RETHINKING CROSS-TASK KNOWLEDGE SHARING: MEDICAL FEDERATED MULTI-TASK LEARNING WITH THE INPUT SPACE KNOWLEDGE DECOUPLING

Anonymous authors
Paper under double-blind review

ABSTRACT

Federated multi-task learning (FMTL) seeks to share information across institutions and heterogeneous tasks while preserving client-specific personalization. Existing methods use local task models, their parameters, representations, or outputs, as carriers for federated knowledge sharing. Because potentially transferable and task-specific information are entangled within these carriers, aggregation cannot determine what should be shared and what should remain local: insufficient sharing may miss beneficial knowledge, whereas excessive sharing can cause negative transfer. To address this conflict, we propose IPL-FL, an FMTL framework that keeps personalization-specific knowledge local while separating cross-client knowledge sharing from local personalization, thereby avoiding the knowledge-selection conflict caused by their coupling. IPL-FL keeps task-specific knowledge in frozen local task models and instead communicates input-adaptation generators, preventing local predictors from being directly modified by aggregation. IPL-FL comprises client-side task-guided perturbation learning (TGPL) and server-side relation-aware personalized aggregation (RAPA). TGPL uses local task supervision to optimize task-beneficial input perturbations and transfers the resulting adaptations to shareable generators. RAPA decomposes generator updates into candidate shared information and client-specific residuals, and selectively aggregates compatible and complementary information using update-direction consistency and conditional information gain. Across eight clients spanning three medical tasks with heterogeneous architectures, IPL-FL improves every client over local training, achieves a 5.65% gain, exceeds the strongest applicable FMTL baseline by 5.27%, and offers a 13% efficiency advantage.

1 INTRODUCTION

Medical institutions often perform diverse clinical tasks, such as disease classification, lesion segmentation, and object detection, while their data remain locally isolated due to privacy and regulatory constraints Rieke et al. (2020); Guan et al. (2024). Despite differences in task objectives and supervision forms, these clinical tasks are not independent: they often rely on related anatomical structures, pathological patterns, and lesion characteristics, creating opportunities for knowledge transfer across institutions and tasks Graham et al. (2023). Federated multi-task learning (FMTL) provides a natural paradigm for such medical collaboration by enabling clients to benefit from cross-task knowledge without directly sharing private data Smith et al. (2017); Sun et al. (2024). Unlike conventional federated learning that primarily targets a common task McMahan et al. (2017), FMTL seeks to exploit transferable knowledge across heterogeneous tasks while retaining the task-specific characteristics required by individual clients Smith et al. (2017); Lu et al. (2024). Therefore, a central goal of medical FMTL is to maximize the benefits of cross-task collaboration without compromising task-specific personalization Lu et al. (2024).

Existing FMTL methods commonly rely on local task models as the primary carrier for federated knowledge sharing Smith et al. (2017); Lu et al. (2024); Wu et al. (2026). However, as illustrated in Figure 1, transferable knowledge that may benefit other tasks and task-specific knowledge required for local personalization are often tightly coupled within the same task model. This coupling makes it difficult for federated aggregation to determine which knowledge should be shared and

