Dynamic Representation Modeling for Federated Medical Image Domain Generalization

Yuxi Ma[1*] **Lin Zhao**[2*] **Jing Yang**[1] **Jiacheng Wang**[3 †] **Liansheng Wang**[1,2†]

[1] National Institute for Data Science in Health and Medicine, Xiamen University, Xiamen, China
[2] Department of Computer Science at the School of Informatics, Xiamen University, Xiamen, China
[3] Manteia Technology, Co., Ltd., Xiamen, China

Abstract

Federated Domain Generalization (FDG) aims to learn a global model robust to heterogeneous domain shifts without sharing raw data, a critical challenge in multi-center medical imaging. Most existing methods implicitly assume static inter-domain discrepancies, overlooking the fact that client representations evolve continuously during local optimization. We identify this phenomenon as representation dynamics, which refers to the temporal evolution of latent features induced by training updates rather than shifts in underlying data distributions. Such dynamics often lead to unstable aggregation, particularly in heterogeneous medical imaging scenarios. To address this issue, we propose Dynamic Knowledge Tracked FDG (DKT-FDG), a framework that explicitly tracks these dynamics by leveraging flow matching to model continuous representation trajectories. By aligning representation evolution instead of static snapshots, DKT-FDG improves robustness to domain shifts. Comprehensive experiments on multiple multi-center medical benchmarks demonstrate consistent improvements over state-of-the-art FDG methods, highlighting the importance of modeling representation dynamics in federated learning.

1 Introduction

Federated learning (FL) enables collaborative model training across decentralized institutions without sharing raw data, providing a privacy-preserving solution for distributed optimization [1–3]. It has attracted growing research interest in medical imaging, where strict data confidentiality and patient privacy regulations pose major challenges [4–6]. However, significant inter-institutional heterogeneity in acquisition protocols and population demographics induces domain shifts across clients, leading to systematic bias and unstable optimization, which further undermines cross-institutional generalization and clinical transferability.

To mitigate this issue, Federated Domain Generalization (FDG) aims to improve robustness to domain shifts by suppressing domain-induced biases during FL. Existing FDG approaches typically align latent representations across domains or design adaptive aggregation strategies to reduce client-specific effects. Representative methods include influence-aware aggregation [7], optimization-aware alignment [8], and representation disentanglement of domain-invariant and domain-specific components [9], as well as fairness-aware and personalized FL variants [10–12].

Despite their differences, most existing FDG methods implicitly rely on a *static alignment* paradigm. Specifically, inter-domain discrepancies are treated as fixed or round-wise independent, and representations are aligned based on instantaneous snapshots at each global aggregation. Crucially, this

*These authors contributed equally to this work.
†Corresponding authors.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

paradigm overlooks the continuous nature of federated optimization. Each client performs multiple local updates, inducing client-specific trajectories of latent representation evolution. Consequently, even when underlying data distributions remain stationary, inter-client representation discrepancies become time-dependent quantities: each client's local optimization unfolds the domain-specific geometric structure of its data distribution along a distinct trajectory in latent space, causing these trajectories to naturally diverge across clients. By restricting alignment to discrete communication rounds, static approaches fail to rectify the inherent divergence in client-specific update directions, leaving the underlying temporal inconsistency unresolved.

Addressing this limitation requires moving beyond the static alignment. We define ***representation dynamics*** as the temporal evolution of latent representations induced by local optimization, which constitutes an intrinsic and unavoidable property of FL (as shown in Appendix A), arising from the optimization process itself and persisting even when underlying data distributions remain stationary. Crucially, in the FDG setting, these dynamics determine whether inter-client representation discrepancies reflect genuine domain differences that warrant alignment or transient optimization artifacts that should be smoothed, a distinction that static alignment methods are fundamentally unable to make. We substantiate this claim in Appendix B, which analyzes why static alignment is structurally ill-suited to handle representation dynamics and derives the design rationale for each component of our framework.

Motivated by this perspective, we propose *Dynamic Knowledge Tracked Federated Domain Generalization (DKT-FDG)*, a framework that explicitly models representation dynamics to mitigate trajectory divergence ignored by static FDG methods. At the client side, DKT-FDG summarizes representation evolution via a Dynamic Flow Signature Module (DFSM), which models trajectories through a learned velocity field to capture their underlying geometry beyond coarse temporal approximations. At the server side, a Knowledge Tracking Strategy (KTS) leverages these dynamics to enable trajectory-aware aggregation. By coupling trajectory-aware representation modeling with temporally adaptive aggregation, DKT-FDG enables more stable federated optimization under domain heterogeneity.

Our contribution lies in the formulation rather than in the parameterization of any individual module. Architectural novelty would consist in a new design for the flow head, and our flow head deliberately does not pursue this: it is built from standard MLP and SiLU blocks so that the added cost stays local and small. What changes is the object the server receives and the use to which it is put. In prior FDG work this object is a static summary of each client, whether model parameters, gradients, or a feature statistic computed once per round, and alignment operates on that summary. In DKT-FDG each client fits a continuous velocity field to its own latent distribution, reads out a compact signature of the resulting geometry, and the server maintains this signature across communication rounds, so that aggregation is conditioned on how a client's representation state is evolving rather than on where it currently sits. Fig. 3(b) isolates where the gain originates: holding the pipeline fixed and replacing only the learned velocity field with a static difference or with a linear interpolation costs 4.55 and 2.45 accuracy points on Fed-ISIC respectively.

Our main contributions are summarized as follows: (i) We characterize the limitations of static alignment in FDG by formalizing representation dynamics as an inherent property of federated optimization. (ii) We propose DKT-FDG, which tracks representation dynamics by learning canonical distributional descriptions of each client's latent space via flow matching, and enables trajectory-aware aggregation through dynamic knowledge tracking, with each component motivated by the dynamic mismatch analysis. (iii) Extensive experiments on multi-center medical image classification and segmentation benchmarks demonstrate consistent improvements over state-of-the-art FDG methods, validating the necessity of dynamic modeling in FL.

2 Related Work

FDG extends the paradigm of FL [1, 13, 14] to scenarios where heterogeneous client data induce significant domain shifts. Existing approaches can be broadly categorized into three directions: *Fairness-oriented*, *Representation-based*, and *Adaptive and knowledge-aware*.

Fairness-oriented methods, such as FedHEAL [10] and FedDG-GA [15], aim to mitigate aggregation bias and update conflicts caused by domain skew. FedHEAL [10] introduces selective update suppression and a fairness-constrained optimization to prevent dominant domains from overwhelming weaker

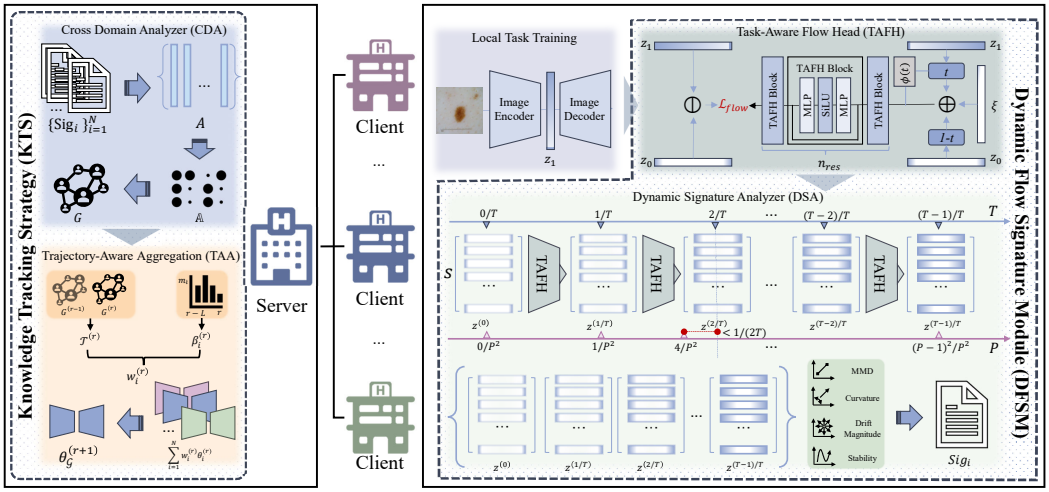

Figure 1: **Overview of the DKT-FDG framework.** It decouples federated optimization into: **(Right) DFSM:** Captures local representation dynamics by learning a continuous flow field v_θ conditioned on time $\phi(t)$. The DSA simulates latent trajectories (blue axis) and extracts compact signatures Sig_i using a quadratic sampling schedule (pink axis) to prioritize early-stage dynamics. **(Left) KTS:** Tracks domain interactions by constructing a sparse affinity graph G via CDA. TAA computes adaptive weights $w_i^{(r)}$ based on topological stability and performance trends, ensuring robust knowledge fusion. (Visualizations of latent evolution are provided in Appendix F).

ones, while FedDG-GA [15] dynamically re-weights client contributions based on generalization theory to tighten the generalization bounds. FedDI [7] further assigns dynamic aggregation weights via data influence analysis. Although these strategies adaptively reweight client contributions, they rely on round-wise static snapshots and neglect the temporal dynamics of local representation evolution. Representation-based approaches seek to align domain features to enhance generalization. FDSE [9] decomposes the network into Domain-agnostic Feature Extractor (DFE) and Domain-specific Skew Eraser (DSE) to iteratively remove client-specific biases. FedOMG [8] maximizes gradient inner products across clients to align optimization directions without extra communication. Despite their effectiveness in harmonizing representations, these methods lack mechanisms to model the continuous temporal evolution of representations across training rounds. Adaptive and knowledge-aware frameworks, such as FEAL [12] and APFL [11], further enhance generalization by integrating uncertainty modeling and analytic personalization. FEAL [12] employs evidential uncertainty to calibrate data informativeness under domain shifts, and APFL [11] decouples global generalization from local refinement via a dual-stream least-squares formulation. While improving robustness to heterogeneity, these designs lack explicit mechanisms to track how client representations evolve across rounds, leaving cross-round temporal consistency unaddressed.

3 Method

3.1 Overview

We consider an FL setting with N clients, where each client $i \in \{1, \ldots, N\}$ holds a private local dataset $\mathcal{D}_i$ drawn from a specific domain distribution $\mathbb{P}_i$. In each communication round r, clients perform local optimization to update their parameters $\theta_i^{(r)}$. The overarching goal is to collaboratively learn a global model $\theta_{\mathcal{G}}^{(r)}$ that generalizes across heterogeneous domains without sharing raw data.

Unlike existing FDG methods that align domains based on static representation snapshots, DKT-FDG explicitly models the **representation dynamics**. As illustrated in Fig. 1, DKT-FDG consists of two coupled components: (i) **Client-side:** DFSM leverages flow matching to learn a deterministic vector field governing the transport of latent features. It encapsulates high-order kinetic dynamics into a compact *flow signature*, summarizing the trajectory of local representation evolution. (ii) **Server-side:** KTS constructs a dynamic domain topology to trace the correlations and stability of these client

trajectories. By quantifying the temporal consistency of evolutionary paths, KTS dynamically adapts aggregation weights to prioritize clients with stable optimization directions while attenuating transient fluctuations.

Through this mechanism, DKT-FDG effectively distinguishes persistent domain shifts from local training noise, thereby improving robust generalization. Crucially, the modular design allows DFSM to serve as a lightweight functional approximator rather than a heavy generative model. The full algorithm is detailed in Appendix C.

The design below follows from this formulation. Sec. 3.2 specifies what each client computes and transmits, namely a flow-based descriptor of its current representation state, and Sec. 3.3 specifies how the server consumes a sequence of such descriptors. The individual modules are intentionally standard; what is new is the object they produce and the way server-side aggregation is conditioned on its evolution.

3.2 Dynamic Flow Signature Module

DFSM provides a continuous-time model of each client's latent distribution at the current round, from which a compact geometric descriptor is read out. The continuous time variable belongs to this transport construction and is not the time axis of local optimization; the quantity that tracks optimization is the evolution of the descriptor across rounds, which Sec. 3.3 makes explicit.

3.2.1 Task-Aware Flow Head (TAFH)

To capture fine-grained representation dynamics, we propose the TAFH, a time-dependent neural network v_θ conditioned on Fourier temporal embeddings. TAFH learns a continuous probability path transporting a source distribution $p_0 = \mathcal{N}(0, I)$ (where I is the identity matrix) to the target empirical distribution p_1 of the client's latent features.

Fourier Temporal Embedding. To capture high-frequency temporal dynamics, we explicitly embed the scalar time $t \in [0, 1]$ into a high-dimensional vector via sinusoidal projection:

$$\phi(t) = \left[\sin(2^k \pi t),\, \cos(2^k \pi t)\right]_{k=0}^{d_t/2-1} \in \mathbb{R}^{d_t}, \tag{1}$$

where $k \in \{0, \ldots, d_t/2 - 1\}$ is the frequency index and d_t is the embedding dimension. This ensures TAFH maintains sensitivity to transient variations during transport.

We define the optimal transport path z_t via conditional linear interpolation, chosen for its deterministic nature and computational efficiency suitable for FL. Given a data sample $z_1 \sim p_1$ and a noise $z_0 \sim p_0$, the intermediate state is defined as:

$$z_t = (1 - (1 - \sigma_{\min})t)z_0 + tz_1, \tag{2}$$

where $\sigma_{\min}$ (e.g., 10^{-5}) ensures numerical stability at $t = 1$.

The flow field v_θ is parameterized by a deep neural network composed of n_{res} stacked residual blocks. First, the time embedding $\phi(t)$ is processed and concatenated with z_t, then mapped to the hidden dimension d_h via an initial projection layer to yield h_0. Subsequently, the hidden state evolves through residual blocks. For the l-th block, the update rule is explicitly defined as:

$$h_{l+1} = h_l + W_2^{(l)} \cdot \text{SiLU}\Big(W_1^{(l)} h_l\Big), \tag{3}$$

where $W_1^{(l)}$ and $W_2^{(l)}$ are learnable weight matrices, and SiLU denotes a non-linear activation function. This residual design ensures consistent gradient flow and enables the network to learn complex trajectory dynamics without vanishing gradients.

Following the flow matching principle [16], TAFH is optimized to regress the target velocity field by minimizing the expected mean squared error:

$$\mathcal{L}_{\text{flow}} = \mathbb{E}_{t,z_0,z_1}\left[\|v_\theta(z_t, t) - (z_1 - (1 - \sigma_{\min})z_0)\|_2^2\right]. \tag{4}$$

Minimizing $\mathcal{L}_{\text{flow}}$ ensures v_θ accurately characterizes the client-specific optimization trajectory.

Optimization Strategy. We adopt a two-stage local training strategy to ensure valid transport learning. First, the client updates the encoder and classifier via the task-specific loss $\mathcal{L}_{\text{task}}$ (e.g.,

Cross-Entropy or Dice loss). Second, we freeze the encoder to fix the latent distribution p_1 and update TAFH to minimize $\mathcal{L}_{\text{flow}}$.

To **minimize computational overhead**, we employ a *warm-start strategy*: TAFH is initialized with weights from the previous round, allowing it to adapt to incremental representational shifts via rapid fine-tuning.

3.2.2 Dynamic Signature Analyzer (DSA)

Although TAFH $v_\theta(z_t, t)$ provides a continuous description of representation dynamics, directly transmitting full latent trajectories across communication rounds is neither communication-efficient nor necessary. Instead, DSA summarizes the *essential dynamical properties* by numerically integrating the learned flow field.

