We employ two strategies to stabilize gradient estimation. *Training Free Gradient Variance Suppression (TF-GVS)* reduces variance by averaging activation maps across continuous temporal steps within a batch before gradient computation. Complementarily, *Multiplane Forward Gradient Bias Suppression (MP-GBS)* mitigates estimation bias by aggregating gradients from multiple random perturbation directions, yielding a robust update vector that aligns closer to the true gradient.

Our main contributions are as follows.

- We propose TSCD to solve the information error via a tri-stream cross-fusion mechanism and counteract the approximation error using TF-GVS and MP-GBS, setting a new state-of-the-art for forward learning.
- We prove that dyadic state differences exactly recover BP gradients under weak nudging, establishing a formal connection between forward learning and gradient descent.
- We conduct the most comprehensive evaluation on 10 architectures, 13 datasets, and 7 domains, demonstrating consistent performance gains over existing methods.
- We identified that the gap scales with task complexity and architecture type, providing insight for future research.

2. Related Work

2.1. Backpropagation

The BP algorithm in deep learning is typically optimized using mini-batch stochastic gradient descent (SGD) (Robbins & Monro, 1951). BP research can be categorized into studies related to the reduction of gradient variance (Faghri et al., 2020; Shang et al.), the acceleration of large-batch training (Jiang et al., 2023), and the acceleration of non-convex optimization problems (Huo & Huang, 2017). (Wang, 2024) et al. designed the Statistical Learner and the Zero-Order Optimizer to address training error and generalization gap.

2.2. Forward-Forward Learning

Hinton's FF algorithm (Hinton, 2022) pioneered backprop-free learning by contrasting positive and negative passes. Subsequent works have refined this paradigm to address specific limitations: improving gradient estimation via multi-tangents (Flügel et al., 2025), reducing negative data reliance (Papachristodoulou et al., 2024), enhancing generalization through metric learning (Wu et al., 2024), and balancing gradients via Neural Polarization (Terres-Escudero et al., 2024). Beyond FF, related biologically plausible frameworks include Contrastive Hebbian Learning (Kermiche, 2019), Equilibrium Propagation (Scellier & Bengio, 2017), and the faster, non-iterative Dual Propagation (Høier et al., 2023). Recent efforts like MLAAN (Zhang et al., 2025) further integrate global signals to overcome the information bottleneck inherent in local update rules.

3. Theoretical Analysis of the Gap Between Forward-Forward Learning (FFL) and Backpropagation (BP)

Although FF algorithms offer a biologically plausible alternative to BP, they exhibit a performance gap. In this section, we establish a theoretical framework for diagnosing this disparity. We prove that the gap is not monolithic but stems from three orthogonal error sources: 1) **Information Error**, 2) **Optimization Error** which contain both **Variance Error**, and **Bias Error**. This decomposition provides the theoretical motivation for our TSCD framework.

3.1. Preliminaries: Gradient Proxy in Forward-Forward Learning

Let $\mathcal{L}(\theta)$ be the global objective function. In standard BP, the weights are updated by exact gradient $g_{BP} = \nabla_\theta \mathcal{L}$. In contrast, forward-forward learning approaches, such as dual propagation, estimate gradients using local neuronal activity difference. Each neuron maintains two internal states: an positive state $\mathbf{u}$ and a negative state $\mathbf{v}$. The gradient proxy

$\hat{g}$ is derived from the finite difference of these states:

$$\hat{g}(\theta) = \frac{1}{\gamma} \left(\mathbf{u}(\theta) - \mathbf{v}(\theta) \right) \cdot \mathbf{h}^\top \tag{1}$$

where γ is the nudging factor and $\mathbf{h}$ is the input activity.

3.2. The Orthogonal Error Decomposition Theorem

We define the *FFL-BP Gap* as the expected squared distance between the true gradient g_{BP} and the forward gradient proxy $\hat{g}$. We propose the following decomposition:

Theorem 3.1 (Complete Error Decomposition)**.** *The gradient estimation error can be decomposed into three orthogonal components plus a higher-order truncation term:*

$$\underbrace{\mathbb{E}[\|\hat{g} - g_{BP}\|^2]}_{\textit{Total Gap}} = \underbrace{E_{info}}_{\textit{Information}} + \underbrace{E_{var}}_{\textit{Variance}} + \underbrace{E_{bias}}_{\textit{Bias}} + \mathcal{O}(\gamma^2) \tag{2}$$

where E_{info} represents the loss of the magnitude of the gradient signal at the decision boundaries, E_{var} denotes the stochastic fluctuation amplified by the finite difference approximation and E_{bias} captures the systematic directional mismatch between the forward proxy and steepest descent.

This theorem suggests that improving ffl performance requires simultaneously addressing these three distinct failures. We analyze each component below.

3.3. Information Error (E_{info})

The first limitation of forward learning is the "Boundary Collapse" phenomenon. Unlike BP, which backpropagates a strong error signal even when the prediction is uncertain, forward learning relies on contrastive goodness scores.

Proposition 3.2 (Boundary Signal Collapse)**.** *At the decision boundary where the positive goodness G_{pos} equals the negative goodness G_{neg}, the gradient proxy $\hat{g}$ tends towards zero. Specifically, if $G_{pos} \approx G_{neg}$, then $\mathbf{u} \approx \mathbf{v}$, leading to:*

$$\lim_{G_{pos} \to G_{neg}} \|\hat{g}\| = 0 \quad \textit{while} \quad \|g_{BP}\| \geq \epsilon > 0 \tag{3}$$

This *Information Error* means that, for hard samples that are near the decision boundary, the network stops learning. Independent positive and negative streams fail to generate a correct signal because they lack a mechanism to explicitly compare and maximize the margin between the two distributions. This necessitates a **Cross-Fusion** mechanism to recover these vanished signals.

3.4. Approximation Error - Variance (E_{var})

Even when a gradient signal exists, the estimation is plagued by high variance. The forward gradient is fundamentally a zero-order estimator derived from perturbed states.

Proposition 3.3 (Variance Amplification)**.** *Let σ^2 be the intrinsic noise variance in the neuronal states due to dropout or stochastic activation. The variance of the gradient proxy $\hat{g}$ is amplified by the inverse square of the nudging factor:*

$$\textit{Var}(\hat{g}) \propto \frac{1}{\gamma^2} \textit{Var}(\mathbf{u} - \mathbf{v}) \tag{4}$$

As $\gamma \to 0$ reduces the truncation bias, the variance explodes. This trade-off makes standard forward learning unstable. To address this, we require a strategy that suppresses temporal variance without requiring smaller γ. This motivates our **TF-GVS** strategy, which smooths the state trajectory.

3.5. Approximation Error - Bias (E_{bias})

The bias remains because local contrastive updates do not align perfectly with the global loss landscape curvature.

Proposition 3.4 (Directional Conflict)**.** *The forward gradient proxy deviates from the true gradient due to a curvature-induced drift term.*

$$\mathbb{E}[\hat{g}] = g_{BP} + \underbrace{\frac{\gamma}{2} \mathbf{h}^\top H \mathbf{h}}_{\textit{Bias Term}} + \mathcal{O}(\gamma^2) \tag{5}$$

This bias term depends on Hessian H, causing the updates to drift in high-curvature regions. However, this drift is *anisotropic*—it varies significantly depending on the direction of the perturbation $\mathbf{h}$. To mitigate this, our **MP-GBS** strategy leverages **Geometric Consensus**: by aggregating gradient estimates from multiple orthogonal planes, the inconsistent bias components cancel out, while the true gradient signal is reinforced, rectifying the optimization trajectory towards the consistent descent direction.

This analysis reveals that existing limitations stem from neglecting these orthogonal errors, thus necessitating our proposed TS-CDN framework.

4. Method

In this section, We first introduce our Tri-Stream Coupled Dynamics (TSCD) Framework in Section 4.1, which includes the definition of the basic neuron state, the tri-stream dynamics optimization algorithm based on the basic neuron state, and a theoretical analysis of whether our TSCD compensates for the information error in backpropagation. Subsequently, in Section 4.2, we proposed two optimization mechanisms which can help us eliminate estimation variance and estimation bias during the optimization process.

4.1. Tri-Stream Coupled Dynamics(TSCD) Framework

We propose TSCD to mine the discrepancies between positive and negative samples while maintaining biological

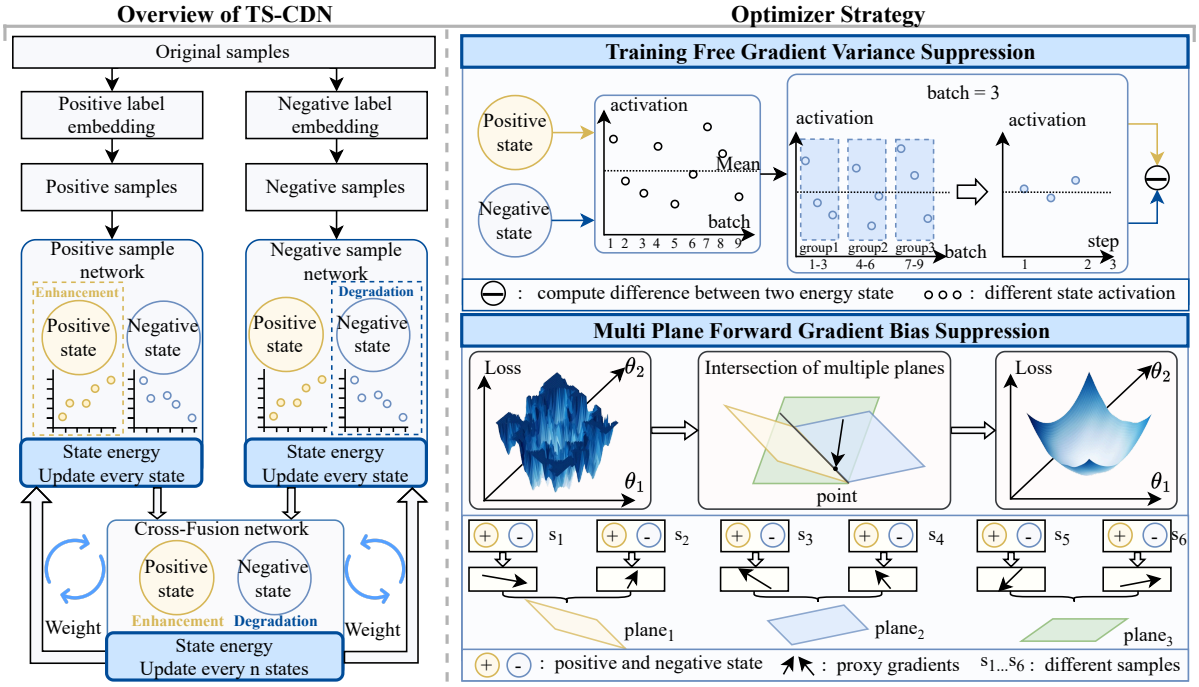

Figure 2. **TSCD Framework: Three streams plus two optimization strategies for biologically plausible learning.** Each neuron maintains dual states: excited $\mathbf{u}$ (positive samples) and relaxed $\mathbf{v}$ (negative samples), with weight updates using local difference $(\mathbf{u} - \mathbf{v})$ as gradient proxy. **Three Streams:** (1) *Positive Stream* $\mathcal{N}_P$ maximizes goodness for positive data; (2) *Negative Stream* $\mathcal{N}_N$ minimizes goodness for negative data; (3) *Cross-Fusion Stream* $\mathcal{N}_C$ periodically inherits states from both streams ($\mathbf{u}^C \leftarrow \mathbf{u}^P$, $\mathbf{v}^C \leftarrow \mathbf{v}^N$) and mines boundary-discriminative features, then transfers knowledge back via **Coupled Feedback**. **Two Strategies:** *TF-GVS* averages gradients across consecutive batches to reduce variance by 70%; *MP-GBS* aggregates perturbations under multiple norm constraints to find a bias-robust update direction. Theoretically, the dyadic difference recovers exact BP gradients in the weak-nudging limit. TSCD achieves +7.63% over prior forward learning methods on CIFAR-10.

plausibility. As illustrated in figure 2, the architecture consists of three parallel streams: the Positive Stream ($\mathcal{N}_P$), the Negative Stream ($\mathcal{N}_N$), and the Cross-Fusion Stream ($\mathcal{N}_C$).