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

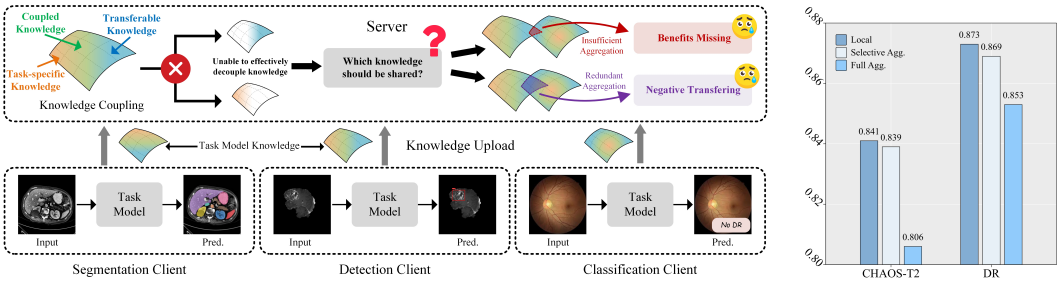

Figure 1: Knowledge-selection conflict in existing FMTL. **Left:** Transferable and task-specific knowledge are coupled in local task models. Insufficient aggregation discards useful cross-task knowledge, whereas redundant aggregation introduces task-irrelevant knowledge and causes negative transfer. **Right:** Cross-task validation between segmentation (Dice) using U-Net with ResNet-18 encoder and classification (Acc) using ResNet-18. Randomly aggregating a subset of encoder causes performance fluctuations by missing benifits, whereas full encoder aggregation substantially degrades performance because of negative transfer.

which should remain local. We define this aggregation conflict induced by knowledge coupling as the conflict in knowledge selection. In this conflict, insufficient selection of transferable knowledge prevents cross-task collaboration from fully realizing its potential benefits, whereas redundant selection may transfer task-specific knowledge to irrelevant clients and consequently cause negative transfer Lu et al. (2024); Wu et al. (2026). Such a conflict places FMTL in a knowledge-selection dilemma, where improving knowledge sharing and preserving task specificity are difficult to achieve simultaneously within coupled task models. This limitation motivates a medical FMTL method that can effectively decouple transferable knowledge from task-specific knowledge.

Motivated by the coupling-induced knowledge-selection conflict, we rethink how knowledge should be shared in FMTL. We attribute this conflict to information isolation between clients and the server, which leaves the server unable to observe how knowledge is coupled within each client and makes it difficult to distinguish transferable knowledge from task-specific knowledge. To address this issue, we propose IPL-FL, an FMTL framework that keeps personalization-specific knowledge local while separating cross-client knowledge sharing from local personalization. Instead of directly aggregating task-model knowledge, IPL-FL freezes local task models and uses dedicated input-adaptation generators as federated knowledge carriers, preventing local predictors from being directly modified during aggregation. Specifically, task-guided perturbation learning (TGPL) uses local task supervision to optimize task-beneficial input perturbations and transfers the resulting adaptations to shareable generators. Relation-aware personalized aggregation (RAPA) further decomposes generator updates into candidate shared information and client-specific residuals, and selectively aggregates compatible and complementary information using update-direction consistency and conditional information gain. In this way, IPL-FL reduces the knowledge-selection conflict while preserving client personalization and enabling effective cross-task knowledge sharing.

Our main contributions are summarized as follows: (1) We propose IPL-FL, a novel FMTL framework that uses TGPL to shift the carrier of knowledge sharing from local task models to input-adapted perturbation generators, enabling client-side decoupling of transferable and task-specific knowledge to alleviate conflicts in knowledge selection. (2) We introduce RAPA, which models cross-client knowledge relations through update-direction consistency and conditional information gain, selectively aggregating beneficial knowledge while preserving client-specific information to reduce negative transfer from task-irrelevant knowledge. (3) We conduct extensive evaluations in fully heterogeneous medical collaboration settings, where IPL-FL achieves a 5.65 % overall improvement over local training and a 5.27 % gain over the strongest applicable FMTL baseline, while providing a 13 % efficiency advantage in federated training.

2 Related Work

2.1 Federated Multi-Task Learning

Federated multi-task learning (FMTL) jointly optimizes personalized models for related distributed tasks Smith et al. (2017). Existing studies focus on sharing task-model knowledge across heteroge-

neous clients. One direction models parameter or update relationships for conflict-aware or personalized aggregation Lu et al. (2024); Liao et al. (2024); Wu et al. (2026). Another separates shared and personalized components or aligns representations to coordinate transferable features while retaining task-specific components Diao et al. (2020); Collins et al. (2021); Oh et al. (2021); Setayesh et al. (2026). When architectural heterogeneity prevents parameter aggregation, knowledge-distillation methods convert model knowledge into predictions, logits, or intermediate features for model-independent transfer Lin et al. (2020); He et al. (2020); Roux et al. (2025). Medical federated learning similarly uses shared representations for multi-disorder diagnosis Huang et al. (2022), addresses heterogeneous clinical label spaces Sun et al. (2024), transfers representations between retinal recognition and segmentation Yang et al. (2025), and coordinates heterogeneous tasks and modalities through personalized components and distillation Zhang et al. (2026). Despite these mechanisms, federated knowledge still originates from task models—their parameters, components, representations, or outputs—leaving transferable and task-specific knowledge coupled. This motivates a knowledge-sharing space beyond local task models.