We evolve S particles from $z^{(0)} \sim \mathcal{N}(0, I)$ via Euler integration over T uniform steps, with $t^{(n)} = n/T$ for $n \in \{0, \dots, T-1\}$:

$$v^{(n)} = v_\theta(z^{(n)}, t^{(n)}), \quad z^{(n+1)} = z^{(n)} + \tfrac{1}{T} v^{(n)}. \tag{5}$$

Rather than using all T steps for signature generation, which would introduce redundancy and dilute critical evolutionary phases, we define a coarser non-uniform schedule of P ($P < T$) sampling points $t^{(p)} = (p/P)^2$, $p = 1, \dots, P$, concentrating samples in the early phase of motion. The closest latent state at each $t^{(p)}$ is recorded as $z^{(n_p)}$, where $n_p = \lfloor T t^{(p)} \rfloor$, forming the simulated flow distribution:

$$\mathcal{P}_{\text{flow}} = \frac{1}{P} \sum_{p=1}^{P} \delta(z - z^{(n_p)}), \tag{6}$$

where $\delta(\cdot)$ is the Dirac delta function.

Rather than preserving the full trajectories, DSA extracts a compact set of statistics designed to capture fundamental aspects of optimization dynamics. These statistics are not chosen heuristically; rather, they correspond to first- and second-order descriptors of continuous optimization trajectories in latent space.

Specifically, we first quantify the **Representation Drift Magnitude** by computing the kinetic energy of the flow across the sampled points, serving as a proxy for the intensity of adaptation. Complementing this, we assess **Temporal Stability** through the smoothness of the velocity field changes, effectively distinguishing consistent optimization trends from transient stochastic fluctuations. To capture the geometric consistency of updates, we evaluate **Trajectory Curvature** via the orthogonality of consecutive velocity vectors; a lower curvature indicates a stable, directed optimization path. Finally, to preserve distributional fidelity, we compute the **Maximum Mean Discrepancy (MMD)** between the simulated flow endpoints and the real empirical features. These statistics jointly characterize whether a client's local optimization exhibits stable convergence or unstable drifting behavior. By aggregating them into a low-dimensional **flow signature** Sig_i, DSA preserves the most aggregation-relevant information while significantly reducing communication overhead. Formal definitions are provided in Appendix D.

3.3 Knowledge Tracking Strategy

While the DFSM models the evolution of local latent features, effective FL requires understanding how these domain distributions differ and interact across clients. To this end, we introduce the KTS for global aggregation, composed of the **Cross-Domain Analyzer (CDA)** and **Trajectory-Aware Aggregation (TAA)**. Unlike standard aggregation methods that perform round-wise greedy updates, KTS is designed to be **history-aware**, allowing the server to trace temporal correlations among evolving client representations. This design forms a dynamic topology over communication rounds, enabling the server to suppress unstable clients and emphasize consistent knowledge contributors based on long-term reliability.

3.3.1 Cross-Domain Analyzer (CDA)

The CDA aims to model the instantaneous compatibility among clients by analyzing similarities in their representation dynamics.

Given a set of flow signatures $\{\text{Sig}_i\}_{i=1}^N$ uploaded by N clients, each signature Sig_i is first standardized to remove scale bias:

$$\hat{s}_i = \text{Standardize}\big(\text{Sig}_i\big). \tag{7}$$

We then compute the pairwise Euclidean distances $D_{ij} = \|\hat{s}_i - \hat{s}_j\|_2$ between all clients. To construct a robust topology, a Gaussian kernel is applied to these distances to obtain a dense affinity matrix A:

$$A_{ij} = \exp\Big(-\tfrac{D_{ij}^2}{2\alpha_D^2+\epsilon}\Big), \tag{8}$$

where α_D denotes the standard deviation of all distances, and ϵ (e.g., 10^{-5}) is a small constant ensuring numerical stability.

To filter out noisy weak correlations and strictly preserve local manifold structures, we apply a rank-based sparsity constraint. Specifically, for each client, we retain only the connections to the top $\lfloor N/2 \rfloor$ most similar peers while setting others to zero, yielding a sparse affinity matrix $\mathbb{A}$. Finally, we symmetrize and degree-normalize $\mathbb{A}$ to construct the federated domain similarity graph:

$$G^{(r)} = D_G^{-\frac{1}{2}}\big(\mathbb{A} + \mathbb{A}^\top\big)D_G^{-\frac{1}{2}}, \quad G^{(r)} \in \mathbb{R}^{N\times N}, \tag{9}$$

where D_G is the diagonal degree matrix and the superscript (r) denotes the current communication round. Beyond serving as a basis for aggregation weights, this graph topology enables deeper structural analysis of domain interactions, which we elaborate on in Appendix E.

We adopt Euclidean distance and Gaussian affinity as a smooth and rotation-invariant approximation of local geometry in the flow signature space. This choice yields a stable and well-conditioned domain topology under heterogeneous representation dynamics. Unlike conventional feature-level similarity, this normalized graph $G^{(r)}$ encodes how different clients evolve their representations, thereby approximating the compatibility of local optimization trajectories across domains. Clients with similar flow signatures are tightly connected in $G^{(r)}$, suggesting transferable knowledge.

3.3.2 Trajectory-Aware Aggregation (TAA)

Single-round updates in FL are susceptible to stochastic noise arising from batch sampling and local overfitting. Equivalently, TAA can be interpreted as an entropy-regularized aggregation scheme under non-stationary domain dynamics, where the temperature adaptively controls the bias–variance trade-off across communication rounds [17, 18]. To ensure robust convergence, TAA tracks the *temporal consistency* of client behaviors over a sliding window $\mathcal{H}$ of length L. Theoretically, this mechanism acts as a **temporal low-pass filter**, suppressing high-frequency fluctuations caused by transient training instability while preserving persistent trend signals that reflect true domain reliability. TAA operates by jointly modeling client reliability from two complementary perspectives: (i) temporal consistency of local performance trends, and (ii) global stability of inter-domain relationships.

Instead of raw metrics, we assess clients based on their *evolutionary momentum*. Let $m_i^{(r)}$ denote the local training performance (e.g., Accuracy or Dice) of client i at round r. Over a sliding window, we estimate the performance trend $\beta_i^{(r)}$ as the slope of the least-squares linear fit to $\{m_i^{(r')}\}_{r'\in\mathcal{H}}$. A positive $\beta_i^{(r)}$ indicates consistent improvement, characterizing a reliable contributor.

To determine the aggregation weights $w_i^{(r)}$ of client i at round r, we employ a history-aware softmax. Crucially, the temperature of this softmax is dynamically adjusted based on the **Global System Instability** $\mathcal{J}^{(r)}$. We define $\mathcal{J}^{(r)}$ as the Frobenius norm of the temporal difference in the global domain similarity graph (defined in Eq. 9):

$$\mathcal{J}^{(r)} = \Big\| G^{(r)} - G^{(r-1)} \Big\|_F. \tag{10}$$

This metric $\mathcal{J}^{(r)}$ quantifies how drastically the inter-client relationships have shifted since the last round. We then define the adaptive weight $w_i^{(r)}$ as:

$$w_i^{(r)} = \exp(\beta_i^{(r)}/\tau_r)\Big/\textstyle\sum_{j=1}^N \exp(\beta_j^{(r)}/\tau_r), \quad \text{with } \tau_r = \gamma(\mathcal{J}^{(r)}), \tag{11}$$

where $\gamma(\cdot)$ is the sigmoid function. The term τ_r acts as an entropy regulator: a higher τ_r (large $\mathcal{J}^{(r)}$) yields a relatively more uniform weight distribution, while a lower τ_r sharpens the softmax toward the highest-trend clients.

Finally, the global model is updated via:

$$\theta_{\mathcal{G}}^{(r+1)} = \sum_{i=1}^{N} w_i^{(r)} \theta_i^{(r)}. \tag{12}$$

By coupling local improvement trends with global topological stability, TAA ensures robust optimization against domain shifts. This design prevents overconfident aggregation when domain relationships fluctuate and enables robust knowledge fusion over training rounds.

Table 1: Comparison of DKT-FDG with state-of-the-art FDG methods on multi-center benchmarks. For classification tasks (**Cls.**), we report Accuracy (Acc) and F1 score. For segmentation tasks (**Seg.**), we report Dice and 95th-percentile Hausdorff distance (Hd95). All p-values are computed via paired t-tests over five random seeds, comparing DKT-FDG against the best baseline. Results with $p < 0.05$ are considered statistically significant.

Method	Fed-ISIC (**Cls.**)		IPMN-T1W (**Cls.**)		IPMN-T2W (**Cls.**)		p-value
	Acc	F1	Acc	F1	Acc	F1	
Centralized	$0.8205_{\pm 0.0053}$	$0.8203_{\pm 0.0054}$	$0.5342_{\pm 0.0654}$	$0.5220_{\pm 0.0725}$	$0.5736_{\pm 0.0764}$	$0.5758_{\pm 0.0746}$	0.1112
FedAvg [1]	$0.7871_{\pm 0.0526}$	$0.7783_{\pm 0.0035}$	$0.4725_{\pm 0.0396}$	$0.4880_{\pm 0.0470}$	$0.5176_{\pm 0.0181}$	$0.5330_{\pm 0.0224}$	< 0.01
FEAL [12]	$0.7868_{\pm 0.1095}$	$0.7763_{\pm 0.0409}$	$0.3752_{\pm 0.0590}$	$0.3887_{\pm 0.0663}$	$0.4982_{\pm 0.0477}$	$0.4984_{\pm 0.0491}$	< 0.01
FedHEAL [10]	$0.7579_{\pm 0.0038}$	$0.7452_{\pm 0.0031}$	$0.4240_{\pm 0.0348}$	$0.4425_{\pm 0.0377}$	$0.4169_{\pm 0.0267}$	$0.4328_{\pm 0.0296}$	< 0.01
FedDG-GA [15]	$0.7884_{\pm 0.0542}$	$0.7766_{\pm 0.0029}$	$0.5458_{\pm 0.0384}$	$0.5736_{\pm 0.0410}$	$0.4436_{\pm 0.0238}$	$0.4371_{\pm 0.0314}$	< 0.01
APFL [11]	$0.7001_{\pm 0.0623}$	$0.6800_{\pm 0.0161}$	$0.4877_{\pm 0.0332}$	$0.4523_{\pm 0.0289}$	$0.5559_{\pm 0.0191}$	$0.4658_{\pm 0.0197}$	< 0.01
FDSE [9]	$0.6993_{\pm 0.0014}$	$0.6742_{\pm 0.0001}$	$0.4969_{\pm 0.0696}$	$0.4044_{\pm 0.0632}$	$0.5239_{\pm 0.0472}$	$0.4473_{\pm 0.0335}$	< 0.01
FedOMG [8]	$0.7156_{\pm 0.0039}$	$0.6547_{\pm 0.0401}$	$0.5366_{\pm 0.1100}$	$0.3939_{\pm 0.0929}$	$0.4512_{\pm 0.1051}$	$0.4568_{\pm 0.1419}$	< 0.01
FedDI [7]	$0.8147_{\pm 0.0045}$	$0.8061_{\pm 0.0047}$	$0.4697_{\pm 0.0216}$	$0.4878_{\pm 0.0310}$	$0.4847_{\pm 0.0470}$	$0.4932_{\pm 0.0399}$	< 0.01
Ours	$\mathbf{0.8370_{\pm 0.0109}}$	$\mathbf{0.8337_{\pm 0.0097}}$	$\mathbf{0.5741_{\pm 0.0601}}$	$\mathbf{0.5826_{\pm 0.0575}}$	$\mathbf{0.5753_{\pm 0.0437}}$	$\mathbf{0.5668_{\pm 0.0804}}$	-
Method	Fed-Fundus (**Seg.**)		Fed-Polyp (**Seg.**)		Fed-Prostate (**Seg.**)		p-value
	Dice	Hd95	Dice	Hd95	Dice	Hd95	
Centralized	$0.9055_{\pm 0.0072}$	$22.7154_{\pm 4.8706}$	$0.8669_{\pm 0.0119}$	$56.1862_{\pm 7.9716}$	$0.9344_{\pm 0.0069}$	$13.1403_{\pm 5.4148}$	0.1523
FedAvg [1]	$0.8616_{\pm 0.1392}$	$54.5016_{\pm 55.0645}$	$0.8284_{\pm 0.0734}$	$87.9998_{\pm 4.0791}$	$0.9156_{\pm 0.0054}$	$32.5477_{\pm 3.8448}$	< 0.05
FEAL [12]	$0.8618_{\pm 0.0124}$	$47.9252_{\pm 10.1428}$	$0.8174_{\pm 0.0027}$	$89.6915_{\pm 6.2164}$	$0.9047_{\pm 0.0065}$	$41.6815_{\pm 17.5955}$	< 0.05
FedHEAL [10]	$0.8397_{\pm 0.2045}$	$51.5909_{\pm 98.1862}$	$0.7743_{\pm 0.0123}$	$114.4351_{\pm 14.7424}$	$0.8497_{\pm 0.0174}$	$47.5847_{\pm 10.2871}$	< 0.05
FedDG-GA [15]	$0.8542_{\pm 0.1356}$	$50.4744_{\pm 51.4395}$	$0.8383_{\pm 0.0721}$	$83.0230_{\pm 4.4666}$	$0.9155_{\pm 0.0075}$	$33.7406_{\pm 6.9382}$	< 0.05
APFL [11]	$0.8461_{\pm 0.1679}$	$95.4528_{\pm 54.1809}$	$0.7136_{\pm 0.0061}$	$120.9772_{\pm 10.0481}$	$0.8517_{\pm 0.0328}$	$84.4674_{\pm 30.8031}$	< 0.05
FDSE [9]	$0.8495_{\pm 0.0629}$	$74.5673_{\pm 71.0382}$	$0.8429_{\pm 0.0208}$	$72.3540_{\pm 8.3111}$	$0.8984_{\pm 0.0543}$	$71.1427_{\pm 53.5549}$	< 0.05
FedOMG [8]	$0.7827_{\pm 0.0348}$	$58.1471_{\pm 12.8679}$	$0.7058_{\pm 0.0410}$	$129.8020_{\pm 14.1472}$	$0.8056_{\pm 0.0201}$	$60.3128_{\pm 13.0133}$	< 0.05
FedDI [7]	$0.8518_{\pm 0.1436}$	$52.7914_{\pm 40.4973}$	$0.8350_{\pm 0.0102}$	$83.0139_{\pm 8.6810}$	$0.9131_{\pm 0.0056}$	$40.3490_{\pm 11.3348}$	< 0.05
Ours	$\mathbf{0.8838_{\pm 0.1701}}$	$\mathbf{37.2861_{\pm 6.0333}}$	$\mathbf{0.8551_{\pm 0.0101}}$	$\mathbf{64.2208_{\pm 7.9496}}$	$\mathbf{0.9314_{\pm 0.0179}}$	$\mathbf{15.3572_{\pm 7.9913}}$	-

4 Experiments

4.1 Experimental Setup

Datasets: We evaluate DKT-FDG on five medical benchmarks. **Fed-ISIC** includes four dermoscopic sources [19] for skin lesion classification, differing in imaging devices and illumination conditions. **IPMN** comprises 652 **T1-weighted** (T1W) and 655 **T2-weighted** (T2W) pancreatic MRI scans from 7 centers [20], exhibiting pronounced scanner- and protocol-induced variations. **Fed-Polyp** integrates 4 endoscopic datasets [21–24] for colorectal polyp segmentation, characterized by diverse endoscope systems and visual environments. **Fed-Prostate** combines 6 prostate MRI cohorts [25–27], reflecting heterogeneity in magnetic field strengths and acquisition sequences. **Fed-Fundus** merges 3 retinal fundus datasets [28–30], showing variability in imaging systems, resolutions, and patient populations. Additional details are provided in the Appendix I.