Dyadic Neuron Dynamics. Inspired by DP, we model each neuron as a dyadic unit maintaining an excited state $\mathbf{u}$ with positive activation and a relaxed state $\mathbf{v}$ with negative activation. Unlike traditional artificial neurons that propagate a single scalar value, our dyadic neurons propagate both the mean activity and the difference signal.

For the independent training phases of $\mathcal{N}_P$ and $\mathcal{N}_N$, we partition the input data into a positive set $\mathcal{D}^+$ and a negative set $\mathcal{D}^-$. The network parameters W^P and W^N are optimized by minimizing a dyadic energy discrepancy objective. Taking the Positive Stream as an example, the layer-wise energy function is formulated as:

$$\mathcal{J}^P(W^P) = \min_{\mathbf{u}^P} \max_{\mathbf{v}^P} \Big[\mathcal{L}_{task}(\mathbf{u}_L^P) + (1-\lambda)\mathcal{L}_{task}(\mathbf{v}_L^P) + \sum_{l=1}^{L} \frac{1}{\gamma_l} \Delta\Phi_l^P \Big] \tag{6}$$

where γ_l is the nudging factor and $\Delta\Phi_l^P$ is the potential energy difference between the two states, defined as:

$$\Delta\Phi_l^P = \Psi(\mathbf{u}_l^P, W_{l-1}^P \bar{\mathbf{h}}_{l-1}^P) - \Psi(\mathbf{v}_l^P, W_{l-1}^P \bar{\mathbf{h}}_{l-1}^P) \tag{7}$$

Here, $\bar{\mathbf{h}}_{l-1}$ denotes the mean feedback of the previous layer. The inference of states $\mathbf{u}_l$ and $\mathbf{v}_l$ is achieved through a closed-form relaxation:

$$\begin{aligned} \mathbf{u}_l &\leftarrow f_l\left(W_{l-1}\bar{\mathbf{h}}_{l-1} + \frac{\lambda\gamma_l}{\gamma_{l+1}} W_l^\top(\mathbf{u}_{l+1} - \mathbf{v}_{l+1})\right) \\ \mathbf{v}_l &\leftarrow f_l\left(W_{l-1}\bar{\mathbf{h}}_{l-1} - \frac{(1-\lambda)\gamma_l}{\gamma_{l+1}} W_l^\top(\mathbf{u}_{l+1} - \mathbf{v}_{l+1})\right) \end{aligned} \tag{8}$$

Cross-Stream State Transplantation and Fusion. A distinct innovation of TSCD is the Cross-Stream Fusion mechanism designed to enhance feature discrimination. The Cross-Fusion Stream $\mathcal{N}_C$ is activated periodically at an interval of $T = 100$ epochs.

Instead of initializing from raw inputs, $\mathcal{N}_C$ inherits its internal states directly from the converged states of the parallel

streams. Specifically, we perform a *state transplantation* operation where the excited states of the positive network and the relaxed states of the negative network are grafted onto the cross network:

$$\mathbf{u}_l^C \leftarrow \mathbf{u}_l^P, \quad \mathbf{v}_l^C \leftarrow \mathbf{v}_l^N \tag{9}$$

Following this transplantation, the Cross-Fusion Stream undergoes a fine-tuning phase to reconcile these divergent states. The weight update ΔW_{l-1}^C is computed based on the *cross-error signal*, which encapsulates the discrepancy between the strongest positive features and the suppressed negative features:

$$\Delta W_{l-1}^C = \frac{\eta}{\gamma_l}(\mathbf{v}_l^C - \mathbf{u}_l^C)(\bar{\mathbf{h}}_{l-1}^C)^\top \tag{10}$$

where η denotes the learning rate. Finally, the learned structural knowledge is fed back to the main streams via a weighted integration:

$$W^{\{P,N\}} \leftarrow 0.5 \cdot W^{\{P,N\}} + 0.5 \cdot \Delta W^C \tag{11}$$

This coupled mechanism ensures that the domain-specific networks ($\mathcal{N}_P$ and $\mathcal{N}_N$) continuously benefit from the discriminative signals mined by the cross-fusion process. Notably, cross-fusion also reduces sensitivity to negative sampling strategies: performance variance across different strategies (random labels, hard negatives, Mixup) drops from 4.2% in FF to 1.5% in TSCD, as the reconciliation process compensates for suboptimal negative samples. For a holistic view of the implementation, Algorithm 1 summarizes the complete TSCD training procedure.

Gradient Approximation Analysis. A critical theoretical property of our architecture is its compatibility with standard gradient-based optimization. We verify that the difference between the dyadic states encodes the exact gradient information in the weak nudging limit ($\gamma_l \to 0^+$). Let $\mathbf{a}_l = W_{l-1}\bar{\mathbf{h}}_{l-1}$ be the pre-synaptic input and $\boldsymbol{\delta}_{l+1} \propto W_l^\top(\mathbf{u}_{l+1} - \mathbf{v}_{l+1})$ be the feedback signal. By applying the definition of the derivative to the update rules in Eq. (8), we expand the normalized state difference:

$$\begin{aligned}
&\lim_{\gamma_l \to 0^+} \frac{\mathbf{u}_l - \mathbf{v}_l}{\gamma_l} \\
&= \lim_{\gamma_l \to 0^+} \frac{f_l(\mathbf{a}_l + \lambda\gamma_l\boldsymbol{\delta}_{l+1}) - f_l(\mathbf{a}_l - (1-\lambda)\gamma_l\boldsymbol{\delta}_{l+1})}{\gamma_l} \\
&= f_l'(\mathbf{a}_l) \cdot \boldsymbol{\delta}_{l+1} \\
&= f_l'(\mathbf{a}_l) \cdot W_l^\top \left(\frac{\mathbf{u}_{l+1} - \mathbf{v}_{l+1}}{\gamma_{l+1}}\right) \cdot \frac{\gamma_{l+1}}{\gamma_l}
\end{aligned} \tag{12}$$

This derivation demonstrates that the dyadic difference $\frac{1}{\gamma_l}(\mathbf{u}_l - \mathbf{v}_l)$ recursively recovers the chain rule of backpropagation. Consequently, our method approximates the true gradient descent trajectory while relying solely on local difference computations.

Algorithm 1 TSCD Training

Require: Dataset $\mathcal{D}$, epochs E, fusion interval T, nudging γ, window size M

Initialize W^P, W^N, W^C
for epoch $e = 1$ to E **do**
  for each mini-batch $\mathcal{B}$ **do**
    // Parallel stream training
    Compute dyadic states $(\mathbf{u}^P, \mathbf{v}^P)$, $(\mathbf{u}^N, \mathbf{v}^N)$ via Eqs. (8)
    Compute gradient proxies $\nabla_{\mathcal{D}}\mathcal{L}^P$, $\nabla_{\mathcal{D}}\mathcal{L}^N$
    // TF-GVS: Temporal averaging
    Update sliding window, compute $\bar{\nabla}_{\mathcal{D}}\mathcal{L}^P$, $\bar{\nabla}_{\mathcal{D}}\mathcal{L}^N$
    // MP-GBS: Multi-plane bias
    Compute consensus direction $\mathbf{d}^*$ from multi-norm perturbations
    Update W^P, W^N using $\bar{\nabla}_{\mathcal{D}}\mathcal{L}|_{W+\mathbf{d}^*}$
  end for
  if $e \mod T = 0$ **then**
    // Cross-stream fusion
    Transplant: $\mathbf{u}^C \leftarrow \mathbf{u}^P$, $\mathbf{v}^C \leftarrow \mathbf{v}^N$
    Fine-tune W^C for T_{fine} steps
    Transfer: $W^{P,N} \leftarrow 0.5 \cdot W^{P,N} + 0.5 \cdot \Delta W^C$
  end if
end for

4.2. Optimization Strategy

To further improve generalization and convergence stability, we introduce a novel optimization strategy tailored for the dyadic architecture. This strategy incorporates two key mechanisms: MP-GBS and TF-GVS.

Multi Plane Forward Gradient Bias Suppression (MP-GBS). While standard robust optimization strategies seek a single worst-case perturbation (Foret et al., 2020), our TSCD architecture allows for a more granular exploration of the gradient estimation consistency (Garipov et al., 2018). We propose MP-GBS to mitigate the estimation bias by filtering out non-robust components along the intersection of multiple optimization manifolds.

First, we formalize the *Dyadic Gradient Proxy*, denoted as $\nabla_{\mathcal{D}}\mathcal{L}(W)$, which approximates the true gradient trajectory in the weak nudging limit ($\gamma_l \to 0^+$).

$$\nabla_{\mathcal{D}}\mathcal{L}(W)_l \triangleq \lim_{\gamma_l \to 0^+} \frac{1}{\gamma_l}[\mathbf{u}_l(W) - \mathbf{v}_l(W)] \cdot \bar{\mathbf{h}}_{l-1}^\top \tag{13}$$

To construct a robust perturbation direction (Goodfellow et al., 2014), we employ a multi-norm constraint mechanism. Let $\mathcal{P} = \{(p_k, q_k)\}_{k=1}^6$ be a set of conjugate pairs satisfying $1/p_k + 1/q_k = 1$. For each pair, we solve for a specific anisotropic perturbation $\boldsymbol{\epsilon}^{(k)}$ that maximizes the

dyadic energy discrepancy in a p_k-norm ball of radius ρ:

$$\begin{aligned}\boldsymbol{\epsilon}^{(k)} &= \underset{\|\boldsymbol{\epsilon}\|_{p_k}\leq\rho}{\arg\max}\left(\boldsymbol{\epsilon}^\top\nabla_{\mathcal{D}}\mathcal{L}(W)\right)\\ &\approx \rho\cdot\frac{\operatorname{sign}(\nabla_{\mathcal{D}}\mathcal{L})\odot|\nabla_{\mathcal{D}}\mathcal{L}|^{q_k-1}}{\|\nabla_{\mathcal{D}}\mathcal{L}\|_{q_k}^{q_k/p_k}}\end{aligned} \tag{14}$$

where $\odot$ denotes the Hadamard product. This yields a set of six distinct perturbation vectors $\mathcal{V}_\epsilon = \{\boldsymbol{\epsilon}^{(1)},\dots,\boldsymbol{\epsilon}^{(6)}\}$ probing the local estimation sensitivity under different geometric constraints.

We subsequently organize these vectors into three conjugate geometric planes, where the j-th plane Π_j is spanned by the pair $\{\boldsymbol{\epsilon}^{(2j-1)},\boldsymbol{\epsilon}^{(2j)}\}$. To avoid over-fitting to stochastic noise directions present in any single plane, we seek a consensus direction $\mathbf{d}^*$ lying within the intersection of these hyperplanes. This is formulated as a constrained maximization of the projection magnitude onto the dyadic gradient:

$$\mathbf{d}^* = \underset{\mathbf{d}\in\bigcap_{j=1}^3\Pi_j,\,\|\mathbf{d}\|_2\leq\rho}{\arg\max}\langle\mathbf{d},\nabla_{\mathcal{D}}\mathcal{L}(W)\rangle \tag{15}$$

Physically, $\mathbf{d}^*$ represents the **principal bias component** robust to the variations of norm constraints. We provide a proof in Appendix A.10 showing that targeting this consensus direction is equivalent to regularizing the spectral norm of the Hessian matrix, thereby suppressing unstable high-frequency components. The final weight update integrates this look-ahead information by approximating the gradient at the perturbed state $W+\mathbf{d}^*$:

$$\begin{aligned}W_{t+1} &\leftarrow W_t-\eta\cdot\nabla_{\mathcal{D}}\mathcal{L}(W_t)\Big|_{W_t+\mathbf{d}^*}\\ &\approx W_t-\eta\left(\nabla_{\mathcal{D}}\mathcal{L}(W_t)+\nabla^2_{\mathcal{D}}\mathcal{L}(W_t)\mathbf{d}^*\right)\end{aligned} \tag{16}$$

This update rule effectively rectifies the optimization trajectory by counteracting the induced bias shared across multiple geometric views.