2.2 Input-space Perturbation Learning

Input-space perturbation adapts visual models without modifying their parameters. Early adversarial reprogramming showed a fixed network could be repurposed for new visual tasks by learning perturbations directly on its inputs Elsayed et al. (2018). This idea was later extended from changing model behavior to improving task performance. Deep Input Transfer optimized medical images using gradients from a fixed segmentation model, demonstrating that task-guided input modification can improve inference without updating the predictor Taghanaki et al. (2019). Recent studies parameterize input-side modifications as learnable prompts or perturbation generators, enabling efficient and input-dependent adaptation of frozen models Jia et al. (2022); Oh et al. (2023); Chen et al. (2024). These studies demonstrate that task-relevant knowledge can be expressed and exploited through input-space modifications while keeping task models unchanged. This observation motivates us to extend input-space perturbation learning to FMTL, where task-guided perturbations are parameterized as transferable knowledge carriers for cross-task federated sharing.

3 Methodology

3.1 Problem Formulation

We consider a medical federated multi-task learning (FMTL) setting with a central server and N clients, indexed by $k \in [N] = \{1, \dots, N\}$. Each client k holds a private labeled dataset $\mathcal{D}_k = \{(x_{k,m}, y_{k,m})\}_{m=1}^{N_k}$ drawn from its own data distribution over the input space $\mathcal{X}_k$ and output space $\mathcal{Y}_k$, where both spaces may differ across clients due to variations in imaging modalities, task types, and label spaces. In particular, clients may perform heterogeneous medical tasks, such as classification, segmentation, and detection, and employ task-specific models $f_k(\cdot; w_k)$ with different architectures and parameter spaces. Using only its local data, each client independently optimizes its task model as $w_k^* = \arg\min_{w_k} \mathbb{E}_{(x_k, y_k) \sim \mathcal{D}_k} \left[\mathcal{L}_k^{\text{task}}(f_k(x_k; w_k), y_k) \right]$, where $\mathcal{L}_k^{\text{task}}$ denotes the task-specific objective of client k. Unlike conventional federated learning that typically assumes a common task or compatible model structures, FMTL coordinates these heterogeneous clients through the server without sharing their private data, enabling each client to benefit from knowledge across related tasks while retaining a customized solution for its own task. Therefore, the goal of medical FMTL is not to learn a single global predictor, but to improve the collection of client-specific task models $\{f_k\}_{k=1}^N$ beyond isolated local training under task, model, data, and modality heterogeneity.

3.2 Overview

As illustrated in Figure 2, IPL-FL addresses the knowledge-selection conflict in medical FMTL by shifting federated sharing away from heterogeneous task models. Each client first trains and freezes its local task model, thereby preserving task-specific knowledge locally. The task-guided perturbation learning (TGPL) then exploits local task supervision to extract task-beneficial input adaptations and distills them into a perturbation generator, externalizing candidate transferable knowledge from the task model into the input space for federated sharing. Clients therefore communicate only