Implementation: Following [12, 31], we adopt EfficientNet-B0 [32] for *Fed-ISIC*, DenseNet-121 [33] for *IPMN*, and U-Net [34] for segmentation benchmarks, with ImageNet preinitialization for classification backbones. DKT-FDG uses n_{res}=2, d_h=2048, S=300, T=50, P=10, L=3, with TAFH pretrained for 100 epochs in the first communication round and fine-tuned for 5 epochs in subsequent rounds. All results are averaged over five random seeds. For segmentation tasks, standard deviations are computed over samples within each client rather than across random seeds, reflecting intra-client annotation variability. Full details are in Appendix J.1.

4.2 Benchmark Comparison

Table 1 presents a comprehensive comparison between DKT-FDG and state-of-the-art FDG methods across multi-center classification and segmentation benchmarks.

Table 2: Computation-time and FLOPs analysis of DKT-FDG versus FedAvg.

Metric	Method	
	FedAvg [1]	**Ours**
Time (Seconds)		
Local Training	63.75	75.18
Global Aggregation	0.07	0.09
Inference	6.25	6.25
Computation (Millions)		
FLOPs	384.60	398.07

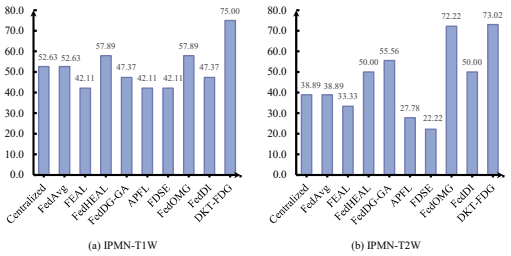

Figure 2: Comparison of model generalization performance on unseen clients in IPMN.

For the **classification tasks**, DKT-FDG consistently outperforms all compared baselines, achieving the best Accuracy and F1 scores on all datasets. Remarkably, DKT-FDG achieves performance comparable to the *Centralized* setting, which assumes full data aggregation. We attribute this to two factors: the DFSM provides the KTS with discriminative trajectory-aware signals that characterize each client's optimization geometry, enabling the server to distinguish persistent domain shifts from transient stochastic fluctuations; and the KTS adaptively re-weights clients according to their domain stability, producing a self-regularized ensemble that enhances global generalization. The observed improvement, therefore, reflects the benefit of temporal dynamic modeling rather than unequal training budgets. For the **segmentation tasks**, DKT-FDG consistently delivers the best performance among all FDG baselines across the three benchmark datasets, yielding notable gains in both Dice and Hd95 metrics. In particular, DKT-FDG substantially reduces the average Hd95, suggesting that dynamic knowledge tracking facilitates more accurate structural alignment and boundary preservation across domains.

Overall, these results confirm that DKT-FDG effectively mitigates domain discrepancy and achieves superior cross-domain generalization, outperforming FDG approaches. We note that several baselines underperform FedAvg on certain benchmarks. This is consistent with prior observations that methods designed for specific heterogeneity regimes can degrade under severe domain skew, where their alignment or reweighting mechanisms amplify client-specific biases rather than correct them.

Matched-budget controls confirming that this advantage is independent of training budget are provided in Appendix J.3.

4.3 Generalizability to Unseen Domains

A key objective of FDG is that the global model can generalize beyond the observed domains and perform reliably on previously unseen domains. To explicitly evaluate this capability, we conduct an *unseen-client validation* experiment on the real multi-center IPMN dataset, following [9]. In each experimental configuration, one client is entirely excluded from the training phase, while the remaining clients collaboratively train the global model using DKT-FDG. After training convergence, the global model is directly evaluated on the excluded client without any fine-tuning, thereby examining its intrinsic cross-domain generalization ability.

As shown in Fig. 2, DKT-FDG consistently achieves the highest performance across all unseen clients, demonstrating its capacity to capture transferable latent dynamics and mitigate domain shift. By dynamically tracking local feature evolution and incorporating these dynamics into global aggregation, DKT-FDG establishes a coherent mechanism for learning representations, enabling the model to generalize robustly across both training domains and unseen domains.

4.4 Computational Efficiency

We evaluate the computational efficiency of DKT-FDG on Fed-ISIC (Table 2). Per-round time is computed by dividing total wall-clock training time by the number of communication rounds.

The backbone requires 384.6M FLOPs per forward pass; introducing TAFH adds only 13.5M FLOPs, an overhead confined entirely to local client training. TAFH consists of lightweight fully connected layers and temporal adapters, contributing negligible impact on inference latency or global

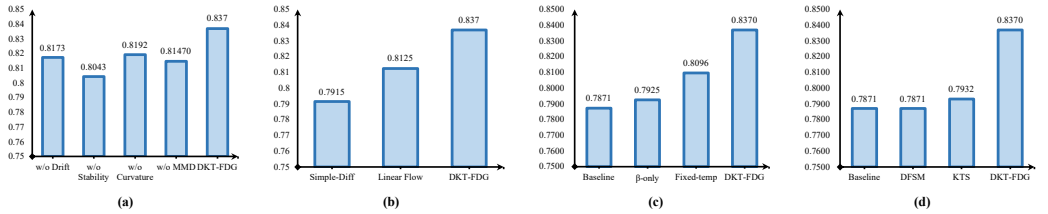

Figure 3: Ablation studies and sensitivity analysis of DKT-FDG on Fed-ISIC, evaluated by Acc. **(a)** Per-statistic ablation of DSA flow signatures; **(b)** Flow modeling alternatives; **(c)** CDA server-side component decomposition; **(d)** Component-wise ablation of DFSM and KTS;

aggregation cost. The resulting increase in per-round local training time (63.75s $\rightarrow$ 75.18s) reflects the frozen-encoder flow head stage, which adds no task optimization. The methods in Table 1 share this backbone, so their per-round cost is dominated by the same forward pass; what differs is where each method places its extra computation, whether in server-side aggregation, in inter-client exchange, or, as here, in local training, and these are not commensurable under a single forward-pass FLOPs count. Table 2 therefore reports our addition against the shared reference. Since TAFH is optimized locally and excluded from global synchronization, it incurs negligible communication overhead relative to FedAvg [1]. The overhead is confined to local training and is independent of client count, making DKT-FDG more scalable than methods requiring inter-client feature exchange. As detailed in Appendix H, DKT-FDG adheres to the principle of data minimization: transmitted flow signatures contain no raw features or labels, substantially limiting information exposure while remaining compatible with standard privacy-enhancing mechanisms such as differential privacy or secure aggregation.

4.5 Ablation Study

We conduct ablations to evaluate the contribution of each component (Fig. 3).

Signature Component Analysis. Fig. 3(a) shows that removing any statistic from the DSA signature degrades performance, with Temporal Stability causing the largest drop (3.27%). These statistics capture complementary aspects of trajectory dynamics, including adaptation intensity (drift), temporal variance (stability), directional consistency (curvature), and distributional fidelity (MMD), forming a compact multi-order descriptor.

Flow Modeling Alternatives. Fig. 3(b) compares DKT-FDG with simplified alternatives that remove explicit trajectory modeling. Simple-Diff ($z_{\text{post}} - z_{\text{pre}}$) corresponds to a path-invariant, static approximation and fails to capture temporal properties of optimization trajectories. Linear Flow introduces limited temporal structure via fixed interpolation but lacks a learned velocity field and thus cannot model trajectory geometry. DKT-FDG consistently outperforms both, indicating that gains cannot be attributed to coarse temporal information alone, but require explicit modeling of trajectory dynamics.

Server-side Aggregation. Fig. 3(c) shows that using only scalar performance trends (β-only) yields limited gains, while incorporating the CDA graph significantly improves performance by capturing cross-client structural relations. Adaptive temperature further stabilizes aggregation under dynamic system conditions.

Component-wise Ablation. Fig. 3(d) shows that DFSM or KTS alone provides limited benefit, while their combination achieves the best performance, confirming that trajectory modeling and structure-aware aggregation are complementary.

Sensitivity analyses are provided in Appendix J.2.

5 Conclusion and Future Directions

In this paper, we proposed **DKT-FDG**, a novel FDG framework that explicitly models *representation dynamics* via flow matching, addressing the critical limitation of static alignment in existing approaches. Comprehensive experiments across multi-center medical benchmarks demonstrate that

DKT-FDG consistently enhances robustness to domain shifts, validating the necessity of dynamic trajectory modeling in FL.

Three points define the assumptions and the scope of the method. First, DKT-FDG assumes that client representations evolve across rounds in a way that is informative for aggregation. We examine this assumption through its consequences rather than asserting it: replacing the flow-based trajectory model with a static variant while holding everything else fixed costs up to 4.55 accuracy points on Fed-ISIC (Fig. 3(b)), and Appendix F shows that the learned flow fields differ visibly across clients. Second, the benefit depends on the aggregation signal being discriminative across clients. When client distributions are nearly homogeneous, the signatures become close to one another and the affinity graph approaches uniform weighting, so DKT-FDG degenerates towards FedAvg and the gain diminishes; the method is therefore aimed at genuinely heterogeneous multi-center settings rather than at settings where client distributions are already close to IID. Third, the signature is a geometric descriptor of each round's representation state rather than a reconstruction of the intra-round optimization path, and the method is designed for a single global model that generalizes to unseen domains rather than for per-client specialization. Practically, the local flow head and its numerical integration add computation on the client, which warm-start and the lightweight residual design mitigate but do not remove, and extending the deterministic field to a stochastic one to capture epistemic uncertainty in trajectory evolution remains open.

We plan to explore integrating standard privacy-enhancing mechanisms, such as differential privacy, directly into the signature transmission pipeline to provide stronger formal guarantees. We also aim to extend DKT-FDG to multimodal federated medical settings, broadening its applicability to more complex clinical scenarios.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 62371409) and National High Level Hospital Clinical Research Funding (2025-PUMCH-D-002 and BJ-2025-138).

References

[1] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282, 2017.

[2] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14(1–2):1–210, 2021.

[3] Jing Yang, Yuxi Ma, Jin-Gang Yu, Feng Gao, Shuting Yang, Du Cai, Jiacheng Wang, and Liansheng Wang. Bridging knowledge discrepancy in retinal image analysis through federated multi-task learning. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 35–44. Springer, 2025.

[4] Hao Guan, Pew-Thian Yap, Andrea Bozoki, and Mingxia Liu. Federated learning for medical image analysis: A survey. *Pattern recognition*, 151:110424, 2024.

[5] Lily Kwak and Harrison Bai. The role of federated learning models in medical imaging. *Radiology: Artificial Intelligence*, 5(3):e230136, 2023.

[6] Yuxi Ma, Jiacheng Wang, Jing Yang, and Liansheng Wang. Model-heterogeneous semi-supervised federated learning for medical image segmentation. *IEEE Transactions on Medical Imaging*, 43(5):1804–1815, 2024.

[7] Zikun Zhou, Wen Huang, Xingyi Wang, Zhishuo Zhang, Zhun Zhang, Jian Peng, and Feihu Huang. Improving federated domain generalization through dynamical weights calculated from data influences on global model update. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 23045–23053, 2025.

[8] Trong-Binh Nguyen, Minh-Duong Nguyen, Jinsun Park, Quoc-Viet Pham, and Won Joo Hwang. Federated domain generalization with data-free on-server matching gradient. *arXiv preprint arXiv:2501.14653*, 2025.

[9] Zheng Wang, Zihui Wang, Xiaoliang Fan, and Cheng Wang. Federated learning with domain shift eraser. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 4978–4987, 2025.

[10] Yuhang Chen, Wenke Huang, and Mang Ye. Fair federated learning under domain skew with local consistency and domain diversity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12077–12086, 2024.

[11] Kejia Fan, Jianheng Tang, Zhirui Yang, Feijiang Han, Jiaxu Li, Run He, Yajiang Huang, Anfeng Liu, Houbing Herbert Song, Yunhuai Liu, et al. Apfl: Analytic personalized federated learning via dual-stream least squares. *arXiv preprint arXiv:2508.10732*, 2025.

[12] Jiayi Chen, Benteng Ma, Hengfei Cui, and Yong Xia. Think twice before selection: Federated evidential active learning for medical image analysis with domain shifts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11439–11449, 2024.

[13] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450, 2020.

[14] Jakub Konečnỳ, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

[15] Ruipeng Zhang, Qinwei Xu, Jiangchao Yao, Ya Zhang, Qi Tian, and Yanfeng Wang. Federated domain generalization with generalization adjustment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3954–3963, June 2023.

[16] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

[17] Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečnỳ, Sanjiv Kumar, and H Brendan McMahan. Adaptive federated optimization. *arXiv preprint arXiv:2003.00295*, 2020.

[18] Panayotis Mertikopoulos, Bruno Lecouat, Houssam Zenati, Chuan-Sheng Foo, Vijay Chandrasekhar, and Georgios Piliouras. Optimistic mirror descent in saddle-point problems: Going the extra (gradient) mile. *arXiv preprint arXiv:1807.02629*, 2018.

[19] Noel CF Codella, David Gutman, M Emre Celebi, Brian Helba, Michael A Marchetti, Stephen W Dusza, Aadi Kalloo, Konstantinos Liopyris, Nabin Mishra, Harald Kittler, et al. Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (isbi), hosted by the international skin imaging collaboration (isic). In *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*, pages 168–172, 2018.

[20] Hongyi Pan, Gorkem Durak, Elif Keles, Deniz Seyithanoglu, Zheyuan Zhang, Alpay Medetalibeyoglu, Halil Ertugrul Aktas, Andrea Mia Bejar, Ziliang Hong, Yavuz Taktak, et al. Cyst-x: Ai-powered pancreatic cancer risk prediction from multicenter mri in centralized and federated learning. *arXiv preprint arXiv:2507.22017*, 2025.

[21] Debesh Jha, Pia H Smedsrud, Michael A Riegler, Pål Halvorsen, Thomas de Lange, Dag Johansen, and Håvard D Johansen. Kvasir-seg: A segmented polyp dataset. In *International Conference on Multimedia Modeling*, pages 451–462, 2020.

[22] Juan Silva, Aymeric Histace, Olivier Romain, Xavier Dray, and Bertrand Granado. Toward embedded detection of polyps in wce images for early diagnosis of colorectal cancer. *International journal of computer assisted radiology and surgery*, 9(2):283–293, 2014.

[23] Nima Tajbakhsh, Suryakanth R Gurudu, and Jianming Liang. Automated polyp detection in colonoscopy videos using shape and context information. *IEEE transactions on medical imaging*, 35(2):630–644, 2015.

[24] Jorge Bernal, F Javier Sánchez, Gloria Fernández-Esparrach, Debora Gil, Cristina Rodríguez, and Fernando Vilariño. Wm-dova maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians. *Computerized medical imaging and graphics*, 43: 99–111, 2015.