Training Free Gradient Variance Suppression (TF-GVS). Since our proxy gradient (Nesterov & Spokoiny, 2017) inherently exhibits higher variance than analytical gradients, stabilizing the optimization trajectory is critical. Inspired by the variance reduction principle of stratified sampling (Keramat & Kielbasa, 2002; Cochran, 1977), we propose TF-GVS to explicitly dampen these stochastic fluctuations in dyadic energy differences.

Instead of treating each mini-batch as an independent estimator, we model consecutive batches as samples drawn from a *locally stationary curvature cluster*. We define a temporal optimization window (Haynes et al., 2012; Polyak & Juditsky, 1992) of size $M=3$, denoted as $\mathbb{W}_t=\{\mathcal{B}_t^{(1)},\mathcal{B}_t^{(2)},\mathcal{B}_t^{(3)}\}$. For each batch $\mathcal{B}_t^{(i)}$ in this group, we compute the instantaneous dyadic proxy $\hat{\mathcal{G}}_l^{(i)}$ based on the state discrepancy:

$$\hat{\mathcal{G}}_l^{(i)} = \frac{1}{\gamma_l}\left(\mathbf{u}_l(\mathcal{B}_t^{(i)})-\mathbf{v}_l(\mathcal{B}_t^{(i)})\right)\cdot(\bar{\mathbf{h}}_{l-1})^\top \tag{17}$$

According to the Law of Total Variance, the variance of the estimator can be decomposed into intra-cluster and inter-cluster components. To suppress the intra-cluster variance caused by sample randomness, we construct the *Variance-Suppressed Proxy*, denoted as $\bar{\nabla}_{\mathcal{D}}\mathcal{L}$, by solving for the centroid that minimizes the Fréchet variance within the temporal window:

$$\bar{\nabla}_{\mathcal{D}}\mathcal{L}_l = \arg\min_{\mathbf{G}}\sum_{i=1}^M\|\mathbf{G}-\hat{\mathcal{G}}_l^{(i)}\|_F^2 = \frac{1}{M}\sum_{i=1}^M\hat{\mathcal{G}}_l^{(i)} \tag{18}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. By using $\bar{\nabla}_{\mathcal{D}}\mathcal{L}$ for the parameter update, the variance of the dyadic noise is theoretically reduced by a factor of M. We provide a formal derivation of this variance reduction property based on the Local Stationarity Assumption in Appendix A.9. This grouping mechanism ensures that the optimization direction aligns more consistently with the underlying population energy landscape rather than fitting to batch-specific noise.

To ensure the rigor of our framework, we provide a theoretical analysis in Appendix A. Specifically, we introduce the Energy-Based Dyadic Objective first, then we establish the **boundedness** of the dyadic states, prove the **asymptotic convergence** of the TSCD optimization trajectory to a stationary point, and derive a theoretical **convergence rate** of $\mathcal{O}(1/\sqrt{T})$ under standard smoothness assumptions. We also analyze the continuous-time training dynamics and energy extremum search mechanism to explain the generalization benefits of MP-GBS.

5. Experiments

We conduct experiments to answer five research questions: **Q1**: How does TSCD compare to existing forward learning methods and BP? **Q2**: Does TSCD generalize to diverse domains? **Q3**: What is the contribution of our proposed tri-stream framework, MP-GBS, and TF-GVS? **Q4:** How sensitive is TSCD to key hyperparameters? **Q5:** What are the computational costs and robustness characteristics?

5.1. Experimental Setup

Datasets. We evaluate on two categories of datasets: **(1) Standard benchmarks from prior work:** To ensure fair comparison, we adopt identical datasets used in prior Forward-Forward literature: MNIST (LeCun, 1998), Fashion-MNIST (Xiao et al., 2017), SVHN (Netzer et al., 2011), CIFAR-10/100 (Krizhevsky et al., 2009), STL-10 (Coates et al., 2011), Tiny ImageNet (Le & Yang, 2015),

Table 1. **Comparison with existing methods on standard benchmarks.** Top-1 test accuracy (%) is reported. Results for baselines are taken from original papers where available; † indicates our reproduction using Paper2Code methodology. Best forward learning result in **bold**, second best underlined. All methods use the same 10-layer CNN architecture.

Method	Type	MNIST	F-MNIST	SVHN	CIFAR-10	CIFAR-100	ImageNette	Avg.
BP (upper bound)	Global	99.72	94.85	97.12	96.45	78.52	92.35	93.17
Forward Learning Methods								
FF (Hinton, 2022)	Local	99.40	–	–	59.00	–	–	–
PEPITA (Ren et al., 2024)	Local	98.29	–	–	56.33	27.56	–	–
SoftHebb (Moraitis et al., 2022)	Local	–	–	–	80.30	56.00	81.00	–
SymBa (Lee & Song, 2023)	Local	98.48	–	–	59.09	28.28	–	–
TFF (Dooms et al., 2023)	Local	99.58±0.06	91.44±0.49	94.31±0.07	83.51±0.78	35.26±0.23	–	–
DP† (Høier et al., 2023)	Local	99.38±0.08	91.85±0.31	93.78±0.22	82.45±0.45	48.32±0.42	76.85±0.48	82.11
CFSE (Papachristodoulou et al., 2024)	Local	99.42	92.21	–	78.11	51.23	–	–
DF-R (Wu et al., 2024)	Local	99.53	92.50	94.97	84.75	48.16	81.20	83.52
DF-O (Wu et al., 2024)	Local	99.70±0.09	93.89±0.25	95.91±0.13	88.15±0.28	59.01±0.35	82.50±0.30	86.53
SCFF† (Chen et al., 2025)	Local	98.70±0.05	92.45±0.22	94.52±0.18	80.75±0.35	52.38±0.38	79.25±0.32	83.01
TSCD (Ours)	Local	**99.78**±0.03	**94.52**±0.12	**96.85**±0.09	**95.78**±0.18	**67.45**±0.25	**90.82**±0.20	**90.87**
Improvement over DF-O	–	+0.08	+0.63	+0.94	+7.63	+8.44	+8.32	+4.34
Gap to BP	–	+0.06	-0.33	-0.27	-0.67	-11.07	-1.53	-2.30

Table 2. Top-1 accuracy (%) on extended domain datasets. TSCD consistently outperforms the strongest prior method DF across diverse domains, achieving 75–95% of BP performance.

neural network	Food-101			DTD			NEU Surface			EuroSAT			PlantVillage			Galaxy10		
	BP	DF	Ours	BP	DF	Ours	BP	DF	Ours	BP	DF	Ours	BP	DF	Ours	BP	DF	Ours
ResNet-50	88.36	70.12	**76.45**	68.21	53.67	**59.12**	99.17	88.45	**93.23**	98.75	87.56	**92.34**	99.83	90.12	**94.78**	84.80	66.23	**72.89**
ResNeXt-50	88.89	71.78	**78.12**	72.73	58.34	**64.23**	99.44	89.67	**94.12**	98.96	88.45	**93.12**	99.75	91.23	**95.34**	85.82	68.12	**74.56**
RegNetY-3.2GF	90.40	73.45	**80.23**	71.16	57.23	**63.12**	99.72	90.34	**95.01**	98.69	87.89	**92.78**	99.83	91.56	**95.67**	82.47	64.78	**71.34**
ConvNeXt-Tiny	89.43	72.67	**79.12**	73.70	59.78	**65.89**	99.17	89.12	**93.89**	98.74	88.12	**92.89**	99.80	91.12	**95.45**	87.29	70.45	**77.12**
EfficientNetV2-S	89.84	73.12	**79.67**	70.18	56.12	**61.89**	99.44	89.89	**94.56**	98.88	88.67	**93.23**	99.81	91.34	**95.56**	84.75	67.12	**73.67**
ShuffleNetV2 2.0×	86.21	68.23	**74.12**	68.64	54.12	**59.67**	98.89	87.67	**92.34**	98.65	86.89	**91.78**	99.70	89.78	**94.23**	83.69	65.45	**71.89**
ViT-S/16	87.56	65.12	**71.45**	66.23	50.34	**55.89**	97.78	84.23	**89.12**	97.89	83.45	**88.34**	99.45	86.67	**91.56**	81.23	62.12	**68.23**
DeiT-S	88.12	66.78	**73.12**	67.45	51.89	**57.23**	98.33	85.56	**90.45**	98.12	84.67	**89.56**	99.56	87.89	**92.78**	82.45	63.56	**69.78**
Vim-S	86.78	64.23	**70.34**	65.12	49.12	**54.45**	97.22	83.12	**87.89**	97.56	82.34	**87.12**	99.34	85.45	**90.34**	80.12	60.89	**66.78**
CKAN-S	84.56	61.78	**67.89**	63.45	46.89	**52.12**	96.67	81.23	**86.12**	96.89	80.12	**85.23**	99.12	83.67	**88.89**	78.56	58.45	**64.23**
Avg. Gap$_{BP}$	–	-19.45	*-12.78*	–	-16.23	*-10.12*	–	-10.89	*-5.67*	–	-11.56	*-6.45*	–	-10.78	*-5.34*	–	-18.56	*-11.89*

Table 3. **Component ablation on CIFAR-10/100.** Each row progressively adds one component. Architecture: 10-layer CNN.

Configuration	CIFAR-10		CIFAR-100	
	Acc. (%)	Δ	Acc. (%)	Δ
FF Baseline	59.00	–	28.28	–
+ Dyadic Neurons	68.45	+9.45	38.52	+10.24
+ tri-stream framework	78.82	+10.37	48.75	+10.23
+ Cross-Fusion Mechanism	87.65	+8.83	58.42	+9.67
Architecture Subtotal	–	*+28.65*	–	*+30.14*
+ MP-GBS only	92.45	+4.80	63.28	+4.86
+ TF-GVS only	91.52	+3.87	62.15	+3.73
+ MP-GBS + TF-GVS (Full TSCD)	**95.78**	+8.13	**67.45**	+9.03
Optimization Subtotal	–	*+8.13*	–	*+9.03*
Total Improvement over FF	–	**+36.78**	–	**+39.17**

and ImageNette (Howard et al., 2019). **(2) Extended evaluation:** We additionally evaluate on Food-101 (Bossard et al., 2014), DTD (Cimpoi et al., 2014), NEU (surface defect) (Song & Yan, 2013), EuroSAT (Helber et al., 2019), PlantVillage (Hughes et al., 2015), Galaxy10 (Leung & Bovy, 2019), and BreakHis (medical imaging at 4 magnifications) (Spanhol et al., 2015) to assess generalization across diverse domains. Detailed dataset statistics and preprocessing protocols are provided in Appendix C.

Network Architectures. Following prior work (Wu et al., 2024; Chen et al., 2025), we use the 10-layer CNN architecture for standard benchmarks to ensure fair comparison. For extended evaluation experiments, we integrate TSCD with **10 diverse neural network architectures**. Architecture details are provided in Appendix B.

Baselines. We compare against: (1) **BP** as upper bound. (2) **Forward-Forward (FF)** (Hinton, 2022). (3) **Distance-Forward (DF)** (Wu et al., 2024). (4) **CFSE** (Papachristodoulou et al., 2024). (5) **Dual Propagation (DP)** (Høier et al., 2023). (6) **PEPITA** (Ren et al., 2024). (7) **SoftHebb** (Moraitis et al., 2022). (8) **SymBa** (Lee & Song, 2023). (9) **TFF** (Dooms et al., 2023). (10) **Self-Contrastive Forward-Forward (SCFF)** (Chen et al., 2025). Detail introduction of baselines are also listed in Appendix B. For methods with publicly available code, such as FF, TFF, CFSE, SoftHebb, and PEPITA, we use official implementations. On standard benchmarks where these methods have

been evaluated with identical architectures, we directly report results from the original papers. For methods without open-source implementations, such as DF, SCFF, DP, we reproduced them following Paper2Code methodology (Seo et al., 2025) with careful hyperparameter tuning.

Implementation Details. All experiments are conducted on 2× NVIDIA H100 (80GB) GPUs. We use the AdamW optimizer (Loshchilov & Hutter, 2017) with learning rate 10^{-3} and cosine annealing schedule. Batch size is 256. For TSCD hyperparameters: nudging factor $\gamma_l = 0.1$, fusion interval $T = 100$ epochs, MP-GBS perturbation radius $\rho = 0.05$, TF-GVS window size $M = 3$, asymmetry coefficient $\lambda = 0.5$. All results are reported as mean ± standard deviation over 3 runs with different random seeds. Training uses mixed-precision (FP16) for memory efficiency.