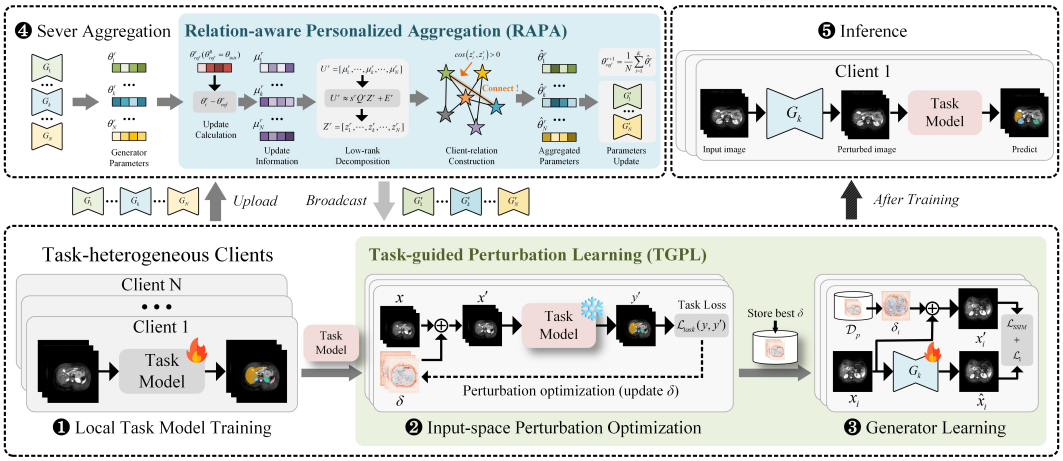

Figure 2: Overview of the proposed IPL-FL framework. Each task-heterogeneous client first trains its local task model. Task-guided perturbation learning (TGPL) then freezes the task model, optimizes task-beneficial input perturbations, and distills the selected perturbations into a generator. At the server, relation-aware personalized aggregation (RAPA) estimates cross-client relations from generator updates and produces personalized aggregated generators. During inference, each generator adapts the input before it is processed by the corresponding task model.

generator parameters rather than their task models. At the server, the relation-aware personalized aggregation (RAPA) further identifies compatible and complementary knowledge from generator updates and performs personalized aggregation, reducing task-irrelevant transfer while retaining client-specific information. The resulting personalized generators are returned to individual clients and used to adapt inputs before prediction by the frozen local task models. In this way, IPL-FL decouples local task-specific knowledge from federated knowledge sharing and enables selective cross-task collaboration in the input space.

3.3 Task-guided Perturbation Learning

Local Task Model Training. As illustrated in Figure 2, each client $k \in \{1, \ldots, N\}$ first trains its local task model $f_k(\cdot; w_k)$ using its private dataset $\mathcal{D}_k$. Specifically, the task model is optimized as $w_k^* = \arg\min_{w_k} \mathbb{E}_{(x_k, y_k)\sim\mathcal{D}_k} \left[\mathcal{L}_k^{\text{task}}(f_k(x_k; w_k), y_k)\right]$, where $\mathcal{L}_k^{\text{task}}$ denotes the task-specific objective for client k. After local training, the optimized parameters are fixed as $w_k = w_k^*$ and are not updated during subsequent federated communication. This preserves the task-specific knowledge learned for each heterogeneous local task and prevents it from being directly altered by cross-client aggregation. The resulting frozen task model $f_k(\cdot; w_k^*)$ is then introduced into task-guided perturbation learning (TGPL) to guide the subsequent input-space perturbation optimization.

Input-space Perturbation Optimization. Given the frozen task model $f_k(\cdot; w_k^*)$, TGPL first searches for task-beneficial perturbations directly in the input space. For each local sample $(x_{k,m}, y_{k,m}) \in \mathcal{D}_k$, an independent perturbation $\delta_{k,m}$ is initialized as $\delta_{k,m}^0 = 0$ and optimized while keeping w_k^* fixed. The perturbed input is defined as $x'_{k,m} = \text{clip}(x_{k,m} + \delta_{k,m})$, and the perturbation is optimized by minimizing the task-specific objective:

$$\delta_{k,m}^* = \arg \min_{\|\delta_{k,m}\|_\infty \le \epsilon} \mathcal{L}_k^{\text{task}} \left(f_k(\text{clip}(x_{k,m} + \delta_{k,m}); w_k^*), y_{k,m}\right), \tag{1}$$

where ϵ constrains the magnitude and $\text{clip}(\cdot)$ restricts the image to the valid input range.