[25] Geert Litjens, Robert Toth, Wendy Van De Ven, Caroline Hoeks, Sjoerd Kerkstra, Bram Van Ginneken, Graham Vincent, Gwenael Guillard, Neil Birbeck, Jindang Zhang, et al. Evaluation of prostate segmentation algorithms for mri: the promise12 challenge. *Medical image analysis*, 18 (2):359–373, 2014.

[26] B Nicholas Bloch, Anant Madabhushi, Henkjan Huisman, John Freymann, Justin Kirby, Michael Grauer, Andinet Enquobahrie, Carl Jaffe, Larry Clarke, and Keyvan Farahani. Nci-isbi 2013 challenge: Automated segmentation of prostate structures (isbi-mr-prostate-2013). *The Cancer Imaging Archive*, 2015.

[27] Guillaume Lemaître, Robert Martí, Jordi Freixenet, Joan C Vilanova, Paul M Walker, and Fabrice Meriaudeau. Computer-aided detection and diagnosis for prostate cancer based on mono and multi-parametric mri: a review. *Computers in biology and medicine*, 60:8–31, 2015.

[28] José Ignacio Orlando, Huazhu Fu, João Barbosa Breda, Karel Van Keer, Deepti R Bathula, Andrés Diaz-Pinto, Ruogu Fang, Pheng-Ann Heng, Jeyoung Kim, JoonHo Lee, et al. Refuge challenge: A unified framework for evaluating automated methods for glaucoma assessment from fundus photographs. *Medical image analysis*, 59:101570, 2020.

[29] Jayanthi Sivaswamy, SR Krishnadas, Gopal Datt Joshi, Madhulika Jain, and A Ujjwaft Syed Tabish. Drishti-gs: Retinal image dataset for optic nerve head (onh) segmentation. In *2014 IEEE 11th international symposium on biomedical imaging (ISBI)*, pages 53–56, 2014.

[30] Francisco José Fumero Batista, Tinguaro Diaz-Aleman, Jose Sigut, Silvia Alayon, Rafael Arnay, and Denisse Angel-Pereira. Rim-one dl: A unified retinal image database for assessing glaucoma using deep learning. *Image Analysis & Stereology*, 39(3):161–167, 2020.

[31] Quande Liu, Cheng Chen, Jing Qin, Qi Dou, and Pheng-Ann Heng. Feddg: Federated domain generalization on medical image segmentation via episodic learning in continuous frequency space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1013–1023, 2021.

[32] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[33] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[34] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[35] Thomas Hakon Gronwall. Note on the derivatives with respect to a parameter of the solutions of a system of differential equations. *Annals of Mathematics*, 20(4):292–296, 1919.

[36] Jean Ogier du Terrail, Samy-Safwan Ayed, Edwige Cyffers, Felix Grimberg, Chaoyang He, Regis Loeb, Paul Mangold, Tanguy Marchand, Othmane Marfoq, Erum Mushtaq, et al. Flamby: Datasets and benchmarks for cross-silo federated learning in realistic healthcare settings. *Advances in Neural Information Processing Systems*, 35:5315–5334, 2022.

Overview of Appendices

This supplementary material provides theoretical motivations, algorithmic details, additional experimental results, and comprehensive discussions to support the main paper. The appendices are organized as follows:

- **Appendix A: Conceptual Illustration.** We provide a conceptual illustration highlighting the necessity of modeling representation dynamics, contrasting static alignment with the proposed dynamic alignment paradigm.
- **Appendix B: Theoretical Motivation and Component Justification.** We provide a motivating analysis characterizing why static alignment is structurally fundamentally inadequate to handle heterogeneous representation dynamics, and formalizing the design rationale for the TAFH, DSA, and CDA components of DKT-FDG.
- **Appendix C: Algorithmic Overview.** We present the complete pseudo-code of DKT-FDG, detailing the closed-loop interaction between the client-side Dynamic Flow Signature Module (DFSM) and the server-side Knowledge Tracking Strategy (KTS).
- **Appendix D: Details of DFSM.** We provide the specific architectural implementation of the Task-Aware Flow Head (TAFH), including input normalization and residual blocks. Additionally, we present the rigorous mathematical formulations for the statistics used in the Dynamic Signature Analyzer (DSA), such as Representation Drift Magnitude and Trajectory Curvature.
- **Appendix E: Details of KTS.** We detail the mathematical implementation of the server-side strategy, including the construction of the sparse affinity graph in the Cross-Domain Analyzer (CDA) and the derivation of adaptive weights in the Trajectory-Aware Aggregation (TAA).
- **Appendix F: Visualization of Latent Dynamics.** We provide qualitative visualizations of the latent flow fields learned by TAFH across multiple datasets (Fed-ISIC, IPMN, Fed-Fundus, Fed-Polyp, and Fed-Prostate), and verify that domain heterogeneity manifests as distinct dynamic trajectories in the latent space.
- **Appendix G: Necessity of Fourier Positional Encoding.** We discuss and empirically validate the importance of using high-dimensional Fourier embeddings over scalar time inputs for accurately capturing fine-grained temporal dynamics in TAFH.
- **Appendix H: Privacy Considerations.** We discuss the privacy-conscious design of DKT-FDG, emphasizing its adherence to data minimization principles and compatibility with standard privacy-enhancing mechanisms.
- **Appendix I: Dataset Details.** We provide comprehensive descriptions of the five multi-center medical imaging benchmarks used in our study, including data sources, sample sizes, and the nature of domain shifts.
- **Appendix J: Experiments and Implementation.** We offer additional implementation details (hyperparameters, hardware), extended experimental results on domain generalizability (leave-one-client-out setting), and formal definitions of the evaluation metrics used for classification and segmentation tasks.

A Conceptual Illustration of Representation Dynamics

Fig. 4 conceptually illustrates why static alignment is insufficient in FDG. Even when client representations are momentarily aligned at the start of round r, heterogeneous local datasets induce divergent optimization velocities ($v_A \neq v_B$), causing renewed misalignment after local updates. This structural divergence cannot be resolved by any single-round alignment strategy. DKT-FDG addresses this by explicitly modeling and aligning the continuous representation trajectories via flow matching, suppressing directional mismatch across clients and improving temporal consistency throughout training.

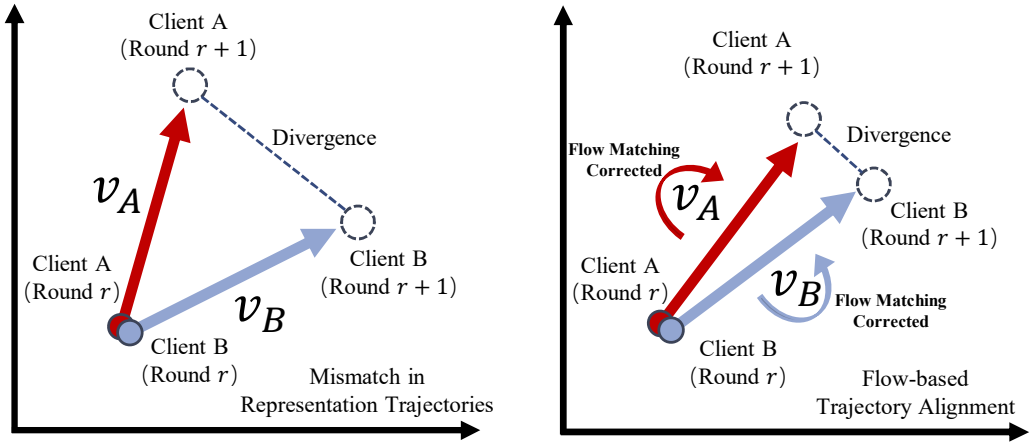

Figure 4: **Conceptual illustration highlighting the necessity of modeling representation dynamics. Static Alignment (left):** Even if client representations are aligned at the start of communication round r, heterogeneous local datasets induce mismatched representation trajectories (velocities v_A and v_B). This intrinsic divergence leads to renewed misalignment after local updates at round $r+1$. **Dynamic Alignment (right):** By explicitly modeling and aligning representation trajectories via flow matching, the proposed framework suppresses trajectory mismatch across clients, resulting in more coherent updates and improved temporal consistency throughout training.

B Theoretical Motivation and Component Justification

This appendix provides theoretical motivation for DKT-FDG rather than end-to-end performance guarantees. The analysis proceeds in three steps: (i) we relate domain-specific latent distributions to the marginal velocity fields induced by the prescribed flow-matching interpolation; (ii) we show that the flow-matching objective approximates these fields, motivating TAFH; and (iii) we discuss the conditions and limitations of finite-sample field separation, motivating the complementary DSA statistics and CDA's comparison of client signatures.

B.1 Domain-Specific Transport Geometry

We consider the marginal transport induced by the interpolation used in the actual flow-matching objective, rather than assuming an optimal-transport coupling. For client i, independently sample $z_0 \sim p_0 = \mathcal{N}(0, I)$ and $z_1 \sim p_1^{(i)}$ and set $z_t = (1-(1-\sigma_{\min})t)z_0 + tz_1$. Write $p_t^{(i)}$ for the marginal law of z_t. For $\sigma_{\min} > 0$, its endpoint is the noise-smoothed law $p_{t=1}^{(i)} = p_1^{(i)} * \mathcal{N}(0, \sigma_{\min}^2 I)$.

Definition 1 (Marginal Flow-Matching Velocity Field)**.** *The reference velocity field of client i is*

$$v_i^*(z,t) \triangleq \mathbb{E}[z_1 - (1-\sigma_{\min})z_0 \mid z_t = z,\, t]\,. \tag{13}$$

It is the unique (up to $p_t^{(i)}\,\mathrm{d}t$-almost-everywhere equivalence) minimizer of the population flow-matching regression loss [16] for this prescribed path. No Benamou–Brenier optimality is assumed.

Assumption 1 (Regularity)**.** *For each client i, the marginal velocity field v_i^* is K-Lipschitz in z and continuous in t, with sufficient regularity for uniqueness of the induced flow and continuity equation from the common initial distribution p_0.*

Proposition 1 (Non-Congruence of Domain Transport)**.** *Under Assumption 1, if clients i and j have different feature distributions ($p_1^{(i)} \neq p_1^{(j)}$), then:*

$$\left\|v_i^* - v_j^*\right\|_{L^2(q)} > 0, \tag{14}$$

where $q_t^{(i,j)} \triangleq \frac{1}{2}(p_t^{(i)} + p_t^{(j)})$ and $\|v\|_{L^2(q)}^2 \triangleq \int_0^1 \mathbb{E}_{z\sim q_t^{(i,j)}}[\|v(z,t)\|^2]\,\mathrm{d}t$.

Proof. Suppose instead that $\|v_i^* - v_j^*\|_{L^2(q)} = 0$. Because $q_t^{(i,j)}$ has an everywhere-positive density for $0 < t < 1$ (Gaussian smoothing), and the fields are continuous, the two fields must coincide

on the relevant space-time domain. Their flows then have identical marginal laws from the shared initial law p_0, by uniqueness under Assumption 1. In particular, $p_{t=1}^{(i)} = p_{t=1}^{(j)}$. Since convolution with a non-degenerate Gaussian is injective on probability measures, the endpoint identity implies $p_1^{(i)} = p_1^{(j)}$, a contradiction. □

Remark 1 (Intrinsic Nature of Transport Geometry)**.** *The transport velocity field v_i^* is an intrinsic property of the domain's feature distribution $p_1^{(i)}$ relative to the reference p_0. For a fixed feature distribution and prescribed interpolation, this field does not depend on the optimisation algorithm used to obtain that distribution; the encoder influences it through its learned features. Proposition 1 establishes that distinct feature distributions induce distinct marginal fields under our regularity conditions, but does not guarantee that a finite, compressed signature separates every client pair.*

Remark 2 (Connection to FDG)**.** *During FL training, each client's encoder $f(\,\cdot\,;\theta_i^{(r)})$ maps its domain data to a feature distribution $p_1^{(i,r)}$ that evolves across rounds. As local optimisation changes these feature distributions, the corresponding marginal velocity fields may change across communication rounds. TAFH approximates the field at each round; tracking the resulting signatures across rounds provides an indirect description of representation evolution, rather than a reconstruction of individual local optimisation steps.*

B.2 Flow Matching Recovers Transport Geometry

We now establish that minimising the flow-matching objective approximates the marginal velocity field of the prescribed interpolation, as defined in Definition 1.

Assumption 2 (Realizability)**.** *The TAFH network class $\{v_\theta : \theta \in \Theta\}$ can realise any K-Lipschitz velocity field on $\mathcal{Z} \times [0,1]$ to within approximation error ϵ_{approx} in $L^2(p_{\text{path}})$ norm, where p_{path} denotes the marginal distribution over transport paths.*

Proposition 2 (Velocity Field Recovery)**.** *Under Assumptions 1 and 2, let v_θ minimise the flow matching loss $\mathcal{L}_{\text{flow}}$ (Eq. 4 of the main paper). Then:*

$$\mathbb{E}_{t,z_t\sim p_t}\Big[\big\|v_\theta(z_t,t) - v_i^*(z_t,t)\big\|^2\Big] \;\le\; \mathcal{L}_{\text{flow}}^*, \tag{15}$$

where $\mathcal{L}_{\text{flow}}^*$ is the minimum value of $\mathcal{L}_{\text{flow}}$ over θ. Moreover, under Assumption 2:

$$\mathbb{E}\big[\|v_\theta - v_i^*\|^2\big] \;\le\; \epsilon_{\text{approx}}^2. \tag{16}$$

Proof. By the flow matching framework [16], the conditional vector field $u(z_t \mid z_0, z_1) = z_1 - (1 - \sigma_{\min})z_0$ is the regression target of $\mathcal{L}_{\text{flow}}$, and satisfies the unbiasedness condition:

$$\mathbb{E}_{z_0,z_1|z_t,t}\big[u(z_t \mid z_0,z_1)\big] \;=\; v_i^*(z_t,t). \tag{17}$$

Decompose the flow matching loss by adding and subtracting v_i^*:

$$\begin{aligned}\mathcal{L}_{\text{flow}} &= \mathbb{E}_{t,z_0,z_1}\big[\|v_\theta(z_t,t) - u(z_t \mid z_0,z_1)\|^2\big] \\ &= \mathbb{E}\big[\|v_\theta - v_i^*\|^2\big] + \mathbb{E}\big[\|u - v_i^*\|^2\big] + 2\,\mathbb{E}[\langle v_\theta - v_i^*,\, u - v_i^*\rangle]\,.\end{aligned} \tag{18}$$

The cross term vanishes *exactly*: conditioning on (t, z_t), since $v_\theta(z_t,t) - v_i^*(z_t,t)$ is deterministic given (t, z_t) and $\mathbb{E}[u - v_i^* \mid z_t, t] = 0$ by unbiasedness,

$$\mathbb{E}[\langle v_\theta - v_i^*,\, u - v_i^*\rangle \mid t, z_t] \;=\; \Big\langle v_\theta - v_i^*,\, \underbrace{\mathbb{E}[u - v_i^* \mid t, z_t]}_{=\,0}\Big\rangle \;=\; 0. \tag{19}$$

Denoting $V_{\text{cond}} \triangleq \mathbb{E}[\|u - v_i^*\|^2] \ge 0$ (the θ-independent conditional variance of the regression target), Eq. (18) simplifies to:

$$\mathcal{L}_{\text{flow}} \;=\; \mathbb{E}\big[\|v_\theta - v_i^*\|^2\big] + V_{\text{cond}}. \tag{20}$$

For a global minimiser v_θ, we have $\mathcal{L}_{\text{flow}} = \mathcal{L}_{\text{flow}}^*$, and therefore:

$$\mathbb{E}\big[\|v_\theta - v_i^*\|^2\big] \;=\; \mathcal{L}_{\text{flow}}^* - V_{\text{cond}} \;\le\; \mathcal{L}_{\text{flow}}^*, \tag{21}$$

which establishes (15). At the global optimum, by Assumption 2:

$$\mathcal{L}_{\text{flow}}^* \;=\; \min_\theta\, \mathbb{E}\big[\|v_\theta - v_i^*\|^2\big] + V_{\text{cond}} \;\le\; \epsilon_{\text{approx}}^2 + V_{\text{cond}}, \tag{22}$$

so $\mathbb{E}[\|v_\theta - v_i^*\|^2] = \mathcal{L}_{\text{flow}}^* - V_{\text{cond}} \le \epsilon_{\text{approx}}^2$. □

Corollary 1 (Trajectory Approximation)**.** *Let $\hat{\gamma}_i$ be the piecewise-linear interpolation of Euler integration of v_θ with step size $h = 1/T$, and let γ_i^* solve the ODE driven by v_i^* from the same initial state. In addition to Assumption 1, suppose the learned and reference fields are sufficiently regular for first-order Euler convergence, and assume the uniform approximation bound $\epsilon_\infty \triangleq \sup_{z,t} \|v_\theta(z,t) - v_i^*(z,t)\| < \infty$ over a region containing the relevant trajectories. Then:*

$$\sup_{t\in[0,1]} \left\|\hat{\gamma}_i(t) - \gamma_i^*(t)\right\| \;\le\; \frac{e^K - 1}{K}\,\epsilon_\infty + O(h), \tag{23}$$

with the usual continuous limit when $K = 0$. The uniform error ϵ_∞ is an additional assumption; it does not follow from the population L^2 guarantee in Proposition 2.