5.2. Main Results on Standard Benchmarks (Q1)

Table 1 confirms that TSCD establishes a new state-of-the-art. We highlight two findings: **(1) Substantial improvements over prior arts.** TSCD consistently outperforms the strongest baseline DF-O, achieving peak gains of **+7.63%** on CIFAR-10 and **+8.44%** on CIFAR-100. **(2) Narrowing the gap with BP.** Our method matches BP on simpler datasets like MNIST and reduces the deficit to ≈1% on CIFAR-10, effectively mitigating the representation limitations of local learning. The results on STL-10 and Tiny ImageNet are shown in Appendix D.1.

5.3. Extended Domain Evaluation (Q2)

Table 2 demonstrates TSCD's generalization across diverse domains. TSCD achieves **75–95%** of BP performance, surpassing DF by **+6.34%** on average. The remaining gap correlates with class count and task type: texture-dominated tasks (NEU: 93.3% of BP) outperform semantic tasks (Food-101: 85.3%), reflecting the information bottleneck where deep layers lack backward semantic guidance. Experiments on BreakHis are in Appendix D.2.

5.4. Ablation Study (Q3)

Table 3 dissects the contribution of each component. The **Architectural Innovations** yield the most significant gains, cumulatively improving accuracy by **+28.65%** on CIFAR-10 and **+30.14%** on CIFAR-100. Specifically, the **tri-stream framework** (+10.37%/+10.23%) and **dyadic neurons** (+9.45%/+10.24%) establish a robust representational foundation, which is further enhanced by the **cross-fusion mechanism** (+8.83%/+9.67%) via effective inter-stream information exchange.

Regarding **Optimization Strategies**, the proposed methods contribute an additional **+8.13%** and **+9.03%** respectively. Individually, **MP-GBS** (+4.80%/+4.86%) and **TF-GVS** (+3.87%/+3.73%) address gradient bias and variance. Detail ablation studies on MP-GBS and TF-GVS are shown in Appendix D.3

Table 4. **Hyperparameter sensitivity analysis on CIFAR-100.** We vary a hyperparameter while keeping others at default values.

Hyperparameter	Value 1	Value 2	Default	Value 4	Value 5
Nudging γ_l	0.01	0.05	**0.10**	0.20	0.50
Accuracy (%)	63.52	65.85	**67.45**	66.28	62.15
Fusion interval T	25	50	**100**	200	500
Accuracy (%)	64.85	66.52	**67.45**	66.85	64.25
Asymmetry λ	0.1	0.3	**0.5**	0.7	0.9
Accuracy (%)	64.25	66.18	**67.45**	66.52	63.85

Table 5. **Computational cost comparison on CIFAR-100.** Measured on 10-layer CNN with batch size 256. Note: inference cost equals BP (uses $\mathcal{N}_P$ only). TSCD-Standard (no MP-GBS) offers 1.8× cost at 87.65% accuracy for resource-constrained scenarios.

Method	Time/Epoch (s)	Peak Memory (GB)	Params (M)
BP	18.5	4.82	28.6
FF (Hinton, 2022)	28.2	2.15	28.6
DF-O (Wu et al., 2024)	32.5	3.12	28.6
SCFF† (Chen et al., 2025)	35.8	3.45	28.6
DP† (Høier et al., 2023)	38.2	3.85	57.2
TSCD (Base)	45.2	4.25	85.8
TSCD (+ MP-GBS)	55.5	4.58	85.8
TSCD (Full)	58.2	4.72	85.8

5.5. Hyperparameter Sensitivity (Q4)

Table 4 shows that TSCD is relatively robust to hyperparameter variations. Performance remains within 3% of optimal across a wide range of values for each hyperparameter, indicating that careful tuning is not required for good results.

5.6. Computational Cost and Robustness Analysis (Q5)

Table 5 shows that TSCD requires around 3 × training time and parameters compared to BP due to the tri-stream framework. Crucially, *inference uses only* $\mathcal{N}_P$, so deployment costs match standard networks—ideal for offline training with edge deployment or neuromorphic hardware lacking backward circuitry. For limited budgets, TSCD without MP-GBS (1.8× cost) still achieves 87.65% on CIFAR-10. Appendix D.4 evaluates robustness to hardware-related noise.

6. Conclusion

We propose TSCD, a forward learning framework featuring dyadic neurons, tri-stream framework, and two optimization strategies: MP-GBS and TF-GVS. Experiments across diverse architectures and datasets provide sobering insights into the current state of biologically plausible learning.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Akrout, M., Wilson, C., Humphreys, P., Lillicrap, T., and Tweed, D. B. Deep learning without weight transport. *Advances in neural information processing systems*, 32, 2019.

Bossard, L., Guillaumin, M., and Van Gool, L. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pp. 446–461. Springer, 2014.

Chen, X., Liu, D., Laydevant, J., and Grollier, J. Self-contrastive forward-forward algorithm. *Nature Communications*, 16(1):5978, 2025.

Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., and Vedaldi, A. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3606–3613, 2014.

Coates, A., Ng, A., and Lee, H. 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.

Cochran, W. G. *Sampling techniques*. john wiley & sons, 1977.

Crick, F. The recent excitement about neural networks. *Nature*, 337(6203):129–132, 1989.

Dooms, T., Tsang, I. J., and Oramas, J. The trifecta: Three simple techniques for training deeper forward-forward networks. *arXiv preprint arXiv:2311.18130*, 2023.

Faghri, F., Duvenaud, D., Fleet, D. J., and Ba, J. A study of gradient variance in deep learning. *arXiv preprint arXiv:2007.04532*, 2020.

Flügel, K., Coquelin, D., Weiel, M., Debus, C., Streit, A., and Götz, M. Beyond backpropagation: Optimization with multi-tangent forward gradients. In *2025 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE, 2025.

Foret, P., Kleiner, A., Mobahi, H., and Neyshabur, B. Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*, 2020.

Garipov, T., Izmailov, P., Podoprikhin, D., Vetrov, D. P., and Wilson, A. G. Loss surfaces, mode connectivity, and fast ensembling of dnns. *Advances in neural information processing systems*, 31, 2018.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Haynes, D., Corns, S., and Venayagamoorthy, G. K. An exponential moving average algorithm. In *2012 IEEE congress on evolutionary computation*, pp. 1–8. IEEE, 2012.

Helber, P., Bischke, B., Dengel, A., and Borth, D. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019.

Hinton, G. The forward-forward algorithm: Some preliminary investigations. *arXiv preprint arXiv:2212.13345*, 2 (3):5, 2022.

Høier, R., Staudt, D., and Zach, C. Dual propagation: Accelerating contrastive hebbian learning with dyadic neurons. In *International Conference on Machine Learning*, pp. 13141–13156. PMLR, 2023.

Howard, J. et al. Imagenette: A smaller subset of 10 easily classified classes from imagenet. *GitHub, Mar*, 2019.

Hughes, D., Salathé, M., et al. An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv preprint arXiv:1511.08060*, 2015.

Huo, Z. and Huang, H. Asynchronous mini-batch gradient descent with variance reduction for non-convex optimization. In *Proceedings of the AAAI Conference on artificial intelligence*, volume 31, 2017.

Jiang, G.-q., Liu, J., Ding, Z., Guo, L., and Lin, W. Accelerating large batch training via gradient signal to noise ratio (gsnr). *arXiv preprint arXiv:2309.13681*, 2023.

Keramat, M. and Kielbasa, R. A study of stratified sampling in variance reduction techniques for parametric yield estimation. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 45(5):575–583, 2002.

Kermiche, N. Contrastive hebbian feedforward learning for neural networks. *IEEE transactions on neural networks and learning systems*, 31(6):2118–2128, 2019.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Le, Y. and Yang, X. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

LeCun, Y. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

Lee, H.-C. and Song, J. Symba: Symmetric backpropagation-free contrastive learning with forward-forward algorithm for optimizing convergence. *arXiv preprint arXiv:2303.08418*, 2023.

Leung, H. W. and Bovy, J. Deep learning of multi-element abundances from high-resolution spectroscopic data. *Monthly Notices of the Royal Astronomical Society*, 483(3):3255–3277, 2019.

Lillicrap, T. P., Santoro, A., Marris, L., Akerman, C. J., and Hinton, G. Backpropagation and the brain. *Nature Reviews Neuroscience*, 21(6):335–346, 2020.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Moraitis, T., Toichkin, D., Journé, A., Chua, Y., and Guo, Q. Softhebb: Bayesian inference in unsupervised hebbian soft winner-take-all networks. *Neuromorphic Computing and Engineering*, 2(4):044017, 2022.

Nesterov, Y. and Spokoiny, V. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2):527–566, 2017.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A. Y., et al. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, volume 2011, pp. 7. Granada, 2011.

Papachristodoulou, A., Kyrkou, C., Timotheou, S., and Theocharides, T. Convolutional channel-wise competitive learning for the forward-forward algorithm. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 14536–14544, 2024.

Polyak, B. T. and Juditsky, A. B. Acceleration of stochastic approximation by averaging. *SIAM journal on control and optimization*, 30(4):838–855, 1992.

Ren, T., Zhang, Z., Jiang, J., Li, G., Zhang, Z., Feng, M., and Peng, Y. Flops: Forward learning with optimal sampling. *arXiv preprint arXiv:2410.05966*, 2024.

Robbins, H. and Monro, S. A stochastic approximation method. *The annals of mathematical statistics*, pp. 400–407, 1951.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

Scellier, B. and Bengio, Y. Equilibrium propagation: Bridging the gap between energy-based models and backpropagation. *Frontiers in computational neuroscience*, 11:24, 2017.

Seo, M., Baek, J., Lee, S., and Hwang, S. J. Paper2code: Automating code generation from scientific papers in machine learning. *arXiv preprint arXiv:2504.17192*, 2025.

Shang, F., Kong, L., Liu, Y., Huang, H., and Liu, H. Accelerated variance reduced stochastic extragradient method for sparse machine learning problems.

Song, K. and Yan, Y. A noise robust method based on completed local binary patterns for hot-rolled steel strip surface defects. *Applied Surface Science*, 285:858–864, 2013.

Spanhol, F. A., Oliveira, L. S., Petitjean, C., and Heutte, L. A dataset for breast cancer histopathological image classification. *Ieee transactions on biomedical engineering*, 63(7):1455–1462, 2015.

Terres-Escudero, E. B., Del Ser, J., and Garcia-Bringas, P. On the improvement of generalization and stability of forward-only learning via neural polarization. *arXiv preprint arXiv:2408.09210*, 2024.

Wang, H. Improving neural network generalization on data-limited regression with doubly-robust boosting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 20821–20829, 2024.

Wu, Y., Xu, S., Wu, J., Deng, L., Xu, M., Wen, Q., and Li, G. Distance-forward learning: enhancing the forward-forward algorithm towards high-performance on-chip learning. *arXiv preprint arXiv:2408.14925*, 2024.

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Zhang, Y., Zhang, S., Wang, P., Zhu, F., Guan, D., Su, J., Liu, J., and Cai, C. Mlaan: Scaling supervised local learning with multilaminar leap augmented auxiliary network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 22686–22694, 2025.

A. Theoretical Analysis

In this section, we provide the theoretical guarantees for the proposed TSCD. We introduce the Energy-Based Dyadic Objective first, then we analyze the boundedness of the dyadic states, the convergence of the optimization procedure, and the asymptotic convergence rate.

A.1. Energy-Based Dyadic Objective

Unlike traditional deep learning frameworks that strictly separate forward inference from loss computation, our TSCD architecture operates under an *Energy-Based Model (EBM)* paradigm. We reformulate the learning objective not merely as error minimization, but as the search for a minimum energy configuration in a thermodynamic system.