Different from universal perturbations shared across samples, TGPL optimizes a sample-specific perturbation for each $x_{k,m}$, such that each perturbation is explicitly associated with its corresponding input. Moreover, unlike adversarial perturbations that maximize the task loss, TGPL searches in the opposite direction to improve the prediction of the frozen task model. During P optimization steps, the perturbation yielding the lowest task loss is continuously retained, with $\delta = 0$ included as the initial candidate. Algorithm 1 summarizes this general optimization procedure. We adopt Adam to optimize $\delta_{k,m}$, while other gradient-based optimizers can also be readily applied.

Algorithm 1 Task-Guided Input-space Perturbation Optimization

Require: Sample $(x_{k,m}, y_{k,m})$, frozen task model $f_k(\cdot; w_k^*)$, task loss $\mathcal{L}_k^{\text{task}}$, perturbation budget ϵ, optimization steps P
Ensure: Optimal perturbation $\delta_{k,m}^*$ and optimized input $x_{k,m}^*$

```
Initialize δ^0_{k,m} ← 0 and δ*_{k,m} ← 0
L_best ← L^task_k(f_k(x_{k,m}; w*_k), y_{k,m})
for t = 1, ..., P do
    x'_{k,m} ← clip(x_{k,m} + δ^{t-1}_{k,m})
    L ← L^task_k(f_k(x'_{k,m}; w*_k), y_{k,m})
    Update δ^{t-1}_{k,m} toward decreasing L using a gradient-based optimizer
    Project the updated perturbation onto ||δ||_∞ ≤ ε to obtain δ^t_{k,m}
    if L < L_best then
        δ*_{k,m} ← δ^t_{k,m}
        L_best ← L
    end if
end for
x*_{k,m} ← clip(x_{k,m} + δ*_{k,m})
Store (x_{k,m}, x*_{k,m}) in D_p
return δ*_{k,m}, x*_{k,m}
```

After P optimization steps, the best perturbation is denoted as $\delta_{k,m}^*$, and the corresponding optimized input is obtained as $x_{k,m}^* = \text{clip}(x_{k,m} + \delta_{k,m}^*)$. We store the original and optimized input pairs in the local perturbation bank $\mathcal{D}_p = \left\{(x_{k,m}, x_{k,m}^*)\right\}_{m=1}^{N_k}$. In this way, task supervision from the frozen predictor is transferred into sample-specific input adaptations, while $\mathcal{D}_p$ serves as the explicit bridge between input-space perturbation optimization and the subsequent generator learning.

Generator Learning. After input-space perturbation optimization, the optimal sample-specific perturbations are stored in the local perturbation bank $\mathcal{D}_p$. For each local sample $x_{k,m}$, its stored perturbation $\delta_{k,m}^*$ is used to construct the optimized target $x'_{k,m} = \text{clip}(x_{k,m} + \delta_{k,m}^*)$. The TGPL then introduces a perturbation generator $G_k(\cdot; \theta_k)$ to learn this task-guided input transformation, producing $\hat{x}_{k,m} = G_k(x_{k,m}; \theta_k)$. In this stage, the frozen task model is no longer involved in parameter optimization; instead, the task-beneficial information captured by the sample-specific perturbations is transferred from $\mathcal{D}_p$ to the parameterized generator.

The generator is trained to reproduce the optimized inputs by jointly preserving their structural and pixel-level characteristics. Specifically, we optimize

$$\mathcal{L}_k^G = \frac{1}{N_k} \sum_{m=1}^{N_k} \left[1 - \text{SSIM}\big(\hat{x}_{k,m}, x'_{k,m}\big) + \lambda_1 \left\|\hat{x}_{k,m} - x'_{k,m}\right\|_1\right], \tag{2}$$

where the SSIM term encourages structural consistency with the optimized target, while the L_1 term constrains the pixel-level adaptation. TGPL is not tied to a specific generator architecture. In our implementation, we adopt FCDenseNet (Tiramisu) as the default perturbation generator, while other image-to-image generation architectures can be readily substituted.