Proof. Let γ_θ denote the exact ODE trajectory of v_θ from the shared initial state. Lipschitz continuity of v_i^* and the uniform velocity error yield, by Grönwall's inequality [35],

$$\sup_t \|\gamma_\theta(t) - \gamma_i^*(t)\| \le \frac{e^K - 1}{K}\,\epsilon_\infty. \tag{24}$$

Under the stated smoothness conditions, the Euler scheme approximates γ_θ to $O(h)$ uniformly in time. The triangle inequality completes the proof. □

B.3 Discriminability of Flow Signatures

The DSA compresses the full trajectory $\hat{\gamma}_i$ into a signature Sig_i. This compression is necessarily lossy; we characterise the conditions under which the signature remains discriminative.

Definition 2 (DSA Separation)**.** *Two clients i, j are Δ-*separated *if:*

$$\left\|v_i^* - v_j^*\right\|_{\mathrm{DSA}}^2 \;\triangleq\; \frac{1}{P}\sum_{p=1}^{P} \mathbb{E}_{z\sim q_{t^{(p)}}^{(i,j)}}\left[\left\|v_i^*(z,\, t^{(p)}) - v_j^*(z,\, t^{(p)})\right\|^2\right] \;\ge\; \Delta^2, \tag{25}$$

where $t^{(p)} = (p/P)^2$ are the DSA quadratic sampling points, and $q_t^{(i,j)} = \frac{1}{2}(p_t^{(i)} + p_t^{(j)})$ is a common reference measure for the theoretical comparison.

Remark 3 (Theoretical vs. Practical Distribution)**.** *The pairwise reference measure $q_t^{(i,j)}$ in Definition 2 is used only to state a common-space separation condition; it is not computed by DKT-FDG. In practice, each client obtains its DSA statistics from S particles simulated under its own learned field. Finite training, Euler integration, Monte Carlo sampling, and temporal subsampling all introduce approximation error. Corollary 1 addresses only the ODE trajectory error under its stated additional assumptions.*

Remark 4 (Finite-Sample Discriminability)**.** *Proposition 1 establishes a difference between continuous marginal velocity fields under its assumptions. However, this does not imply a nonzero difference at every fixed finite collection of sampling times: two continuous fields may agree at all chosen $t^{(p)}$ while differing elsewhere. Thus, Δ-separation is an explicit condition on the sampled fields, not a universal consequence of domain heterogeneity. Moreover, distinct sampled fields need not yield distinct four-statistic DSA signatures.*

Remark 5 (Signature Coverage and the Role of CDA)**.** *The DSA statistics summarize complementary properties of within-client transport, including drift, temporal stability, curvature, and distributional discrepancy. Because this compression is lossy, no claim of complete pairwise identifiability is made. CDA compares the full signatures $\{\mathrm{Sig}_i\}_{i=1}^N$ to construct a domain-affinity graph $G^{(r)}$. The per-statistic ablation (Fig. 3(a)) empirically supports the value of these complementary statistics, without establishing a general separation guarantee.*

Remark 6 (Practical Gap)**.** *Proposition 2 concerns the population flow-matching minimizer, while Corollary 1 additionally assumes a uniform velocity approximation bound and sufficient regularity for Euler convergence. DKT-FDG instead uses finite training, finite integration steps, and finite samples; these approximations need not improve monotonically with training time. The sensitivity analysis in Fig. 6(d,g) shows that $T = 50$ and five fine-tuning epochs provide favorable empirical performance, but does not establish a numerical bound on signature error.*

Algorithm 1 Dynamic Knowledge Tracked Federated Domain Generalization (DKT-FDG)

Input: N clients with datasets $\mathcal{D}_i$, communication rounds R, flow integration steps T, **sampling points** P, history window size L.
Initialize: Global model $\theta_{\mathcal{G}}^{(1)}$, initial graph $G^{(0)} \leftarrow I$.
Initialize: History buffers $\mathcal{H}_i \leftarrow \emptyset$ for all clients.
for round $r = 1$ **to** R **do**
  — Client Side: Dynamic Flow Signature Module (DFSM) —
  for client $i = 1$ **to** N in parallel **do**
    Receive global model $\theta_{\mathcal{G}}^{(r)}$.
    $\theta_i^{(r)}, \text{Sig}_i^{(r)} \leftarrow \text{CLIENTUPDATE}(\theta_{\mathcal{G}}^{(r)}, \mathcal{D}_i)$.
    Calculate local training metric $m_i^{(r)}$ on $\mathcal{D}_i$.
    Upload $\theta_i^{(r)}$, $\text{Sig}_i^{(r)}$, and $m_i^{(r)}$ to server.
  end for
  — Server Side: Knowledge Tracking Strategy (KTS) —
  1. Cross-Domain Analyzer (CDA):
  Construct affinity matrix A from signatures $\{\text{Sig}_i^{(r)}\}_{i=1}^N$.
  Compute sparse domain graph $G^{(r)}$.
  2. Trajectory-Aware Aggregation (TAA):
  Compute Global System Instability: $\mathcal{J}^{(r)} = \|G^{(r)} - G^{(r-1)}\|_F$.
  Update history windows $\mathcal{H}_i \leftarrow \mathcal{H}_i \cup \{(r, m_i^{(r)})\}$ (retain last L entries).
  for client $i = 1$ **to** N **do**
    Estimate trend $\beta_i^{(r)}$ via OLS on $\mathcal{H}_i$ (set $\beta_i^{(r)} \leftarrow 0$ if $|\mathcal{H}_i| < 2$).
  end for
  Determine temperature $\tau_r = \gamma(\mathcal{J}^{(r)})$.
  Compute adaptive weights: $w_i^{(r)} = \frac{\exp(\beta_i^{(r)}/\tau_r)}{\sum_{j=1}^N \exp(\beta_j^{(r)}/\tau_r)}$.
  Update global model: $\theta_{\mathcal{G}}^{(r+1)} \leftarrow \sum_{i=1}^N w_i^{(r)} \theta_i^{(r)}$.
end for

Function: $\text{CLIENTUPDATE}(\theta, \mathcal{D})$
Stage 1: Task Learning
Update parameters θ by minimizing $\mathcal{L}_{\text{task}}$ on $\mathcal{D}$.
Stage 2: Flow Matching (Frozen Encoder)
Freeze feature encoder in θ.
Update TAFH v_θ by minimizing $\mathcal{L}_{\text{flow}}$.
Stage 3: Signature Generation (DSA)
Sample particles $z^{(0)} \sim \mathcal{N}(0, I)$.
Simulate trajectories $\{z^{(n)}\}_{n=0}^T$ via Euler integration of v_θ.
Extract P **snapshots** $\{z^{(n_p)}\}_{p=1}^P$ **using quadratic schedule** $t^{(p)} = (p/P)^2$**.**
Compute statistics over snapshots: $E_{\text{drift}}, S_{\text{stability}}, D_{\text{curvature}}, \text{MMD}$.
$\text{Sig} \leftarrow [E_{\text{drift}}, S_{\text{stability}}, D_{\text{curvature}}, \text{MMD}]$.
return Updated θ, Sig
End Function

C Algorithmic Overview of DKT-FDG

To clearly illustrate the complete training and aggregation process of DKT-FDG, we provide the pseudo-code in Algorithm 1. This procedure summarizes the interaction between the local DFSM and the server-side KTS. The DFSM models the temporal evolution of local latent features, while the KTS adaptively tracks and aggregates inter-client domain dynamics across communication rounds. Together, they establish a closed feedback loop that enables stable and temporally consistent FL.

D Implementation Details of DFSM

In this section, we provide the specific architectural details of TAFH and the rigorous mathematical formulations for the statistics used in DSA.

D.1 TAFH Architecture Details

Input Normalization. Prior to feeding latent features into the TAFH, we standardize the input $z_1 \in \mathbb{R}^{d_f}$ (where d_f denotes the feature dimension) using local running statistics to ensure training stability across heterogeneous clients. Let μ_i and s_i be the mean and standard deviation of the latent features for client i. The normalized input is given by $\tilde{z}_1 = (z_1 - \mu_i)/(s_i + \epsilon)$, where ϵ (e.g., 10^{-5}) is a small constant for numerical stability.

Temporal Injection. As defined in Eq. 1 of the main paper, we use a Fourier embedding $\phi(t) \in \mathbb{R}^{d_t}$. To integrate this time information into the network, we project it into the hidden dimension d_h using a 2-layer MLP:

$$e_t = W_2 \,\mathrm{SiLU}(W_1 \phi(t) + b_1) + b_2, \tag{26}$$

where $W_1 \in \mathbb{R}^{d_h \times d_t}$ and $W_2 \in \mathbb{R}^{d_h \times d_h}$ are learnable projection matrices, and $b_1 \in \mathbb{R}^{d_h}, b_2 \in \mathbb{R}^{d_h}$ are the corresponding bias vectors. The temporal embedding e_t is then concatenated with the normalized interpolated state $\tilde{z}_t$, obtained by applying the above normalization to the transport path of Eq. (2), and passed through a linear projection layer to yield the initial hidden state:

$$h_0 = W_{\text{proj}}[\tilde{z}_t;\, e_t] + b_{\text{proj}}, \tag{27}$$

where $W_{\text{proj}} \in \mathbb{R}^{d_h \times (d_f + d_h)}$ and $b_{\text{proj}} \in \mathbb{R}^{d_h}$.

Residual Block Structure. The TAFH is composed of n_{res} stacked residual blocks. For the l-th residual block, the update rule matches the formulation in the main text:

$$h_{l+1} = h_l + W_{l,2} \,\mathrm{SiLU}(W_{l,1} h_l + b_{l,1}) + b_{l,2}, \tag{28}$$

where $W_{l,1}, W_{l,2}$ are weight matrices, and $b_{l,1}, b_{l,2}$ are the learnable biases for the bottleneck layers. This specific residual design mitigates the vanishing gradient problem during the optimization of the flow field. Finally, a linear output layer maps the hidden state back to the data dimension d_f to predict the velocity v_θ.

D.2 DSA Statistic Formulations

The DSA summarizes the trajectory dynamics into a flow signature using four key statistics. Consistent with the main text, these statistics are computed based on the sequence of velocity vectors $\{v^{(n_p)}\}_{p=1}^{P}$ recorded at the quadratic sampling steps $t^{(p)} = (p/P)^2$.

Representation Drift Magnitude. This metric quantifies the total energy of the transformation along the sampled trajectory points. A higher drift magnitude implies a more significant shift from the Gaussian prior. It is computed as the mean squared velocity norm over the P sampled points:

$$E_{\text{drift}} = \frac{1}{P} \sum_{p=1}^{P} \|v^{(n_p)}\|_2^2. \tag{29}$$

Temporal Stability. We estimate the stability of the flow by analyzing the variance of the velocity changes between sampled intervals. Let the discrete change in velocity be $\Delta v^{(n_p)} = v^{(n_p)} - v^{(n_{p-1})}$. The stability score is defined as the inverse of the volatility of these changes:

$$S_{\text{stability}} = \frac{1}{1 + \mathrm{Std}\left(\{\|\Delta v^{(n_p)}\|_2\}_{p=2}^{P}\right)}. \tag{30}$$

A high $S_{\text{stability}}$ indicates a smooth optimization path where velocity changes are consistent, whereas a low value suggests erratic updates.

Trajectory Curvature. To measure the geometric consistency of the optimization path, we compute the deviation from a straight line using the cosine similarity between consecutive velocity vectors. We define the curvature metric as:

$$D_{\text{curvature}} = 1 - \frac{1}{P-1} \sum_{p=2}^{P} \left| \frac{\langle v^{(n_p)}, v^{(n_{p-1})} \rangle}{\|v^{(n_p)}\|_2 \|v^{(n_{p-1})}\|_2} \right|. \tag{31}$$

A lower value implies that consecutive velocity directions are collinear, indicating a linear and stable domain shift, whereas a higher value reflects orthogonal transitions between consecutive velocities, indicative of oscillatory instability or noise.

Flow Signature Construction. Finally, as mentioned in the main text, we calculate the Maximum Mean Discrepancy (MMD) between the simulated flow endpoints $\{z_s^{(n_P)}\}_{s=1}^{S}$ (at $t \approx 1$) and the real empirical features to ensure distributional fidelity. The final flow signature Sig_i for client i is:

$$\text{Sig}_i = [E_{\text{drift}}, S_{\text{stability}}, D_{\text{curvature}}, \text{MMD}]. \tag{32}$$

This compact vector is transmitted to the server for KTS.

E Implementation Details of Knowledge Tracking Strategy (KTS)

This section provides the rigorous mathematical formulations and implementation details for the CDA and TAA.

E.1 Cross-Domain Analyzer (CDA)

To remove scale bias from the raw flow signatures prior to distance computation, we perform Z-score normalization. For a signature vector $\text{Sig}_i \in \mathbb{R}^{d_s}$, where d_s denotes the signature dimension, the standardized version $\hat{s}_i$ is computed as:

$$\hat{s}_{i,k} = \frac{\text{Sig}_{i,k} - \mu_k}{s_k + \epsilon}, \quad \forall k \in \{1, \dots, d_s\}, \tag{33}$$

where μ_k and s_k are the mean and standard deviation of the k-th dimension across all N clients.