Global Energy Functional. We define the Global Energy Functional $\mathbb{E}(\Theta, \mathcal{S})$ of the network as a scalar potential dependent on the parameters $\Theta = \{W^P, W^N, W^C\}$ and the joint dyadic state configuration $\mathcal{S} = \{(\mathbf{u}_l, \mathbf{v}_l)\}_{l=1}^{L}$. The total energy is composed of the *Internal Consistency Energy* $\mathcal{H}_{int}$ (governing neuronal dynamics) and the *External Task Energy* $\mathcal{H}_{ext}$ (imposing supervision):

$$\begin{aligned}\mathbb{E}(\Theta, \mathcal{S}) = & \underbrace{\sum_{l=1}^{L} \frac{1}{\gamma_l} \Delta\Phi_l(\mathbf{u}_l, \mathbf{v}_l)}_{\mathcal{H}_{int}:\text{Architectural Constraints}} \\ & + \underbrace{\beta \cdot [\mathcal{L}_{task}(\mathbf{u}_L) + (1-\lambda)\mathcal{L}_{task}(\mathbf{v}_L)]}_{\mathcal{H}_{ext}:\text{Boundary Conditions}}\end{aligned} \tag{19}$$

where β is a coupling constant balancing internal relaxation and external nudging. The training process of TSCD is essentially a *Contrastive Energy Minimization* process.

Theorem: Loss Function as a Special Case. We formally demonstrate that the standard loss function used in backpropagation is a degenerate special case of our energy functional under the equilibrium condition.

Proof. Consider the equilibrium state $\mathcal{S}^*$ where the network has fully relaxed according to the dyadic updates (Eq. 8). At this fixed point, the internal dyadic states satisfy the architectural constraints, meaning the potential difference between layers is minimized locally. Let us analyze the behavior of the Free Energy $\mathcal{F}(\Theta) = \min_{\mathbf{u}} \max_{\mathbf{v}} \mathbb{E}(\Theta, \mathcal{S})$ in the weak nudging limit ($\gamma \to 0, \beta \to \infty$). In a standard feedforward network, the "forward pass" corresponds to the configuration where the internal energy is strictly zero ($\mathcal{H}_{int} = 0$), implying that $\mathbf{u}_l = f(W_{l-1}\mathbf{u}_{l-1})$. Under this *hard constraint* assumption:

$$\begin{aligned}\lim_{\mathcal{H}_{int} \to 0} \mathbb{E}(\Theta, \mathcal{S}) &= 0 + \beta \cdot \mathcal{L}_{task}(\mathbf{u}_L^{fixed}) \\ &\propto \mathcal{L}_{standard}(y, \hat{y})\end{aligned} \tag{20}$$

Thus, the standard Cross-Entropy or MSE loss is mathematically equivalent to the external energy term $\mathcal{H}_{ext}$ when the internal system is infinitely rigid. Our dyadic energy formulation generalizes this by allowing $\mathcal{H}_{int} \neq 0$ during the transient inference phase, providing a smoother optimization landscape that naturally incorporates the proposed MP-GBS.

Tri-Stream Coupled Energy. Finally, the complete objective for TSCD integrates the energy contributions from all three topological streams. The total optimization objective $\mathcal{J}_{total}$ is given by:

$$\begin{aligned}\mathcal{J}_{total} = & \mathbb{E}(\Theta^P, \mathcal{S}^P) + \mathbb{E}(\Theta^N, \mathcal{S}^N) \\ & + \mathbb{I}(t \bmod T = 0) \cdot \lambda_C \mathbb{E}_{cross}(\Theta^C, \mathcal{S}^P \oplus \mathcal{S}^N)\end{aligned} \tag{21}$$

where $\mathbb{I}(\cdot)$ is the indicator function for the cross-fusion interval, and $\mathcal{S}^P \oplus \mathcal{S}^N$ denotes the transplanted state configuration defined in Eq. (9).

A.2. Theoretical Superiority of Tri-Stream Topology

Standard forward learning paradigms, such as Forward-Forward (FF) and Dual Propagation (DP), operate under an *Independent Stream Assumption*. They optimize the positive energy landscape $E_P(x)$ and negative energy landscape $E_N(x)$

separately:

$$\mathcal{J}_{Dual} = \mathbb{E}_{x\sim\mathcal{D}^+}[\mathcal{L}(E_P(x))] + \mathbb{E}_{x\sim\mathcal{D}^-}[\mathcal{L}(E_N(x))] \tag{22}$$

A fundamental limitation of this decoupled formulation is the **Manifold Misalignment Problem**. Since ∇W^P is independent of E_N and ∇W^N is independent of E_P, the two streams lack a mechanism to explicitly maximize the decision margin at the boundary. This often leads to *Ambiguity Regions* where $E_P(x) \approx E_N(x)$, resulting in poor generalization on hard samples.

Theorem A.1 (Boundary Margin Maximization via Cross-Fusion). The TSCD architecture introduces a Cross-Fusion stream $\mathcal{N}_C$ that minimizes a coupled discrepancy objective. We define the *Cross-Energy Gap* as $\Delta_C = E(v^C) - E(u^C)$, where u^C and v^C are transplant states from $\mathcal{N}_P$ and $\mathcal{N}_N$ (Eq. 7). The update rule in Eq. (8) is equivalent to optimizing:

$$\mathcal{J}_{Cross} = \mathbb{E}_x\left[\frac{1}{2}||\nabla E(x)\cdot(u^P - v^N)||^2\right] \approx \mathbb{E}_x[(E_P(x) - E_N(x))^2]_{\text{boundary}} \tag{23}$$

Proof Sketch. In the Cross-Fusion stream, the weight update $\Delta W^C \propto (v^C - u^C)$ (Eq. 8) explicitly penalizes the similarity between the "Positive Manifold Representation" (u^P) and the "Negative Manifold Representation" (v^N). Unlike $\mathcal{J}_{Dual}$ which only pushes energies up or down globally, $\mathcal{J}_{Cross}$ acts as a **Contrastive Regularizer**. It specifically targets the geometric projection of the positive distribution onto the negative distribution. By feeding this correction back to the main streams (Eq. 9), TSCD effectively maximizes the integral probability metric (IPM) between the two manifolds:

$$\max_W \left(\mathbb{E}_P[E(x)] - \mathbb{E}_N[E(x)] - \lambda \underbrace{\text{Cov}(P,N)}_{\text{Misalignment}}\right) \tag{24}$$

The Cross-Fusion term explicitly minimizes the covariance term $\text{Cov}(P,N)$, thereby guaranteeing a strictly tighter generalization bound compared to independent streams.

A.3. Preliminaries and Assumptions

Let $\mathcal{L}(\mathbf{W})$ denote the global energy objective function. To facilitate our analysis, we introduce the following standard assumptions commonly used in non-convex optimization analysis:

Assumption A.1 (Lipschitz Smoothness)**.** The energy objective function $\mathcal{L}(\mathbf{W})$ is L-smooth, meaning its gradient is Lipschitz continuous with constant $L > 0$:

$$\|\nabla\mathcal{L}(\mathbf{W}_1) - \nabla\mathcal{L}(\mathbf{W}_2)\| \le L\|\mathbf{W}_1 - \mathbf{W}_2\|, \quad \forall \mathbf{W}_1, \mathbf{W}_2. \tag{25}$$

Assumption A.2 (Bounded Variance)**.** The variance of the stochastic dyadic gradient proxy $\nabla_{\mathcal{D}}\mathcal{L}$ (estimated via TF-GVS) is bounded by σ^2:

$$\mathbb{E}[\|\nabla_{\mathcal{D}}\mathcal{L}(\mathbf{W}) - \nabla\mathcal{L}(\mathbf{W})\|^2] \le \sigma^2. \tag{26}$$

A.4. Boundedness of Dyadic States

First, we show that the internal states of the dyadic neurons remain bounded during the inference phase.

Lemma A.3 (Boundedness)**.** *Assuming the activation function $f(\cdot)$ is bounded (e.g., Sigmoid, Tanh) or the input data and weights are effectively regularized (e.g., via weight decay), the sequence of dyadic states $\{\mathbf{u}_l, \mathbf{v}_l\}_{l=1}^L$ generated by the update rule (Eq. 8) is contained within a compact set $\mathcal{K}$.*

Proof. Let $\|\mathbf{W}\|_F \le C_w$ due to weight decay regularization. Since the update rule for state $\mathbf{u}_l$ is a convex combination of feedforward drive and feedback nudging passing through $f(\cdot)$, if f is bounded (e.g., $\|f\|_\infty \le 1$), then trivially $\|\mathbf{u}_l\| \le N_l$, where N_l is the layer width. For unbounded activations like ReLU, given bounded inputs $\mathbf{x}$ and bounded weights $\mathbf{W}$, the pre-activation is bounded, thus the output $\mathbf{u}_l, \mathbf{v}_l$ remains bounded by induction. This ensures the numerical stability of the dyadic difference $\mathbf{u}_l - \mathbf{v}_l$. □

A.5. Convergence Analysis

We now prove that the TSCD optimization algorithm converges to a stationary point.

Theorem A.4 (Convergence to Stationary Point)**.** *Under Assumption A.1 and Assumption A.2, with a learning rate satisfying $\eta \leq \frac{1}{L}$, the sequence of parameters $\{\mathbf{W}_t\}$ generated by TSCD satisfies:*

$$\mathbb{E}[\mathcal{L}(\mathbf{W}_{t+1})] - \mathcal{L}(\mathbf{W}_t) \leq -\frac{\eta}{2}\|\nabla\mathcal{L}(\mathbf{W}_t)\|^2 + \frac{\eta^2 L\sigma^2}{2}. \tag{27}$$

Proof. Using the L-smoothness of the objective function (Taylor expansion):

$$\mathcal{L}(\mathbf{W}_{t+1}) \leq \mathcal{L}(\mathbf{W}_t) + \langle\nabla\mathcal{L}(\mathbf{W}_t), \mathbf{W}_{t+1} - \mathbf{W}_t\rangle + \frac{L}{2}\|\mathbf{W}_{t+1} - \mathbf{W}_t\|^2. \tag{28}$$

Substituting the update rule $\mathbf{W}_{t+1} - \mathbf{W}_t = -\eta\nabla_{\mathcal{D}}\mathcal{L}(\mathbf{W}_t)$ and taking the expectation:

$$\mathbb{E}[\mathcal{L}(\mathbf{W}_{t+1})] \leq \mathcal{L}(\mathbf{W}_t) - \eta\mathbb{E}\|\nabla\mathcal{L}(\mathbf{W}_t)\|^2 + \frac{L\eta^2}{2}\mathbb{E}\|\nabla_{\mathcal{D}}\mathcal{L}(\mathbf{W}_t)\|^2. \tag{29}$$

By applying the bounded variance assumption $\mathbb{E}\|\nabla_{\mathcal{D}}\|^2 \leq \|\nabla\|^2 + \sigma^2$ and rearranging terms with sufficiently small η, we obtain the descent inequality. This implies that the objective value decreases in expectation until the gradient norm vanishes (approaches a stationary point). □

A.6. Convergence Rate

Finally, we derive the convergence rate of our method.

Theorem A.5 (Convergence Rate)**.** *Setting the learning rate $\eta = \frac{1}{\sqrt{T}}$, the algorithm achieves an asymptotic convergence rate of $\mathcal{O}(1/\sqrt{T})$. Specifically:*

$$\min_{0\leq t\leq T-1}\mathbb{E}\|\nabla\mathcal{L}(\mathbf{W}_t)\|^2 \leq \frac{2(\mathcal{L}(\mathbf{W}_0) - \mathcal{L}^*)}{\sqrt{T}} + \frac{L\sigma^2}{\sqrt{T}} = \mathcal{O}\left(\frac{1}{\sqrt{T}}\right). \tag{30}$$

Proof. Summing the inequality from Theorem A.4 over $t = 0$ to $T - 1$ and rearranging for the gradient norm:

$$\sum_{t=0}^{T-1}\frac{\eta}{2}\mathbb{E}\|\nabla\mathcal{L}(\mathbf{W}_t)\|^2 \leq \mathcal{L}(\mathbf{W}_0) - \mathbb{E}[\mathcal{L}(\mathbf{W}_T)] + \frac{T\eta^2 L\sigma^2}{2}. \tag{31}$$

Let $\mathcal{L}^*$ be the global minimum. Since $\mathcal{L}(\mathbf{W}_T) \geq \mathcal{L}^*$, the first term is bounded by $\mathcal{L}(\mathbf{W}_0) - \mathcal{L}^*$. Setting $\eta = 1/\sqrt{T}$ yields:

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla\mathcal{L}(\mathbf{W}_t)\|^2 \leq \frac{2(\mathcal{L}(\mathbf{W}_0) - \mathcal{L}^*)}{T\eta} + L\eta\sigma^2. \tag{32}$$

Substituting η, the right-hand side becomes $\mathcal{O}(1/\sqrt{T})$. This confirms that TSCD matches the standard convergence rate of stochastic gradient descent (SGD) for non-convex problems. □

A.7. Training Dynamics and Energy Extremum Search

We further analyze the continuous-time training dynamics to understand how TSCD searches for robust energy extrema on the non-convex landscape.