At communication round r, client k locally updates its generator from θ_k^r to $\theta_k^{r,+}$ by minimizing Eq. 2, and uploads only $\theta_k^{r,+}$ to the server for subsequent federated aggregation. The local task model, perturbation bank $\mathcal{D}_p$, and private data remain on the client. By parameterizing the sample-wise optimized perturbations with G_k, TGPL replaces iterative perturbation search with a single forward transformation and, more importantly, transfers candidate shareable knowledge from the task model to an independent input-space carrier. The generator therefore serves as the sole federated knowledge carrier. This enables subsequent cross-client knowledge sharing without directly modifying or aggregating the heterogeneous local task models. Appendix A.5 provides further theoretical analyses.

3.4 RELATION-AWARE PERSONALIZED AGGREGATION

Generator Parameter Low-Rank Decomposition. Following local generator learning, in Figure 2, each client $k \in \{1, \ldots, N\}$ uploads its generator parameters θ_k^r to the server at communication round r. Although TGPL transfers candidate shareable knowledge from task models to perturbation generators, the resulting generator updates may still contain both cross-client common information and client-specific information. Directly aggregating the entire generator therefore risks reintroducing the knowledge-selection conflict. To obtain a more suitable space for selective sharing, RAPA first decomposes generator updates into public and private components.

The server maintains a reference generator θ_{ref}^r to provide a common coordinate origin for all clients. Since all perturbation generators start from the same initialization, the reference is initialized as $\theta_{\text{ref}}^0 = \theta_{\text{init}}$. For client k, RAPA computes its relative update as $\mu_k^r = \text{vec}(\theta_k^r - \theta_{\text{ref}}^r)$ and stacks all client updates into $U^r = [\mu_1^r, \ldots, \mu_N^r]$. We then perform truncated singular value decomposition on the normalized update matrix to obtain

$$U^r \approx s^r Q^r Z^r + E^r, \tag{3}$$

where s^r is a scale factor, Q^r denotes the orthogonal basis of the shared low-rank subspace, $Z^r = [z_1^r, \ldots, z_N^r]$ contains the client coordinates in this subspace, and E^r represents the residual updates that cannot be explained by the shared structure. We retain the minimum rank covering 90% of the singular-value energy, with the maximum rank limited to four. Accordingly, $s^r Q^r Z^r$ serves as the candidate public knowledge for cross-client sharing, whereas E^r preserves client-specific information. The subsequent relation modeling is therefore performed on Z^r, which compactly characterizes client relations within the shared subspace, while E^r remains unchanged to avoid unnecessary interference with private knowledge. Appendix A.6 provide further theoretical analyses.

Client Relation Aggregation. After low-rank decomposition, RAPA constructs client relation graph in the shared subspace via $Z^r = [z_1^r, \ldots, z_N^r]$. Since Q^r is a basis shared by all clients, the client-specific relation within the public subspace is characterized by their coordinates z_k^r. For a target client k, RAPA first measures the directional consistency with client j as $a_{kj}^r = \max(\cos(z_k^r, z_j^r), 0)$, such that negatively aligned updates are excluded from knowledge sharing. However, directional consistency alone may select redundant clients. Therefore, given the selected neighbor set $\mathcal{S}_k$, we further evaluate the conditional information gain of candidate client j as

$$g_{j|\mathcal{S}_k}^r = \frac{1}{2} \log \left(1 + \frac{\text{Var}\left(z_j^r \mid Z_{\mathcal{S}_k}^r\right)}{\sigma^2} \right), \tag{4}$$

where $Z_{\mathcal{S}_k}^r$ contains the public coordinates of the selected neighbors and the conditional variance is computed through a regularized Schur complement. Thus, the information gain is also derived entirely from Z^r, without involving the private residual E^r. RAPA combines compatibility and complementarity as $R_{kj}^r = a_{kj}^r g_{j|\mathcal{S}_k}^r$ and greedily selects informative neighbors for each client, followed by symmetrization to obtain the relation matrix R^r. In this way, an edge is established only when another client provides both directionally compatible and non-redundant public knowledge. Appendices A.7 and A.3 provide further theoretical analyses.