In the main paper, we introduced a sparsity constraint to filter out noisy correlations. Specifically, we implement this via a K-Nearest Neighbor (K-NN) approach. For each client i, we define the set of neighbors $\mathcal{K}_i$ corresponding to the top-K largest values in the dense affinity matrix A. The sparse matrix $\mathbb{A}$ is constructed as:

$$\mathbb{A}_{ij} = \begin{cases} A_{ij} & \text{if } j \in \mathcal{K}_i \text{ or } i \in \mathcal{K}_j, \\ 0 & \text{otherwise.} \end{cases} \tag{34}$$

This ensures that the graph topology relies only on statistically significant correlations, preserving the local manifold structure of the domain space. The subsequent symmetrization $\mathbb{A}+\mathbb{A}^\top$ is intentional: it ensures graph connectivity by including edges where either endpoint considers the other a top-$\lfloor N/2 \rfloor$ neighbor, preventing isolated clients from being excluded from the aggregation topology.

The resulting normalized graph $G^{(r)}$ serves as the basis for computing adaptive aggregation weights in TAA, as described in the following section.

E.2 Trajectory-Aware Aggregation (TAA)

Least-Squares Trend Estimation. We estimate the performance trend $\beta_i^{(r)}$ using an Ordinary Least Squares (OLS) linear fit over the historical window $\mathcal{H} = \{r_1, r_2, \dots, r_L\}$. Let $m_i^{(r)}$ be the metric value at round r. The slope $\beta_i^{(r)}$ has a closed-form solution:

$$\beta_i^{(r)} = \frac{\sum_{j=1}^{L}(r_j - \bar{r})(m_i^{(r_j)} - \bar{m}_i)}{\sum_{j=1}^{L}(r_j - \bar{r})^2}, \tag{35}$$

where $\bar{r}$ and $\bar{m}_i$ are the means of the round indices and performance metrics in the window, respectively.

Entropy Regulation via Temperature. The temperature $\tau_r = \gamma(\mathcal{J}^{(r)})$ controls the entropy of the aggregation weights. The Shannon entropy of the weight distribution $w^{(r)}$ is given by $H(w^{(r)}) = -\sum_i w_i^{(r)} \log w_i^{(r)}$.

- **High Instability ($\mathcal{J}^{(r)} \uparrow$):** τ_r increases (approaching 1 if unscaled). The weights $w_i^{(r)}$ become more uniform, maximizing entropy $H(w^{(r)})$. This corresponds to a maximum

uncertainty state where the server trusts all clients equally to avoid overfitting to a potentially shifting topology.

- **High Stability ($\mathcal{J}^{(r)} \downarrow$):** τ_r decreases. The softmax function becomes sharper, minimizing entropy. This focuses the aggregation mass solely on clients with the highest improvement trends $\beta_i^{(r)}$.

F Visualization of Latent Representations Evolving

To more faithfully interpret the temporal dynamics modeled by the TAFH on Fed-ISIC, we visualize the learned flow fields by explicitly simulating the latent trajectories under the deterministic flow. To ensure strict cross-client comparability, we sample *one shared set* of $S = 128$ (reduced from $S = 300$ used in training for visualization efficiency) latent particles $z^{(0)} \sim \mathcal{N}(0, I)$ and use the *same initialization* for all clients. For each client, we load the corresponding learned flow head and perform Euler integration for $T = 50$ timesteps.

During simulation, we record latent states at a fixed set of time points $t^{(p)} = (p/P)^2$, $p = 1, \ldots, P$ using a quadratically–spaced schedule. This design captures early dynamics more densely, as latent trajectories typically undergo their most rapid changes in velocity and directional onset immediately after the Gaussian initialization, even though their global geometry remains visually unstructured at that stage. As t increases, temporal variations become smoother, permitting progressively coarser sampling.

Since raw latent velocities may include outliers caused by abrupt domain-specific drifts, we apply an interquartile range (IQR)-based velocity normalization strategy to suppress extreme deviations while preserving flow directions. Furthermore, to enable consistent 2D visualization across heterogeneous client manifolds, we jointly fit a PCA projection on the concatenated latent states from all clients and all time points, and project every latent trajectory onto this *shared* PCA basis. To avoid misleading scale variations, we further determine a unified plotting range for both axes based on the 95th percentile of the absolute values of all projected coordinates across both axes, ensuring that the same spatial scale is used across all clients and time intervals.

The final visualization (Fig. 7) plots, for each client and each time window $t \in [0, 1]$, the projected latent particles and their normalized instantaneous flow vectors $v_\theta(z_t, t)$. Arrows illustrate the local direction and magnitude of the temporal evolution in latent space.

At very early timesteps ($t < 0.1$), latent particles exhibit an outward expansion that appears nearly isotropic in geometry, reflecting the diffusion-like behavior expected from a shared Gaussian initialization. However, subtle directional drifts already begin to emerge in this phase, precisely the type of high-sensitivity dynamics that the dense early sampling is designed to capture. At intermediate stages ($t \approx 0.23$–0.52), these small deviations evolve into pronounced, domain-dependent flow patterns with coherent drifts characteristic of each client's imaging distribution. In later phases ($t > 0.8$), trajectories elongate along stable directions, indicating convergence toward client-specific latent submanifolds.

These visual patterns indicate that TAFH captures both the kinetic dynamics and the geometric structure of domain divergence. Clients exhibiting larger inter-domain shifts show more pronounced and anisotropic flow fields. The resulting temporal trajectories summarize the latent-space evolution and are subsequently distilled by the DSA into compact flow signatures for downstream aggregation. Additional flow visualizations for other datasets are shown in Fig. 8–12.

This qualitative evidence supports our hypothesis that domain heterogeneity manifests as *dynamic latent trajectories* rather than static offsets, and that explicitly modeling these dynamics enables DKT-FDG to estimate representation drift more reliably and assign temporally consistent aggregation weights.

G Necessity of Fourier positional encoding

A core challenge in DKT-FDG is the accurate characterization of fine-grained latent dynamic trajectories induced by asynchronous local updates. To estimate the instantaneous flow field $v_\theta(z_t, t)$ in the TAFH module, thereby capturing fine-grained temporal behaviors such as velocity changes,

Table 3: Details of multi-center datasets used in our study

Dataset	Source	Train	Test
Fed-ISIC	Client 0: BCN [36]	9,930	2,483
	Client 1: HAM_vidir_molemax [36]	3,163	791
	Client 2: HAM_vidir_modern [36]	2,691	672
	Client 3: HAM_rosendahl [36]	1,807	452
Fed-Polyp	Client 0: Kvasir [21]	800	200
	Client 1: ETIS [22]	157	39
	Client 2: ColonDB [23]	304	75
	Client 3: ClinicDB [24]	490	122
Fed-Prostate	Client 0: BIDMC [25]	225	36
	Client 1: BMC [26]	306	78
	Client 2: HK [25]	133	24
	Client 3: I2CVB [27]	394	74
	Client 4: RUNMC [26]	328	93
	Client 5: UCL [25]	147	28
Fed-Fundus	Client 0: Drishti-GS [29]	81	20
	Client 1: RIM-ONE-r3 [30]	128	31
	Client 2: REFUGE [28]	320	80
	Client 3: REFUGE [28]	320	80
IPMN (T1W Modality)	Client 0: NYU [20]	120	30
	Client 1: MCF [20]	107	27
	Client 2: NU [20]	148	38
	Client 3: AHN [20]	12	4
	Client 4: MCA [20]	19	5
	Client 5: IU [20]	51	13
	Client 6: EMC [20]	62	16
IPMN (T2W Modality)	Client 0: NYU [20]	120	31
	Client 1: MCF [20]	104	26
	Client 2: NU [20]	149	38
	Client 3: AHN [20]	14	4
	Client 4: MCA [20]	18	5
	Client 5: IU [20]	50	13
	Client 6: EMC [20]	66	17

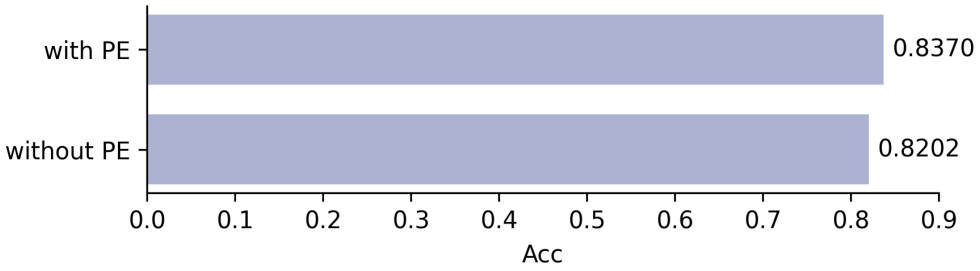

Figure 5: Ablation study on Fourier positional encoding (PE). *without PE* refers to using scalar time inputs, while *with PE* incorporates the Fourier positional encoding.

curvature, and distributional drift, the model must effectively perceive the continuous temporal variable $t \in [0, 1]$.

A straightforward approach is to directly input the scalar t into TAFH. However, this representation provides only a low-dimensional and weakly expressive temporal signal, making it difficult for the model to learn multi-scale or non-linear temporal dependencies. Empirically, this limitation leads to degraded performance, yielding an accuracy of 0.8202 (Fig. 5).

To address this insufficiency, we incorporate a Fourier positional encoding (Eq. 1), which maps the scalar t into a high-dimensional, information-rich embedding $\phi(t) \in \mathbb{R}^{d_t}$. This encoding expands t through a set of sinusoidal functions whose frequencies increase exponentially, endowing the model with multi-scale temporal sensitivity. Low-frequency components capture smooth, long-range temporal evolution, while high-frequency components encode localized and rapidly varying temporal patterns, both of which are essential for estimating fine-grained flow fields in latent space.

This multi-frequency construction transforms a single scalar into a structured temporal representation, enabling TAFH to learn continuous-time latent dynamics with markedly higher fidelity. Experimental results show that incorporating Fourier positional encoding improves the overall performance to 0.8370, corresponding to an absolute gain of 1.68% over the scalar-time baseline. These results confirm that a rich temporal representation is crucial for capturing asynchronous dynamic evolution in federated optimization.

H Privacy Considerations

DKT-FDG is designed in accordance with the principle of data minimization in privacy-conscious FL. First, the DFSM extracts latent flow representations derived solely from local updates, without transmitting raw images, labels, or intermediate feature maps. These signatures encapsulate the dynamic evolution of local representations as a compressed statistical abstraction, reducing the information content exposed during training compared to transmitting raw features or intermediate representations.

Second, all inter-client communication is restricted to model parameters and a compact flow signature vector per client per round, rather than any patient-level data. Such a design ensures that the information exchanged during training remains strictly abstracted from raw medical content, consistent with data minimization principles.

In addition, DKT-FDG follows a decentralized optimization paradigm, which inherently avoids centralized data storage. While the current framework operates under standard FL assumptions, its modular design is compatible with additional privacy-enhancing mechanisms, such as homomorphic encryption, secure aggregation, or differential privacy, should stronger formal guarantees be required in practice. These techniques can be incorporated into the communication pipeline without altering the core dynamics modeling process.

Finally, the KTS adaptively regulates aggregation weights using temporal statistics, enabling stable global optimization while avoiding the need for raw or fine-grained local information.

We acknowledge two current limitations of the privacy analysis. First, while flow signatures do not contain raw data or labels, they encode statistical properties of local latent distributions and may, under adversarial inference attacks, leak distributional information, particularly when individual client datasets are small. Second, the current framework does not provide formal differential privacy guarantees on signature transmission. Quantifying the information leakage of transmitted signatures under rigorous privacy accounting remains a direction for future work. These limitations are inherent to the broader class of feature-statistic-based FL methods, and DKT-FDG's modular design allows differential privacy or secure aggregation mechanisms to be incorporated into the communication pipeline without altering the core dynamics modeling process.

I Dataset

- **Fed-ISIC**: A skin lesion classification dataset. This dataset is composed of four different public datasets, each serving as an independent client. The four clients contain 12,413, 3,954, 3,363, and 2,259 dermoscopic images, respectively. Domain shifts mainly arise from the fact that the datasets from different sources were collected at different clinical centers using different dermoscopic devices and lighting conditions, resulting in significant differences in color, resolution, and lesion appearance of the images [19].
- **IPMN**: A dataset for classifying the risk of intraductal papillary mucinous neoplasm (IPMN) of the pancreas. The dataset includes 764 patients collected from seven international medical centers, comprising 652 T1-weighted (T1W) and 655 T2-weighted (T2W) MRI scans. The participating centers include New York University (NYU) Langone Health, Mayo Clinic Florida (MCF), Northwestern University (NU), Allegheny Health Network (AHN), Mayo Clinic Arizona (MCA), Istanbul University Faculty of Medicine (IU), and Erasmus Medical Center (EMC) [20].
- **Fed-Polyp**: A colorectal polyp segmentation dataset. This dataset integrates four widely used public polyp segmentation datasets, each serving as a client. The four clients contain 1,000, 196, 379, and 612 samples, respectively. These datasets were collected from different medical institutions using heterogeneous endoscopic devices and illumination settings, which introduce significant domain shifts in polyp morphology, size, and texture [21–24].
- **Fed-Prostate**: A prostate magnetic resonance imaging (MRI) segmentation dataset. This dataset combines six data sources from three well-known public challenges: NCI-ISBI 2013 [26], I2CVB [27], and PROMISE12 [25]. Each data source is regarded as an independent client, containing 261, 384, 158, 468, 421, and 175 2D axial slices, respectively. The multi-center MRI data capture substantial inter-institutional variations in scanning protocols, including magnetic field strengths, acquisition sequences, and imaging quality, thereby creating realistic domain shifts for FL.
- **Fed-Fundus**: A retinal fundus image segmentation dataset for segmenting the optic disc and optic cup in the retina. This dataset integrates three public datasets (REFUGE [28], Drishti-GS [29], and RIM-ONE-R3 [30]), forming four clients in total, where REFUGE is partitioned into two independent clients. The four clients contain 101, 159, 400, and 400 images, respectively. Domain shifts are mainly caused by different fundus cameras, imaging resolutions, fields of view, illumination conditions, and patient populations of varying ethnic backgrounds.