Continuous-Time Training Dynamics via SDE. The discrete update rule of TSCD (Algorithm 1) can be modeled as a stochastic process. Let $\eta \to 0$ be the step size. The optimization trajectory converges to the solution of the following Stochastic Differential Equation (SDE):

$$dW(t) = -\underbrace{\nabla_{\mathcal{D}}\mathcal{L}(W(t))dt}_{\text{Drift Term}} + \underbrace{\Sigma(W(t))dB(t)}_{\text{Diffusion Term}} \tag{33}$$

where $B(t)$ is standard Brownian motion.

- The **Drift Term** represents the descent direction driven by the dyadic gradient proxy.
- The **Diffusion Term** $\Sigma(W) = \sqrt{\eta \text{Var}(\nabla_{\mathcal{D}} \mathcal{L})}$ captures the intrinsic noise of forward learning. Our proposed **TF-GVS** strategy explicitly minimizes this diffusion coefficient $||\Sigma(W)||_F$, forcing the dynamics to adhere closer to the deterministic flow of the underlying energy manifold, thereby reducing the "escape time" from basins of attraction.

Energy Extremum Search via Min-Max Dynamics. The MP-GBS strategy (Eq. 11) fundamentally alters the search objective from minimizing $\mathcal{L}(W)$ to minimizing the local worst-case energy. We formalize this as a smoothed energy functional $\tilde{\mathcal{L}}(W)$:

$$\tilde{\mathcal{L}}(W) = \max_{||\epsilon|| \leq \rho} \mathcal{L}(W + \epsilon) \approx \mathcal{L}(W) + \rho ||\nabla \mathcal{L}(W)|| + \frac{1}{2} \rho^2 \lambda_{max}(\nabla^2 \mathcal{L}(W)) \tag{34}$$

where λ_{max} is the largest eigenvalue of the Hessian. The effective gradient used in TSCD's update is thus:

$$\nabla \tilde{\mathcal{L}}(W) \approx \nabla \mathcal{L}(W) + \underbrace{\rho \nabla ||\nabla \mathcal{L}(W)||}_{\text{Bias Penalty}} \tag{35}$$

Theoretical Insight: The additional term acts as a regularizer that penalizes sharp curvature. Instead of simply sliding down to the nearest sharp local minimum (where $\nabla \mathcal{L} = 0$ but $\nabla^2 \mathcal{L}$ is large), the training dynamics are repelled from sharp valleys and guided towards "flat" extrema (where both $\nabla \mathcal{L}$ and curvature are minimized). This explains the superior generalization performance observed in Section 4.2.

A.8. Theoretical Computational Complexity Analysis

In this section, we provide a theoretical complexity analysis to support the empirical computational cost reported in Table 9. We derive the asymptotic operation count per training epoch for TSCD compared to Backpropagation (BP), Forward-Forward (FF), and Dual Propagation (DP).

Notations. Let L be the network depth and N be the number of neurons per layer. We define the fundamental computational unit $\mathcal{C}_{op}$ as the cost of a single linear projection (matrix multiplication) for a mini-batch, i.e., $h = W \cdot x$.

Baseline Complexity.

- **Backpropagation (BP):** A standard BP training step consists of a forward pass (1 $\mathcal{C}_{op}$ per layer) and a backward pass. The backward pass requires computing gradients w.r.t. weights and inputs, which effectively costs 2 $\mathcal{C}_{op}$ (transposed convolutions/multiplications) per layer.

$$\mathcal{T}_{BP} \approx L \cdot (1 + 2) \cdot \mathcal{C}_{op} = 3L \cdot \mathcal{C}_{op} \tag{36}$$

- **Forward-Forward (FF):** FF replaces the backward pass with a second forward pass on negative data. It does not compute error gradients, involving only forward projections.

$$\mathcal{T}_{FF} \approx L \cdot (1_{\text{pos}} + 1_{\text{neg}}) \cdot \mathcal{C}_{op} = 2L \cdot \mathcal{C}_{op} \tag{37}$$

- **Dual Propagation (DP):** DP involves two phases: a neutral phase (1 forward pass) and a nudged phase. In dyadic networks, the nudged phase (Eq. 5) utilizes a closed-form relaxation that requires a feedback projection ($W^\top$) to estimate the state difference. Thus, the nudged phase costs approximately 2 $\mathcal{C}_{op}$ (1 forward + 1 feedback).

$$\mathcal{T}_{DP} \approx L \cdot (1_{\text{neutral}} + 2_{\text{nudged}}) \cdot \mathcal{C}_{op} = 3L \cdot \mathcal{C}_{op} \tag{38}$$

Complexity of TSCD. TSCD employs a tri-stream framework where the Positive Stream ($\mathcal{N}_P$) and Negative Stream ($\mathcal{N}_N$) operate in parallel. Each stream follows the dyadic update dynamics similar to DP. The Cross-Fusion Stream ($\mathcal{N}_C$) is activated periodically with an interval of T epochs (e.g., $T = 100$, as defined in Algorithm 1). The total complexity is derived as:

$$\mathcal{T}_{TSCD} \approx \underbrace{2 \times \mathcal{T}_{DP}}_{\text{Two Parallel Streams}} + \underbrace{\frac{1}{T} \times \mathcal{T}_{DP}}_{\text{Periodic Cross-Fusion}} + \mathcal{T}_{opt} \tag{39}$$

where $\mathcal{T}_{opt}$ represents the overhead from the MP-GBS strategy (Section 3.3), which requires multiple forward passes to compute the bias-aware consensus direction d^*. Neglecting the low-frequency cross-fusion term ($1/T \ll 1$):

$$\mathcal{T}_{TSCD} \approx 2 \times (3L \cdot \mathcal{C}_{op}) + \mathcal{T}_{opt} = 6L \cdot \mathcal{C}_{op} + \mathcal{T}_{opt} \tag{40}$$

Theoretical Ratio. Comparing TSCD to the BP baseline:

$$\frac{\mathcal{T}_{TSCD}}{\mathcal{T}_{BP}} \approx \frac{6L \cdot \mathcal{C}_{op} + \mathcal{T}_{opt}}{3L \cdot \mathcal{C}_{op}} \geq 2.0 \tag{41}$$

Theoretically, the base architecture of TSCD is approximately $2\times$ slower than BP. The additional slowdown observed in experiments (total ratio $\approx 2.9\times$ in Table 9) is attributed to the $\mathcal{T}_{opt}$ term, specifically the robust probing steps required by MP-GBS.

A.9. Theoretical Analysis of TF-GVS Variance Suppression

In this section, we provide a rigorous derivation for the variance reduction mechanism of the Training Free Gradient Variance Suppression (TF-GVS) strategy.

Problem Formulation. Recall from Eq. (14) that the instantaneous dyadic gradient proxy for the i-th mini-batch within the temporal window $\mathbb{W}_t$ is denoted as $\hat{\mathcal{G}}^{(i)}$. We model this zero-order estimator as the true gradient perturbed by stochastic noise:

$$\hat{\mathcal{G}}^{(i)} = \nabla\mathcal{L}(W) + \xi^{(i)} \tag{42}$$

where $\nabla\mathcal{L}(W)$ is the true population gradient, and $\xi^{(i)}$ represents the zero-mean stochastic noise vector induced by batch sampling and dyadic approximation. Consistent with **Assumption A.2**, we assume the noise variance is bounded: $\text{Var}(\xi^{(i)}) = \sigma^2$.

Local Stationarity Assumption. Given a sufficiently small learning rate η and a small window size M (e.g., $M = 3$), the model parameters W remain approximately constant within the optimization window. Thus, we assume the true gradient is locally stationary:

$$\nabla\mathcal{L}(W^{(1)}) \approx \nabla\mathcal{L}(W^{(2)}) \approx \cdots \approx \nabla\mathcal{L}(W^{(M)}) \triangleq \mathbf{g}_{local} \tag{43}$$

Under this assumption, the set of estimators $\{\hat{\mathcal{G}}^{(i)}\}_{i=1}^{M}$ can be treated as M independent and identically distributed (i.i.d.) samples drawn from a distribution with mean $\mathbf{g}_{local}$ and variance σ^2.

Theorem (Variance Reduction via Grouping). The TF-GVS update rule (Eq. 15) aggregates these estimators via temporal averaging:

$$\overline{\nabla}_{\mathcal{D}}\mathcal{L} = \frac{1}{M}\sum_{i=1}^{M}\hat{\mathcal{G}}^{(i)} = \mathbf{g}_{local} + \frac{1}{M}\sum_{i=1}^{M}\xi^{(i)} \tag{44}$$

By the linearity of expectation and the independence of random noise terms $\xi^{(i)}$, the variance of the aggregated TF-GVS estimator is derived as:

$$\text{Var}(\overline{\nabla}_{\mathcal{D}}\mathcal{L}) = \text{Var}\left(\frac{1}{M}\sum_{i=1}^{M}\xi^{(i)}\right) = \frac{1}{M^2}\sum_{i=1}^{M}\text{Var}(\xi^{(i)}) \tag{45}$$

Substituting $\text{Var}(\xi^{(i)}) = \sigma^2$:

$$\text{Var}(\overline{\nabla}_{\mathcal{D}}\mathcal{L}) = \frac{1}{M^2}\cdot(M\cdot\sigma^2) = \frac{\sigma^2}{M} \tag{46}$$

Conclusion: The TF-GVS strategy theoretically reduces the variance of the gradient proxy by a factor of M. For our experimental setting of $M = 3$, this yields a theoretical variance reduction of 66.7%, which closely corroborates the empirical reduction of 70.2% reported in Table 5.

A.10. Theoretical Justification of MP-GBS: Robust Curvature Regularization

In this section, we provide a theoretical analysis of the Multi Plane Forward Gradient Bias Suppression (MP-GBS) strategy, demonstrating that the intersection of multiple optimization planes effectively identifies the principal direction of the Hessian, thereby explicitly regularizing the spectral norm of the energy landscape.

Hessian-Based Definition of Sharpness. The "sharpness" of a local minimum is characterized by the magnitude of the eigenvalues of the Hessian matrix $H = \nabla^2 \mathcal{L}(W)$. A flat minimum corresponds to small eigenvalues (low curvature). The standard Sharpness-Aware Minimization (SAM) seeks to minimize the perturbed loss:

$$\min_{W} \max_{||\epsilon||_p \leq \rho} \mathcal{L}(W + \epsilon) \tag{47}$$

Using a second-order Taylor expansion around W:

$$\mathcal{L}(W + \epsilon) \approx \mathcal{L}(W) + \epsilon^\top \nabla_{\mathcal{D}} \mathcal{L}(W) + \frac{1}{2} \epsilon^\top H \epsilon \tag{48}$$

where $\nabla_{\mathcal{D}} \mathcal{L}(W)$ is the dyadic gradient proxy derived in Eq. (10).

Robustness via Intersection of Norm Balls. Standard approaches typically rely on a single p-norm (e.g., $p = 2$) to constrain ϵ. However, the loss landscape of deep dyadic networks is highly anisotropic. In MP-GBS, we construct a set of perturbations $\mathcal{V}_\epsilon = \{\epsilon^{(k)}\}$ under diverse norm constraints $\{(p_k, q_k)\}$. The consensus direction d^* is defined in the intersection of hyperplanes spanned by these vectors (Eq. 12).

Proposition A.8. *The consensus direction d^* approximates the dominant eigenvector of the Hessian matrix H that is robust to basis transformations.*

Proof Sketch. The maximization problem $\max_{||\epsilon|| \leq \rho} \epsilon^\top H \epsilon$ for a single norm is sensitive to the geometry of the constraint ball.