Given R^r, RAPA constructs the graph Laplacian $L^r = D^r - R^r$ and propagates knowledge only within the public subspace:

$$Y^r = Z^r \left(I + \lambda L^r\right)^{-1}, \qquad \widehat{U}^r = s^r Q^r Y^r + E^r, \tag{5}$$

where λ controls the aggregation strength. The graph smoothing encourages strongly related clients to exchange public knowledge, while weakly related or conflicting clients are not forced toward the same solution. Importantly, E^r is directly added back without cross-client smoothing, such that the personalized update $\widehat{\mu}_k^r$ jointly contains relation-aware public knowledge and the original client-specific residual. The personalized generator is then reconstructed as $\widehat{\theta}_k^r = \theta_{\text{ref}}^r + \text{unvec}(\widehat{\mu}_k^r)$. Therefore, RAPA avoids indiscriminate aggregation while preserving the client-specific information required by heterogeneous tasks.

The server returns $\widehat{\theta}_k^r$ to each client rather than broadcasting a global generator. After aggregation, the server updates the reference generator as $\theta_{\text{ref}}^{r+1} = \frac{1}{N} \sum_{k=1}^{N} \widehat{\theta}_k^r$, which serves only as the common

anchor for constructing client updates in the next communication round. After federated training, each client uses its personalized generator to adapt the input before feeding it into the frozen local task model for inference. Appendices A.8 and A.9 provide further theoretical analyses.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Datasets and Tasks. We evaluate IPL-FL on seven public medical imaging datasets spanning three task categories. Segmentation uses CHAOS CT/MRI Kavur et al. (2021) and ISIC 2018 Codella et al. (2019); classification uses Pneumonia Kermany et al. (2018), DDR Li et al. (2019), and PCam Veeling et al. (2018); and detection uses the RSNA Pneumonia Detection Challenge Shih et al. (2019) and Ultralytics Brain Tumor Ultralytics (2023) datasets.

Implementation Details. We implement IPL-FL in PyTorch on NVIDIA RTX A6000 GPUs. Convolutional and Transformer task models use NAdam and AdamW, respectively. Task-guided perturbations are optimized for 100 steps, while generators are trained for 100 communication rounds with one local epoch per round, a learning rate of 10^{-3}, and a batch size of 32 (128 for Tiramisu on PCam). We retain official data splits when available; otherwise, data are partitioned into training, validation, and test sets at an $8{:}1{:}2$ ratio with a fixed seed of 42 and stratified sampling. Subject-level grouping is used to prevent data leakage.

Performance Metrics. We use Dice score for segmentation, accuracy (Acc) for classification, and mAP_{50} for detection. Overall federated benefit is measured by the mean relative improvement over local training, $\Delta = \frac{1}{N}\sum_{k=1}^{N} \frac{M_k^{\text{fed}} - M_k^{\text{local}}}{M_k^{\text{local}}} \times 100\%$, where M_k^{fed} and M_k^{local} are the federated and local performances on client k. A larger Δ indicates greater collaborative benefit.

Baselines. Because most existing FMTL methods assume fully or partially homogeneous client models, they do not apply to our setting with heterogeneous tasks and model architectures. We employ two baseline groups: **G1.** Compare with baselines that can accommodate heterogeneous client models, including FedMuscle Setayesh et al. (2026), FedHeNN Makhija et al. (2022), and MH-pFLID Xie et al. (2024). **G2.** Compare with baselines based on TGPL, including FedAvg McMahan et al. (2017), FedProx Li et al. (2020), SCAFF Karimireddy et al. (2020), FedAdam Reddi et al. (2021), Ditto Li et al. (2021), FedSim Pillutla et al. (2022), FedAS Yang et al. (2024), FedAWA Shi et al. (2025), FedOMG Nguyen et al. (2025), ConFREE Zheng et al. (2025), and FedMerge Chen et al. (2026). All methods are evaluated under the same implementation and environment.

4.2 MAIN RESULTS

Quantitative Analysis of Full FMTL Methods. Table 1 compares IPL-FL with FMTL methods applicable to heterogeneous client models. IPL-FL achieves the best performance on all eight clients and improves the overall performance over local training by 5.65%. In contrast, FedHeNN and MH-pFLID produce negative overall gains of -1.92% and -1.03%, respectively, while FedMuscle yields only a 0.38% improvement. These results demonstrate that input-space knowledge decoupling and relation-aware personalized aggregation enable IPL-FL to transfer beneficial knowledge while reducing cross-task interference across segmentation, classification, and detection.