J Experiments

J.1 Additional Implementation Details

Our experiments were implemented using PyTorch 1.13.1 on NVIDIA A100-SXM4 GPUs. Following prior works [12, 31], we optimized the model parameters using the Adam optimizer with an initial learning rate of 5×10^{-4} and a weight decay of 5×10^{-4}. To facilitate stable convergence, we applied a multi-step learning rate annealing strategy, decaying the learning rate by a factor of 0.1 at the 50th round. Regarding data augmentation and input dimensions, distinct strategies were adopted: Fed-ISIC images were processed at a resolution of 224×224 with random rotation and flipping; IPMN volumes were resized to $96 \times 96 \times 96$ with random axis flipping and rotation; and for segmentation tasks (Fed-Fundus, Fed-Prostate, and Fed-Polyp), we utilized random cropping to generate 320×320

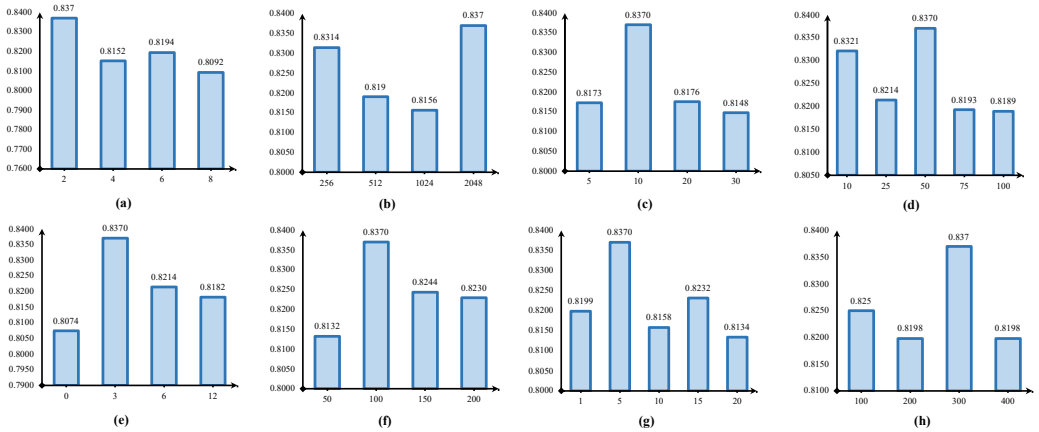

Figure 6: Sensitivity analysis of DKT-FDG on Fed-ISIC, evaluated by Acc. **(a)** Number of TAFH residual blocks n_{res}; **(b)** Hidden dimension d_h in TAFH; **(c)** Coarse-grained time points P; **(d)** Number of discrete timesteps T; **(e)** Historical window length L; **(f)** Pretraining epochs of TAFH; **(g)** Fine-tuning epochs of TAFH; **(h)** Number of latent particles S.

inputs, with Fed-Polyp specifically employing a foreground-aware cropping mechanism to ensure valid lesion inclusion.

Unless otherwise specified, all N clients participate in training. After the final round r, the converged global model $\theta_{\mathcal{G}}^{(r)}$ is distributed to each client and evaluated on its own private test subset $\mathcal{D}_i^{\text{test}}$, which remains unseen during local optimization. For the IPMN dataset, the training is conducted for $r = 50$, while for all other datasets we use $r = 200$. The reported results correspond to the *average performance of the global model across all clients*, reflecting how well the global representation generalizes under heterogeneous but jointly trained domains. Paired *t*-tests ($p < 0.05$) are conducted between DKT-FDG and the baseline using per-client results to assess statistical significance.

J.2 Sensitivity Analysis

The number of residual blocks n_{res} (Fig. 6(a)) peaks at $n_{\text{res}} = 2$; further increasing depth leads to gradual decline, suggesting that excessive stacking over-smooths latent dynamics. The hidden dimension d_h (Fig. 6(b)) shows a non-monotonic trend: performance drops from 256 to 1024 but recovers at 2048, indicating that sufficient latent capacity is needed to capture rich trajectory information. The dip at $d_h = 1024$ may reflect an intermediate capacity regime where the network is large enough to overfit local noise but insufficient to capture the full trajectory complexity, a phenomenon resolved at $d_h = 2048$. The number of coarse-grained snapshots P (Fig. 6(c)) shows that too few snapshots underrepresent the trajectory distribution while too many add redundancy; $P = 10$ achieves the best trade-off. The number of discrete integration steps T (Fig. 6(d)) confirms that small T increases integration error while large T incurs extra computation with diminishing returns; $T = 50$ works best in practice. The sliding window length L (Fig. 6(e)) shows that $L = 0$ produces noisy slope estimates while excessively large L over-smooths domain shifts; $L = 3$ achieves a favorable balance between responsiveness and stability. Pretraining epochs (Fig. 6(f)) confirm that sufficiently learned initial flows are essential for stable DSA simulation, as weak initialization causes numerical instability in particle trajectories, while excessively long pretraining provides only marginal gains at significant computational cost; 100 epochs achieves the best trade-off. Short fine-tuning (Fig. 6(g)) continues to improve performance without overfitting, with 5 epochs being optimal; beyond this, performance plateaus or slightly degrades. Finally, increasing the number of latent particles S (Fig. 6(h)) reduces variance in the simulated flow distribution and stabilizes signature statistics, with benefits tapering off beyond $S = 300$.

J.3 Matched-Budget Controls

To verify that the performance advantage of DKT-FDG does not stem from unequal training budgets, we conduct matched-budget controls on Fed-ISIC. Specifically, we augment each baseline with

additional local optimization steps to match the total per-round computation of DKT-FDG (which includes the frozen-encoder TAFH fine-tuning stage). Note that this additional computation provides *no extra task optimization* for DKT-FDG, as the encoder is frozen during flow head updates.

As shown in Table 4, increasing local optimization does not improve baselines and can even degrade performance under severe non-IID settings, consistent with the known instability of static alignment under heterogeneous local optimization: more local steps amplify representation divergence without providing the server any additional signal about inter-client trajectory differences. This confirms that the performance gap reflects a fundamental mismatch between static aggregation and heterogeneous optimization dynamics, rather than any tuning or budget advantage.

Table 4: Matched-budget controls on Fed-ISIC (Acc/F1). "+matched" denotes augmenting each baseline with additional local steps to match DKT-FDG's per-round computation.

Method	Acc	F1
FedAvg (1 local epoch)	0.7871	0.7783
FedAvg (+matched)	0.7668	0.7530
FedDI (1 local epoch)	0.8147	0.8061
FedDI (+matched)	0.7736	0.7618
Ours	**0.8370**	**0.8337**

J.4 Generalizability to Different Domains

On training clients, DKT-FDG maintains competitive accuracy with most baselines despite optimising primarily for global generalisability rather than in-distribution performance; on the unseen client, it achieves substantial improvements, reaching accuracies of 0.7500 and 0.7302 on IPMN-T1W and IPMN-T2W, respectively. This balance between seen-domain stability and unseen-domain generalisation indicates that the trajectory-aware aggregation in DKT-FDG captures transferable latent dynamics without sacrificing performance on observed domains, a property that distinguishes it from personalisation-oriented baselines whose improvements on training clients do not transfer to unseen sites.

J.5 Evaluation Metrics

For classification tasks, we report the classification accuracy (Acc) and the weighted average F1 score (F1). Accuracy is computed as

$$\mathrm{Acc} = \frac{1}{M}\sum_{i=1}^{M}\mathbb{I}(\hat{y}_i = y_i), \tag{36}$$

where M denotes the number of test samples, $\hat{y}_i$ is the predicted label, and y_i is the ground-truth label. The weighted F1 score is defined as

$$\mathrm{F1} = \sum_{c=1}^{C}\omega_c \cdot \mathrm{F1}_c, \quad \omega_c = \frac{N_c}{\sum_j N_j}, \tag{37}$$

where C is the number of classes and N_c is the number of samples belonging to class c.

For segmentation tasks, we report the Dice similarity coefficient (Dice) and the 95% Hausdorff Distance (Hd95). The Dice score for class c is computed using one-hot encodings of the prediction $\hat{y}$ and ground truth y:

$$\mathrm{Dice}_c = \frac{2\sum_p \hat{y}_p^{(c)} y_p^{(c)} + \epsilon}{\sum_p \hat{y}_p^{(c)} + \sum_p y_p^{(c)} + \epsilon}, \tag{38}$$

where p indexes spatial locations and ϵ is a small smoothing constant (e.g., 10^{-5}); the final Dice score is obtained by averaging over classes and samples.

For Hd95, we compute for each foreground class the 95th percentile of the bidirectional surface distance between the predicted and ground-truth binary masks:

$$\mathrm{Hd95}_c = \max\left\{P_{95}(d(\partial\hat{Y}^{(c)}, \partial Y^{(c)})), P_{95}(d(\partial Y^{(c)}, \partial\hat{Y}^{(c)}))\right\}, \tag{39}$$

Table 5: Accuracy comparison on the IPMN-T1W and IPMN-T2W datasets under the leave-one-client-out setting. *Training Clients* refers to the clients participating in training, while *Unseen Client* refers to the unseen client.

Method	IPMN-T1W		IPMN-T2W	
	Training Clients	Unseen Client	Training Clients	Unseen Client
Centralized	0.6623	0.5263	0.5237	0.3889
FedAvg [1]	0.4656	0.5263	0.4798	0.3889
FEAL [12]	0.4796	0.4211	0.5342	0.3333
FedHEAL [10]	0.4569	0.5789	0.4399	0.5000
FedDG_GA [15]	0.5318	0.4737	0.5002	0.5556
APFL [11]	0.5404	0.4211	0.5254	0.2778
FDSE [9]	0.5594	0.4211	0.4887	0.2222
FedOMG [8]	0.1902	0.5789	0.2265	0.7222
FedDI [7]	0.4999	0.4737	0.5174	0.5000
Ours	**0.5594**	**0.7500**	**0.5570**	**0.7302**

where ∂Y denotes the object boundary, $d(\cdot, \cdot)$ is the Euclidean surface distance, and P_{95} extracts the 95th percentile. The overall Hd95 is computed by averaging over all non-empty foreground classes and batch samples.

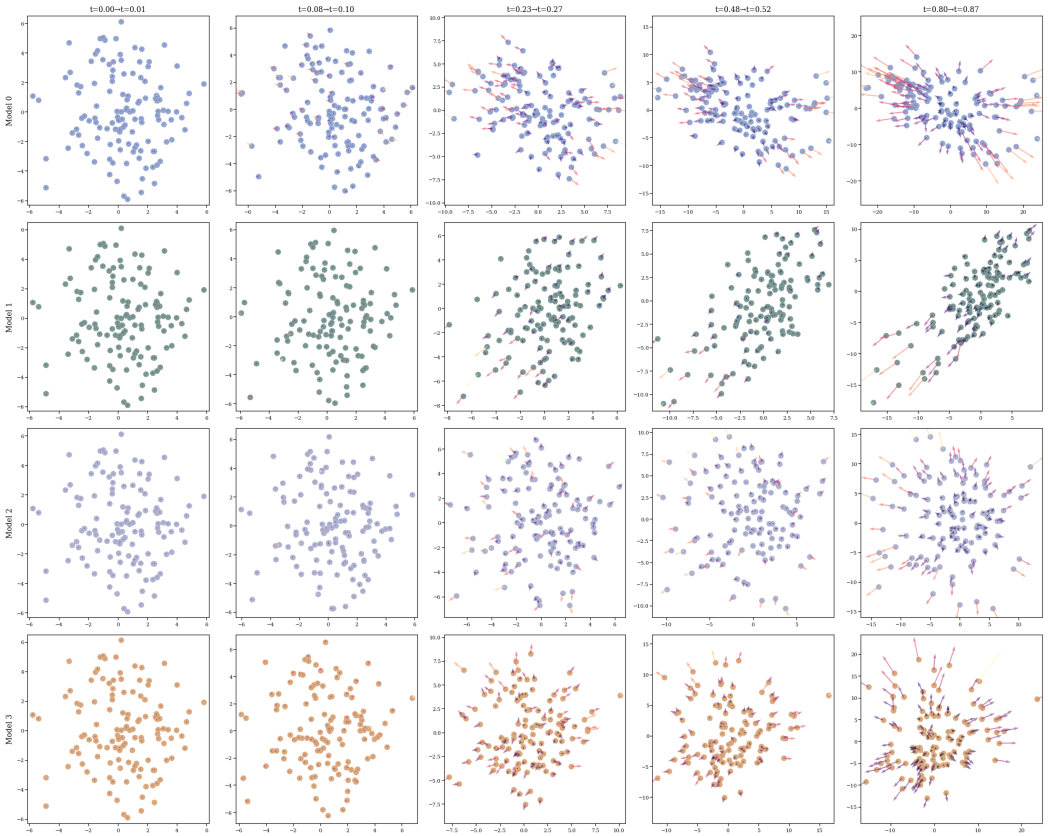

Figure 7: **Visualization of latent flow dynamics across different Fed-ISIC clients.** Each row corresponds to one local model (i.e., client site), and each column shows a different temporal interval during flow simulation by the TAFH. Arrows represent the instantaneous velocity fields $v_\theta(z_t, t)$ of latent particles, visualizing how feature representations evolve over time. Early stages exhibit isotropic motion from Gaussian initialization, while later stages reveal directionally coherent drifts reflecting client-specific domain biases and adaptation trajectories. The figure highlights that different clients exhibit distinct flow patterns, validating the necessity of dynamic modeling for capturing heterogeneous representation evolution in federated settings.

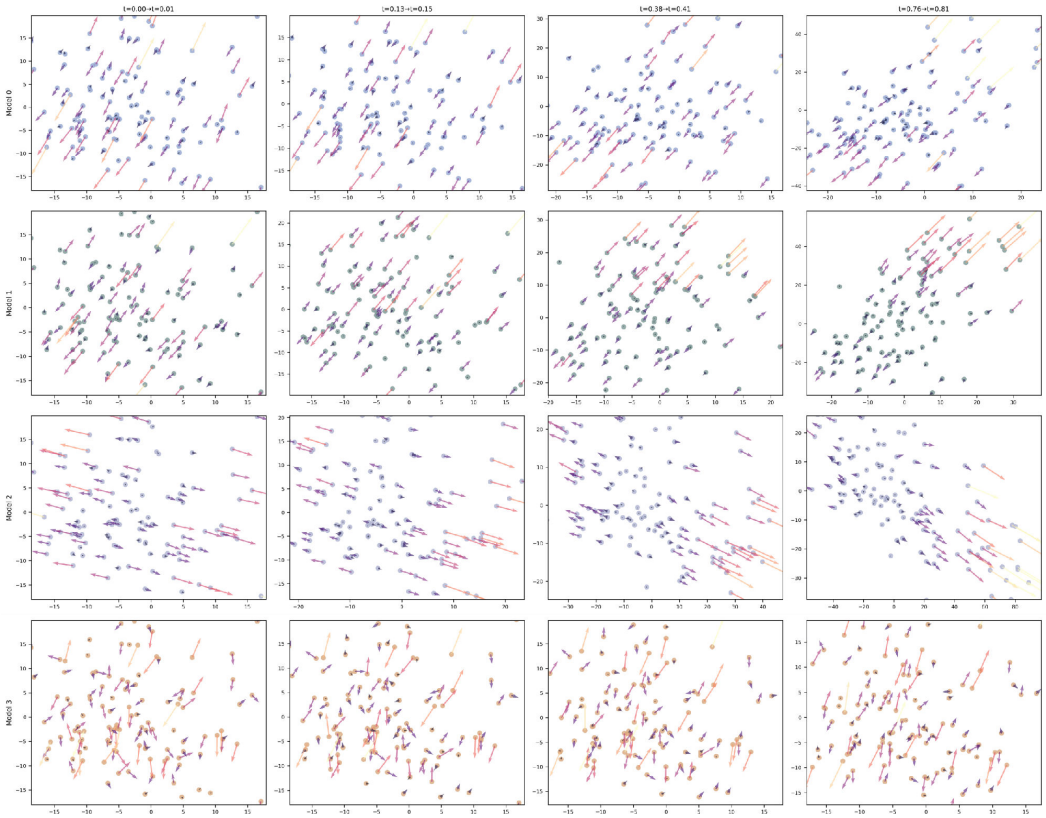

Figure 8: **Visualization of latent flow dynamics across different IPMN-T1W clients.** Each row corresponds to one local model (i.e., client site), and each column shows a different temporal interval during flow simulation by the TAFH. Arrows represent the instantaneous velocity fields $v_\theta(z_t, t)$ of latent particles, visualizing how feature representations evolve over time.