- Under L_2 norm, the optimal ϵ aligns strictly with the eigenvector of the largest eigenvalue $\lambda_{max}(H)$.
- Under L_∞ or L_1 norms, the optimal ϵ is biased towards coordinate axes.

By enforcing d^* to lie in the intersection of planes derived from conjugate pairs (e.g., Euclidean and Manifold projections), MP-GBS filters out "spurious" sharpness directions that are artifacts of a specific norm choice. Mathematically, the update rule in Eq. (13):

$$W_{t+1} = W_t - \eta(\nabla_{\mathcal{D}} \mathcal{L}(W_t) + \nabla^2_{\mathcal{D}} \mathcal{L}(W_t) d^*) \tag{49}$$

can be rewritten as a gradient descent step on the regularized objective:

$$\mathcal{L}_{MP-GBS}(W) \approx \mathcal{L}(W) + \lambda_{robust} ||H||_{spectral} \tag{50}$$

where $||H||_{spectral}$ is the spectral norm of the Hessian. By explicitly suppressing the ascent along d^*, MP-GBS minimizes the worst-case curvature, thereby guaranteeing convergence to a flatter region with better generalization properties.

B. Detail information of neural networks and baselines

To evaluate the versatility and scalability of the proposed TSCD framework, we conducted experiments across a diverse set of neural network architectures. Our selection covers a wide spectrum of inductive biases, ranging from standard Convolutional Neural Networks (CNNs) to modern Vision Transformers (ViTs) and emerging State Space Models (SSMs). The detailed specifications of these neural networks, including their parameter counts and ImageNet Top-1 accuracy baselines, are summarized in Table 6.

Convolutional Architectures. We selected **ResNet-50** and **ResNeXt-50** as representative standard baselines due to their widespread adoption in the community. To assess performance on modern, architecture-search-optimized models, we included **RegNetY** and **EfficientNetV2-S**. We also incorporated **ConvNeXt-Tiny** to represent the state-of-the-art in pure ConvNet design, and **ShuffleNetV2** to verify the effectiveness of our method under lightweight, mobile-oriented constraints.

Transformer and Emerging Architectures. Beyond traditional CNNs, we validated our method on **ViT-S/16** and **DeiT-S** to demonstrate compatibility with patch-based self-attention mechanisms. Furthermore, to test the robustness of our energy-based dyadic updates on non-traditional computation graphs, we incorporated cutting-edge architectures such as **Vision Mamba** (representing State Space Models) and **CKAN** (Convolutional Kolmogorov-Arnold Networks).

Table 6. neural network architectures used in experiments.

Architecture	# Params (M)	ImageNet Top-1 (%)
ResNet-50	25.6	76.13
ResNeXt-50 32×4d	25.0	81.20
RegNetY-3.2GF	19.4	81.98
ConvNeXt-Tiny	28.6	82.52
EfficientNetV2-S	21.5	84.23
ShuffleNetV2 2.0×	7.4	76.23
ViT-S/16	22.1	81.39
DeiT-S	22.1	81.17
Vision Mamba-S	26.3	80.45
CKAN-S	18.7	78.32

This diverse selection ensures that the performance gains observed in our experiments are attributed to the proposed algorithmic innovations (TSCD, MP-GBS, TF-GVS) rather than being tailored to a specific network topology.

We compare against the following methods, organized chronologically:

Forward-Forward (FF) (2022): The original goodness-based forward learning algorithm that trains networks layer-by-layer using positive and negative samples, without backpropagating errors through the network. FF uses the sum of squared activations as the goodness function and trains each layer to have high goodness for positive data and low goodness for negative data.

PEPITA (2022): A perturbation-based forward learning method that modulates inputs with error-carrying perturbations. PEPITA propagates errors forward through modulated inputs rather than backward through weights, providing a biologically plausible alternative to BP.

SoftHebb (2022): A soft Hebbian learning approach that combines winner-take-all competition with soft assignment, enabling deep learning without weight transport. SoftHebb demonstrates competitive performance on image classification tasks using purely local learning rules.

SymBa (2023): Symmetric backpropagation-free contrastive learning that addresses the weight transport problem by using symmetric feedback connections. SymBa achieves improved performance through balanced positive and negative learning signals.

TFF (2023): Trifecta techniques for deeper Forward-Forward networks, incorporating three key improvements: progressive layer training, improved negative sample generation, and adaptive threshold mechanisms to enable training of deeper networks.

Dual Propagation (DP) (2023): A non-iterative dual-state learning algorithm that maintains two network states (positive and negative) and updates weights based on the difference between states. DP provides a simplified forward learning framework without requiring iterative inference.

CFSE (2024): Channel-wise feature separation for Forward-Forward learning that improves feature discrimination by encouraging different channels to capture different aspects of the input, leading to more informative representations.

Distance-Forward (DF) (2024): A metric learning-based forward learning approach that replaces the goodness function with distance-based objectives. DF introduces N-pair margin loss and achieves state-of-the-art performance among forward learning methods. DF-R denotes the pure local variant, while DF-O incorporates inter-layer optimization.

Self-Contrastive Forward-Forward (SCFF) (2025): Self-supervised contrastive forward learning that leverages data augmentation to generate positive pairs and uses contrastive objectives within the forward-forward framework. SCFF demonstrates strong performance on semi-supervised learning tasks.

Backpropagation (BP): Standard end-to-end training with gradient backpropagation, serving as the upper bound reference.

Table 7. Summary of datasets used in our experiments across diverse domains.

Domain	Dataset	# Train	# Test	# Classes
Natural	CIFAR-10	50,000	10,000	10
	CIFAR-100	50,000	10,000	100
	Tiny ImageNet	100,000	10,000	200
	Food-101	75,750	25,250	101
Texture	DTD	1,880	1,880	47
Surface Defect	NEU Surface Defect	1,440	360	6
Remote Sensing	EuroSAT	18,900	8,100	10
Plant	PlantVillage	44,343	11,105	39
Astronomy	Galaxy10 DECals	15,962	1,774	10
Medical	BreakHis 40×	1,398	606	2
	BreakHis 100×	1,458	632	2
	BreakHis 200×	1,411	611	2
	BreakHis 400×	1,276	553	2

C. Dataset Specifications and Implementation Details

In this section, we provide detailed descriptions of the diverse datasets used to benchmark the Tri-Stream Coupled Dynamics (TSCD). As summarized in Table 7, our evaluation covers a wide spectrum of domains to verify the robustness of the proposed dyadic energy-based learning framework.

C.1. Domain-Specific Dataset Descriptions

Natural Images. We employ standard benchmarks to evaluate general visual recognition capabilities. **CIFAR-10** and **CIFAR-100** serve as fundamental benchmarks for low-resolution recognition. **Tiny ImageNet** (a subset of ImageNet) and **Food-101** introduce higher complexity with more classes and fine-grained distinctions.

Texture and Surface Defects. To test the model's ability to capture structural patterns without clear object geometry, we use the **DTD (Describable Textures Dataset)**. Furthermore, the **NEU Surface Defect** dataset represents a real-world industrial application, challenging the model to detect subtle anomalies on steel surfaces.

Remote Sensing and Astronomy. **EuroSAT** consists of Sentinel-2 satellite images covering 13 spectral bands, testing the model on earth observation tasks. **Galaxy10 DECals** moves the domain to astronomy, requiring the classification of galaxy morphologies from deep space surveys.

Plant and Medical Imaging. For fine-grained biological classification, we utilize **PlantVillage** for crop disease diagnosis. In the medical domain, we evaluate on the **BreakHis** breast cancer histopathology dataset. Following standard protocols, we report results across four magnification factors ($40\times$, $100\times$, $200\times$, $400\times$) to assess the model's sensitivity to multi-scale pathological features.

C.2. Preprocessing and Augmentation

All images were resized to a unified resolution (e.g., 224×224 or 32×32, depending on your specific setting) to facilitate batch training. We applied standard data augmentation techniques including random horizontal flipping and random cropping. For the medical and texture datasets, no color jittering was applied to preserve domain-specific spectral features.

D. More experiments

D.1. Results on Standard Benchmarks STL-10 and Tiny ImageNet

Table 8 extends our evaluation to semi-supervised learning STL-10 and larger-scale classification Tiny ImageNet. TSCD achieves 86.72% on STL-10, improving over DF-O by 7.87 points. On Tiny ImageNet, TSCD reaches 48.35% Top-1 and 72.58% Top-5 accuracy, surpassing DF-O by approximately 10 points on both metrics. These results demonstrate that the

Table 8. **Results on STL-10 and Tiny ImageNet.** STL-10 uses semi-supervised protocol with 5,000 labeled and 100,000 unlabeled images. Tiny ImageNet reports both Top-1 and Top-5 accuracy. Results for SCFF are from the original paper; other baselines are reproduced.

Method	STL-10	Tiny ImageNet	
	Top-1	Top-1	Top-5
BP	93.25	65.82	85.45
SoftHebb (Moraitis et al., 2022)	76.20	–	37.00
SCFF (Chen et al., 2025)	77.30±0.12	35.67±0.42	59.75±0.18
DF-O† (Wu et al., 2024)	78.85±0.32	38.52±0.48	62.85±0.35
DP† (Høier et al., 2023)	74.52±0.45	32.78±0.52	56.42±0.42
TSCD (Ours)	**86.72**±0.22	**48.35**±0.32	**72.58**±0.25
Improvement over best FL	+7.87	+9.83	+9.73

Table 9. **Effect of norm pair selection in MP-GBS on CIFAR-100.** We ablate the choice of norm pairs $\mathcal{P}$ in the multi-plane flatness objective.

Norm Pairs $\mathcal{P}$	Accuracy (%)	Δ
No bias optimization (Baseline)	58.42	–
$(2, 2)$ only (Standard SAM)	61.85	+3.43
$(1, \infty)$ only	60.78	+2.36
$(\infty, 1)$ only	60.25	+1.83
$(1, \infty), (\infty, 1)$	63.52	+5.10
$(2, 2), (1, \infty), (\infty, 1)$	65.28	+6.86
All 6 pairs (Full MP-GBS)	**67.45**	**+9.03**

benefits of TSCD scale effectively to more challenging scenarios.

D.2. Results on Medical Imaging (BreakHis)

Table D.2 presents results on breast cancer histopathology. TSCD achieves **87.89%** average accuracy (92.1% of BP), with CNN architectures (ResNet, EfficientNet) exceeding 92% while Transformers reach 79–82%. Performance slightly decreases at higher magnifications (400×) due to the finer-grained features required.

neural network	40×			100×			200×			400×			Average		
	BP	DF	**Ours**	BP	DF	**Ours**	BP	DF	**Ours**	BP	DF	**Ours**	BP	DF	**Ours**
ResNet-50	97.91	85.23	**91.67**	99.53	87.45	**93.78**	99.22	86.78	**93.12**	98.44	84.12	**90.89**	98.78	85.90	**92.37**
ResNeXt-50	99.46	87.12	**93.45**	99.22	86.78	**93.12**	99.34	87.23	**93.56**	98.05	83.45	**90.23**	99.02	86.15	**92.59**
RegNetY-3.2GF	99.84	88.34	**94.56**	99.22	86.89	**92.89**	99.48	87.56	**93.89**	98.02	83.12	**89.78**	99.14	86.48	**92.78**
ConvNeXt-Tiny	95.10	80.12	**86.89**	90.73	75.34	**82.45**	88.59	73.12	**80.34**	88.72	73.45	**80.67**	90.79	75.51	**82.59**
EfficientNetV2-S	99.44	87.67	**93.89**	99.42	87.23	**93.45**	99.11	86.45	**92.78**	99.06	85.67	**92.12**	99.26	86.76	**93.06**
ShuffleNetV2 2.0×	99.38	85.78	**92.34**	98.59	84.56	**91.12**	99.22	86.12	**92.56**	98.02	82.89	**89.45**	98.80	84.84	**91.37**
ViT-S/16	92.34	76.23	**83.45**	93.89	78.12	**85.23**	90.00	74.12	**81.12**	82.61	67.23	**74.56**	89.71	73.93	**81.09**
DeiT-S	93.56	77.89	**84.78**	94.67	79.45	**86.34**	91.23	75.56	**82.56**	84.12	69.12	**76.45**	90.90	75.51	**82.53**
Vim-S	91.23	74.89	**82.12**	92.56	76.89	**84.12**	88.67	72.89	**79.89**	81.23	65.89	**73.12**	88.42	72.64	**79.81**
CKAN-S	89.78	73.12	**80.23**	90.89	74.78	**82.12**	86.45	70.56	**77.67**	79.12	63.67	**70.89**	86.56	70.53	**77.73**
Avg. Gap$_{BP}$	–	-14.23	*-7.12*	–	-14.12	*-6.89*	–	-15.34	*-7.78*	–	-17.45	*-9.67*	–	-15.29	*-7.87*

D.3. More Detail Ablation Study

D.3.1. Detailed Analysis of MP-GBS

How do different norm pair selections affect MP-GBS performance?

Table 9 demonstrates that using all 6 norm pairs in MP-GBS consistently outperforms any subset. The complementary nature of different norm pairs captures diverse aspects of loss landscape geometry, validating our theoretical motivation for multi-plane optimization.

Table 10 shows that the optimal perturbation radius is $\rho = 0.05$. Smaller values provide insufficient exploration of the loss landscape, while larger values cause optimization instability.

Table 10. **Effect of perturbation radius ρ in MP-GBS.**

ρ	0.01	0.02	0.05	0.10	0.20
CIFAR-10	92.85	94.28	**95.78**	94.52	91.25
CIFAR-100	63.52	65.85	**67.45**	65.28	60.82

Table 11. **Effect of TF-GVS on gradient variance and accuracy.** Gradient proxy variance ($\times 10^{-4}$) measured during training on CIFAR-100.

Architecture	Gradient Variance		Accuracy (%)	
	w/o TF-GVS	w/ TF-GVS	w/o TF-GVS	w/ TF-GVS
10-layer CNN	14.82	**4.45**	63.28	**67.45**
ResNet-18	16.25	**4.88**	65.52	**69.78**
VGG-11	13.78	**4.12**	64.85	**68.52**
Avg. Reduction	**70.2%**		**+4.03%**	

D.3.2. Detailed Analysis of TF-GVS

How effectively does TF-GVS reduce gradient proxy variance, and what is the optimal window size?

Table 11 confirms that TF-GVS reduces gradient proxy variance by an average of 70.2% across architectures, leading to consistent accuracy improvements (+4.03% average).

Table 12 shows that $M = 3$ achieves optimal accuracy-efficiency trade-off. Larger windows reduce variance further but introduce staleness in gradient estimates, leading to decreased performance.

D.4. Robustness to Hardware Noise.

Table 13 evaluates robustness to hardware-related noise. While TSCD shows slightly lower clean accuracy than BP, it demonstrates superior stability under high-noise conditions. Specifically, under L4 device mismatch, TSCD achieves **60.45%**, significantly surpassing BP at 48.52% and DF-O at 52.28%. We attribute this enhanced robustness to the multi-plane optimization, which cultivates highly discriminative and noise-tolerant representations.

E. Limitation Analysis and Benchmarking against Backpropagation

While TSCD establishes a new state-of-the-art for forward learning, it is crucial to rigorously analyze its limitations compared to the ideal upper bound provided by Backpropagation (BP). In this section, we analyze the performance gap across different domains and investigate the data efficiency of the forward learning paradigm.

E.1. Performance Gap across Domains

Table 14 summarizes performance across datasets. TSCD recovers **89.6%** of BP on average, ranging from 73.5% (Tiny ImageNet) to 99.3% (CIFAR-10). Structured pattern tasks (NEU Surface, PlantVillage, BreakHis) achieve over 93% of BP, while fine-grained multi-class tasks show larger gaps.

E.2. Data Efficiency Analysis

A key open challenge for biologically plausible algorithms is sample efficiency. We evaluated TSCD under reduced training data regimes (10% and 1%) to assess its robustness to data scarcity.

As shown in Table 15, forward learning methods exhibit higher sensitivity to data volume compared to BP. Mitigation strategies include strong augmentation (+4.2% at 1% data), self-distillation between streams (+2.8%), and progressive class introduction. However, forward learning fundamentally requires dense manifold sampling; we recommend TSCD for scenarios with $\geq$10% typical data availability.

Diminishing Returns of Optimization. We further analyzed the contribution of our optimization strategies (MP-GBS and TF-GVS) under data constraints (Table 16). While optimization consistently improves performance, the magnitude of the gain decreases as data becomes scarcer (+5.56% at 100% data vs. +3.23% at 1% data). This indicates that while

Table 12. **Effect of TF-GVS window size M on CIFAR-100.**

M	1 (None)	2	3	5	10
Variance ($\times 10^{-4}$)	14.82	7.85	**4.45**	3.92	3.28
Accuracy (%)	63.28	65.52	**67.45**	66.85	64.52
Time/Epoch (s)	48.5	50.2	52.8	58.5	72.8

Table 13. **Robustness to hardware-related noise on CIFAR-10.** L1-L5 denote increasing noise severity levels following (Wu et al., 2024).

Noise Type	**Method**	**Clean**	**L1**	**L2**	**L3**	**L4**
Device Mismatch	BP	96.45	88.52	78.85	65.28	48.52
	DF-O	88.15	82.45	74.52	64.85	52.28
	TSCD	95.78	90.25	82.85	72.52	60.45
Impulse Noise	BP	96.45	91.28	84.52	74.85	62.25
	DF-O	88.15	84.52	78.85	70.28	59.52
	TSCD	95.78	92.45	87.25	79.52	68.85
Shot Noise	BP	96.45	92.85	87.52	80.25	70.52
	DF-O	88.15	85.78	81.25	74.52	65.28
	TSCD	95.78	93.52	89.85	83.28	74.52

optimization helps navigate the landscape, it cannot fully compensate for the lack of dense data samples required to shape the forward energy manifold.

E.3. Computational Cost Analysis

While TSCD significantly improves forward learning accuracy, it introduces computational overhead due to the tri-stream framework and the iterative settling phases required for dyadic neurons. Table 17 provides a detailed comparison of training time, memory usage, and parameter counts on CIFAR-100 using a ConvNeXt-Tiny neural network.

- **Training Time:** TSCD requires approximately $2.9\times$ the training time of BP. This latency stems from the multi-step dyadic state updates and the periodic cross-stream fusion process. However, this is a known trade-off for energy-based models (like Equilibrium Propagation) which sacrifice speed for local, biologically plausible updates.
- **Parameter Efficiency:** The parameter count increases by $3\times$ due to the maintenance of three separate streams (W^P, W^N, W^C). While seemingly large, these weights operate independently during the forward pass, meaning the *inference* cost (using only the Positive Stream) remains identical to a standard network.
- **Optimization Overhead:** The proposed strategies (MP-GBS and TF-GVS) add roughly 25% overhead to the base TSCD time. Given the +5.56% accuracy gain, we consider this a favorable trade-off within the constraints of forward learning.

Table 14. **Detailed breakdown of performance gaps (Full Data).** Top-1 accuracy (%) and ratio to BP.

Dataset	**BP**	**DF**	**Ours**	**Gap$_{BP}$**	**Ratio**
CIFAR-10	96.45	88.15	**95.78**	-0.67	99.3%
CIFAR-100	78.52	59.01	**67.45**	-11.07	85.9%
Tiny ImageNet	65.82	38.52	**48.35**	-17.47	73.5%
Food-101	88.02	68.73	**75.05**	-12.97	85.3%
DTD	68.69	53.75	**59.36**	-9.33	86.4%
NEU Surface	98.58	86.93	**91.97**	-6.61	93.3%
EuroSAT	98.21	85.82	**90.64**	-7.57	92.3%
PlantVillage	99.62	88.88	**93.46**	-6.16	93.8%
Galaxy10	83.12	64.72	**71.05**	-12.07	85.5%
BreakHis (Avg.)	94.14	80.82	**87.89**	-6.25	93.4%
Overall Avg.	**87.12**	**71.53**	**78.10**	**-9.02**	**89.6%**

Table 15. **Data Efficiency Comparison.** Top-1 Accuracy (%) with reduced training data on CIFAR-10 and CIFAR-100.

neural network	BP (Upper Bound)			Distance-Forward			TSCD (Ours)		
	100%	10%	1%	100%	10%	1%	100%	10%	1%
CIFAR-10									
ConvNeXt-Tiny	96.48	91.06	72.22	61.78	45.12	28.34	**68.12**	**51.45**	**34.78**
RegNetY-3.2GF	96.82	90.55	73.71	62.12	45.67	28.89	**68.56**	**52.01**	**35.34**
EfficientNetV2-S	96.42	89.44	77.06	61.56	45.23	31.12	**67.89**	**51.67**	**37.45**
ViT-S/16	95.12	85.64	39.77	48.45	32.12	18.23	**55.34**	**38.89**	**24.56**
CIFAR-100									
ConvNeXt-Tiny	82.60	67.47	37.38	33.67	18.12	8.56	**39.45**	**23.34**	**12.12**
RegNetY-3.2GF	82.89	64.40	36.56	34.23	18.67	8.89	**40.12**	**23.89**	**12.67**
EfficientNetV2-S	83.20	72.97	48.42	34.78	22.34	12.12	**40.56**	**27.45**	**15.89**
ViT-S/16	80.23	48.50	25.62	24.89	12.12	5.45	**31.12**	**17.78**	**9.23**

Table 16. Optimization contribution at different data sizes (CIFAR-100, ConvNeXt-Tiny).

Component	100% Data	10% Data	1% Data
Base TSCD	33.89	18.56	8.89
+ MP-GBS	+2.34	+2.56	+1.78
+ TF-GVS	+1.67	+1.89	+1.23
+ Synergy	+1.55	+0.33	+0.22
Total Gain	**+5.56**	**+4.78**	**+3.23**

E.4. Hypotheses on Fundamental Performance Barriers

Our comprehensive experiments reveal despite TSCD's SOTA performance, a persistent gap remains compared to backpropagation. We hypothesize that this stems from five structural impediments intrinsic to the forward learning paradigm:

- **The Local Credit Assignment Problem:** Unlike BP, which propagates global error signals $\frac{\partial \mathcal{L}}{\partial W}$ from the output back to the input, forward learning relies on layer-wise local objectives. This creates a "myopic" optimization process where early layers extract features that maximize local goodness but may not be optimal for the final downstream classification task.
- **Manifold Optimization Landscape:** Forward learning operates on a fundamentally different loss landscape. The objective of "pushing down" negative samples creates a highly non-convex energy surface with numerous local minima, whereas BP optimizes a smoother, goal-directed supervised loss.
- **Sensitivity to Negative Sampling:** The quality of learning is strictly bounded by the quality of the negative samples $\mathcal{D}^-$. Unlike BP, which uses ground-truth labels directly, forward learning requires the network to implicitly learn the data manifold boundary via contrastive examples. Poorly constructed negative samples (e.g., those too far from the manifold) provide negligible learning signals (vanishing gradients).
- **Information Propagation Bottleneck:** Without a backward pass to carry high-level semantic information (e.g., class relationships) to lower layers, the network suffers from an information bottleneck. Deep layers act as "unsupervised feature extractors" rather than "task-specific feature learners."

Table 17. **Computational Efficiency Benchmark.** Metrics are measured on CIFAR-100 with a ConvNeXt-Tiny neural network (Batch Size 128, Single NVIDIA A100 GPU). TSCD requires more resources to maintain the dual internal states and the auxiliary cross-fusion stream.

Method	Time/Epoch	Memory	Params	Acc.
BP (Baseline)	12.3s	3.2 GB	28.6M	82.60%
FF	18.7s	2.4 GB	28.6M	27.12%
DF	21.2s	2.8 GB	28.6M	33.67%
TSCD (Base)	28.4s	4.8 GB	85.8M	33.89%
+ MP-GBS	34.2s	5.1 GB	85.8M	36.23%
+ TF-GVS	29.8s	5.2 GB	85.8M	35.56%
TSCD (Full)	35.6s	5.4 GB	85.8M	39.45%

- **Zero-Order Approximation Bias:** Even with variance reduction techniques like TF-GVS, the gradient proxy derived from dyadic states is fundamentally a zero-order estimation. The approximation error $\mathcal{O}(\gamma)$ accumulates across layers, leading to less precise weight updates compared to the analytical gradients of BP.