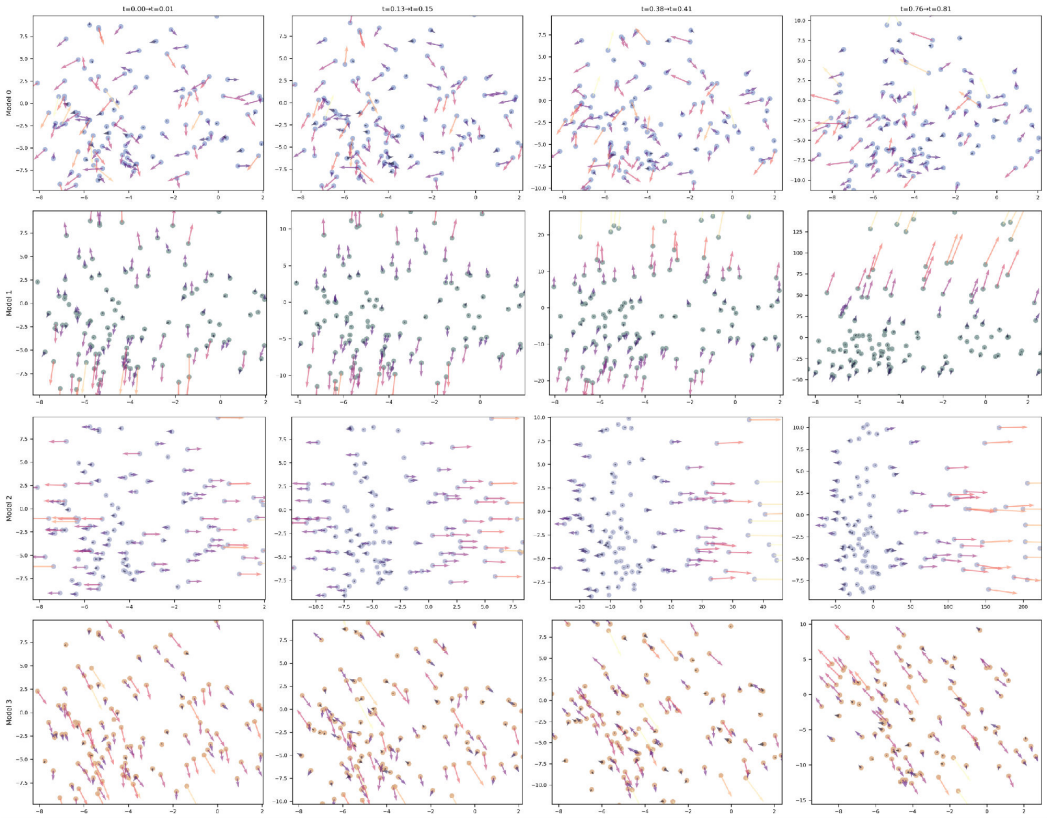

Figure 9: **Visualization of latent flow dynamics across different IPMN-T2W clients.** Each row corresponds to one local model (i.e., client site), and each column shows a different temporal interval during flow simulation by the TAFH. Arrows represent the instantaneous velocity fields $v_\theta(z_t, t)$ of latent particles, visualizing how feature representations evolve over time.

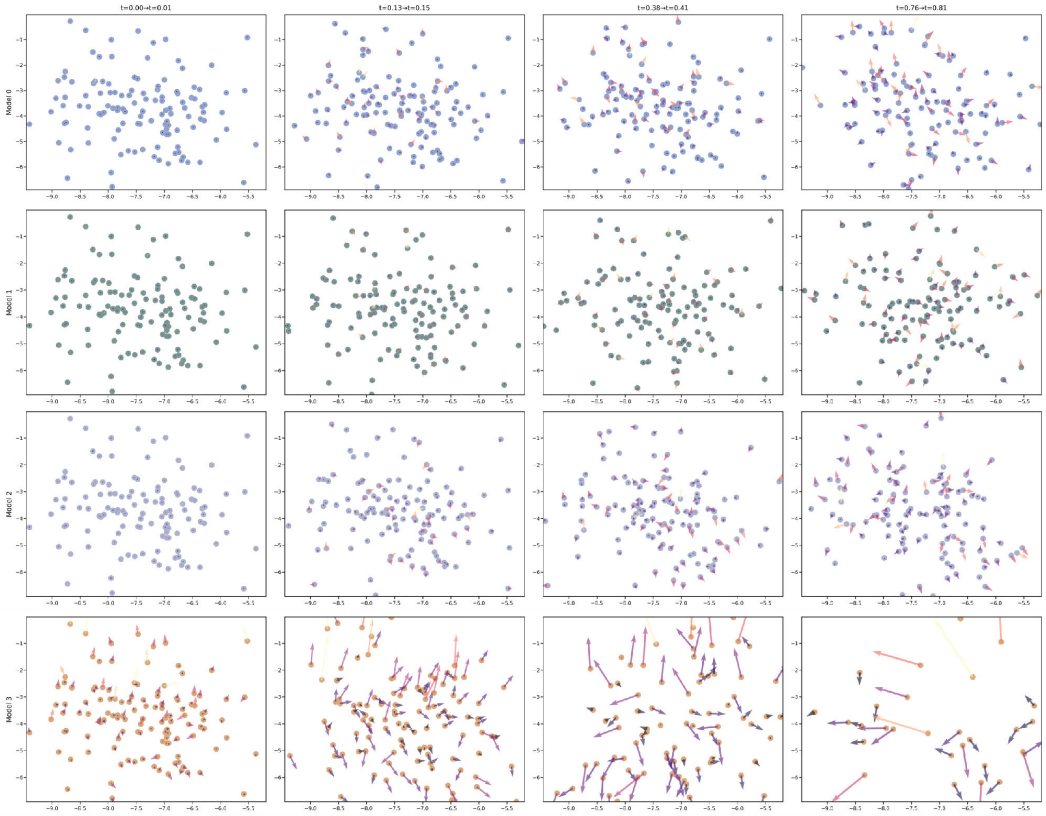

Figure 10: **Visualization of latent flow dynamics across different Fed-Fundus clients.** Each row corresponds to one local model (i.e., client site), and each column shows a different temporal interval during flow simulation by the TAFH. Arrows represent the instantaneous velocity fields $v_\theta(z_t, t)$ of latent particles, visualizing how feature representations evolve over time.

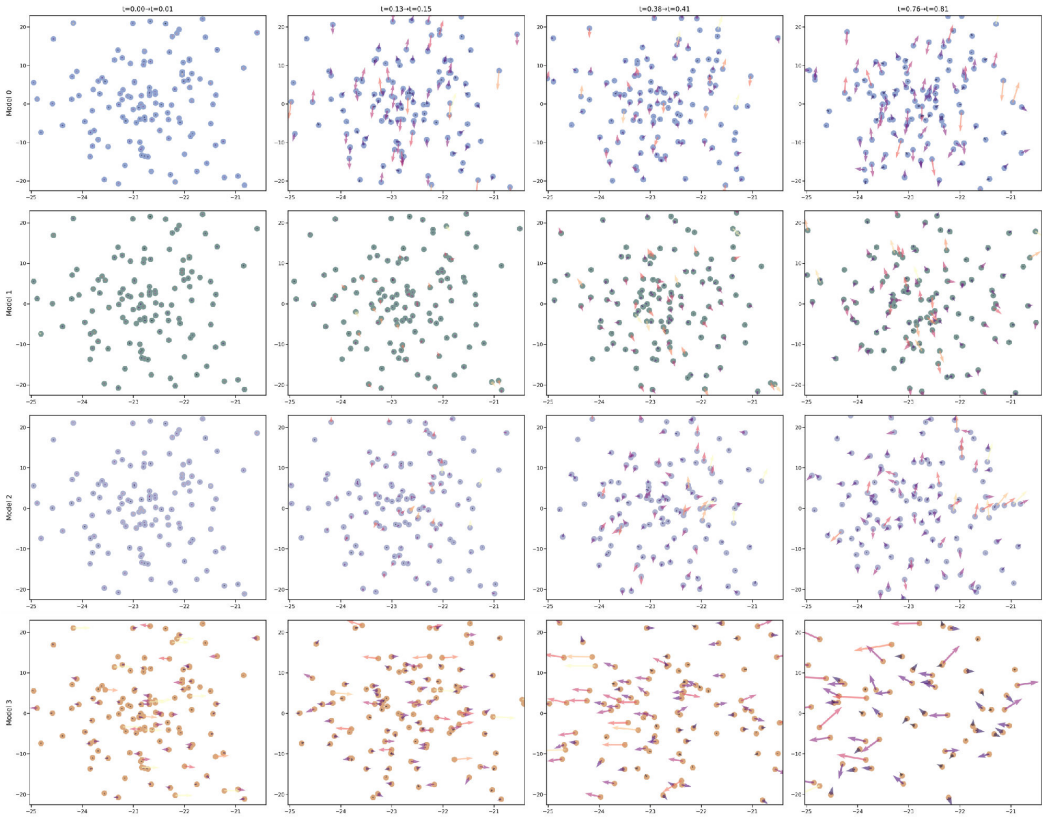

Figure 11: **Visualization of latent flow dynamics across different Fed-Polyp clients.** Each row corresponds to one local model (i.e., client site), and each column shows a different temporal interval during flow simulation by the TAFH. Arrows represent the instantaneous velocity fields $v_\theta(z_t, t)$ of latent particles, visualizing how feature representations evolve over time.

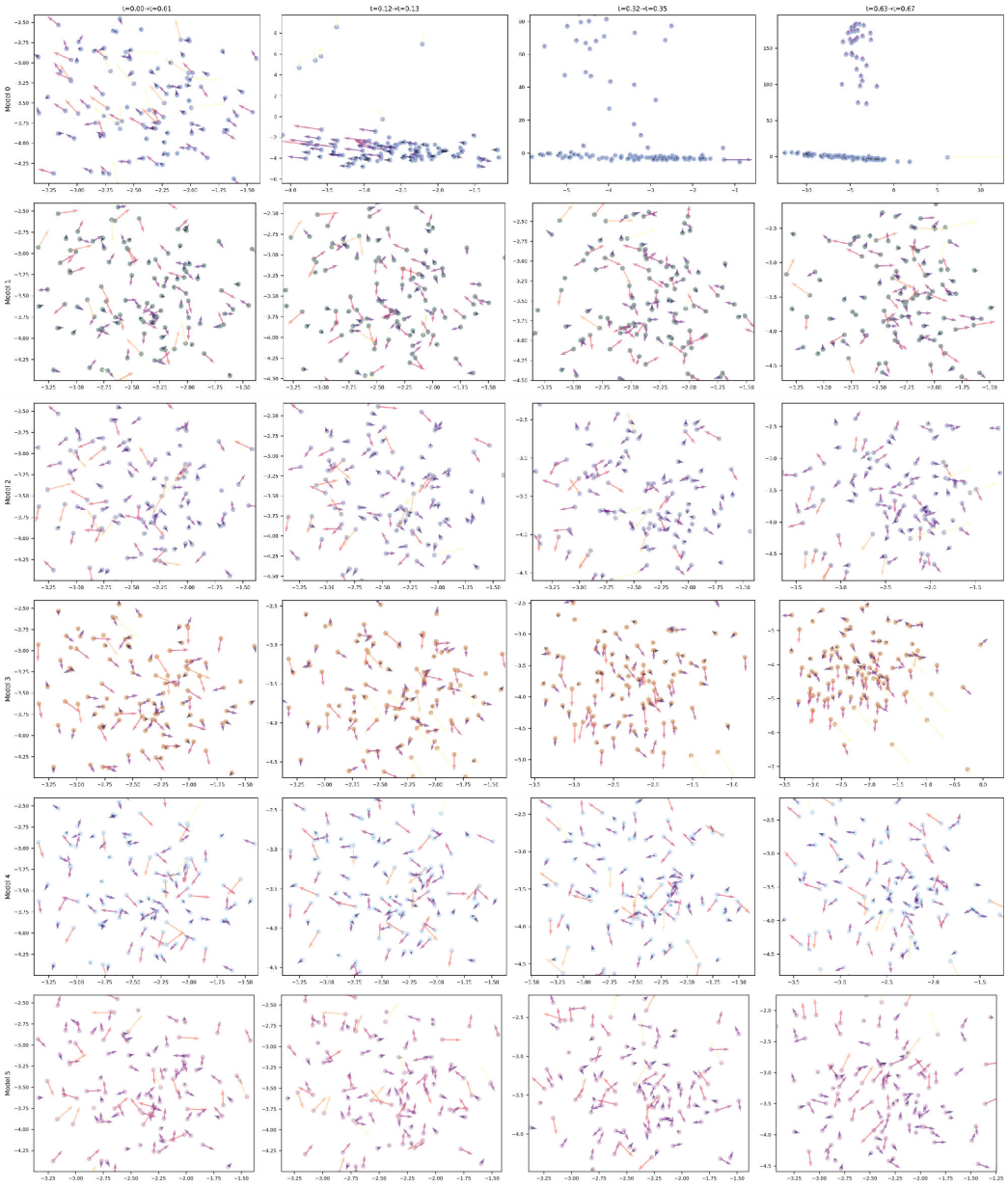

Figure 12: **Visualization of latent flow dynamics across different Fed-Prostate clients.** Each row corresponds to one local model (i.e., client site), and each column shows a different temporal interval during flow simulation by the TAFH. Arrows represent the instantaneous velocity fields $v_\theta(z_t, t)$ of latent particles, visualizing how feature representations evolve over time. Note that Model 0 exhibits a near-linear trajectory collapse at intermediate timesteps, reflecting the highly homogeneous acquisition protocol of the BIDMC cohort rather than numerical instability.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims made in the abstract and introduction accurately reflect the paper's contributions and scope.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of this work in Section 5

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: All assumptions are stated explicitly prior to each proposition. The analysis in the Appendix is intended as design motivation rather than end-to-end performance guarantees.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: This paper disclose all the information needed to reproduce the main experimental results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: We will release our code upon acceptance.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: The training and test details are presented in section J and section 4.1.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We provide p-values and std in Table 1.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The computer details are presented in Section J and Section 4.4. All experiments were conducted on NVIDIA A100-SXM4-40GB GPUs.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have carefully reviewed and adhered to the code of ethics throughout our research and writing process.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the broader impacts in Section 5. Regarding potential negative impacts: while DKT-FDG is designed for privacy-conscious medical FL, misclassification errors in clinical deployment could negatively affect patient outcomes, particularly for underrepresented populations whose imaging data may be less prevalent across participating centers. Additionally, although flow signatures do not contain raw data, their statistical properties could, in principle, leak distributional information if not protected by additional mechanisms such as differential privacy. We note these risks are inherent to federated medical AI systems broadly, and discuss mitigation directions (e.g., compatibility with differential privacy) in the Appendix.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: Our work poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All publicly available assets (models, code, and data) used in this work have been properly credited with original citations. Their respective terms of use have been adhered to throughout this work.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: We do not release new assets in the submission.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This paper does not directly involve crowdsourcing or the collection of data from human subjects. All datasets used are publicly available benchmarks whose original data collection procedures were conducted and ethically approved by their respective releasing institutions.

    Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The paper does not use Large Language Models (LLMs) as an important, original, or non-standard component of the core methods.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.