Quantitative Analysis of FMTL Aggregation. Table 2 evaluates different aggregation methods using the same TGPL knowledge carrier. Local+TGPL improves the overall performance over local training by 2.52%, confirming that TGPL effectively extracts task-beneficial input adaptations. With RAPA, IPL-FL further increases the gain to 5.65%, achieves the best results on six of eight clients, and outperforms all competing aggregation methods overall. The additional 3.13 percentage-point gain over Local+TGPL shows that the improvement stems not only from task-beneficial perturbation learning but also from relation-aware cross-client sharing. This advantage highlights the importance of RAPA, which selectively integrates compatible and complementary updates while preserving client-specific information, thereby reducing negative transfer across heterogeneous tasks. Appendix A.2 provides more comparisons.

Table 1: Comparison between IPL-FL with FMTL methods under heterogeneous tasks and models. Bold denotes the best result for each client.

Task	Metrics	Clitnes	Datasets	Task Model	Methods				
					Local	IPL-FL (Ours)	FedHeNN ICML22'	MH-pFLID ICML24'	FedMuscle ICLR26'
Seg.	Dice Score	C1	CHAOS-CT	U-Net	0.7577	**0.7695**	0.7514	0.7469	0.7618
		C2	CHAOS-MRI	Swin-UNETR	0.3536	**0.3786**	0.3478	0.3424	0.3519
		C3	ISIC	U-Net	0.8243	**0.8849**	0.8175	0.8118	0.8297
Cls.	Accuracy	C4	Pneumonia	ResNet-18	0.9399	**0.9709**	0.9341	0.9297	0.9442
		C5	DDR	Swin-Tiny	0.8041	**0.8458**	0.7986	0.7933	0.8063
		C6	PCam	ResNet-18	0.8068	**0.8361**	0.8019	0.7964	0.8105
Det.	mAP@50	C7	RENA	YOLO11n	0.2797	**0.3094**	0.2742	0.2705	0.2811
		C8	BRAIN	YOLOv8	0.4948	**0.5267**	0.4896	0.4837	0.4979
Δ (%)					0.00	**+5.65**	-1.03	-1.92	0.38

Table 2: Comparison of FMTL aggregation methods under heterogeneous tasks and models. Local+TGPL denotes independently training the TGPL generator at each client without federated aggregation. Bold denotes the best result for each client.

Task	Metrics	Clitnes	Datasets	Task Model	Methods											
					Local	Local +TGPL	IPL-FL (Ours)	FedAvg AIS17'	FedProx MLS20'	SCAFF ICML20'	FedAdam ICLR21'	Ditto ICML21'	FedSim ICML22'	FedAS CVPR24'	FedAWA CVPR25'	ConFREE AAAI25'	FedMerge AAAI26'
Seg.	Dice Score	C1	CHAOS-CT	U-Net	0.7577	0.7684	0.7695	0.7641	0.7669	0.7658	0.7681	0.7687	0.7690	0.7679	0.7707	**0.7712**	0.7699
		C2	CHAOS-MRI	Swin UNETR	0.3536	0.3708	**0.3786**	0.3668	0.3716	0.3697	0.3735	0.3722	0.3751	0.3744	0.3763	0.3777	0.3768
		C3	ISIC	U-Net	0.8243	0.8649	**0.8849**	0.8543	0.8607	0.8678	0.8704	0.8764	0.8697	0.8749	0.8801	0.8818	0.8776
Cls.	Accuracy	C4	Pneumonia	ResNet-18	0.9399	0.9532	**0.9709**	0.9517	0.9564	0.9593	0.9608	0.9652	0.9619	0.9680	0.9665	0.9696	0.9674
		C5	DDR	Swin-T	0.8041	0.8362	0.8458	0.8275	0.8310	0.8338	0.8386	0.8408	0.8380	0.8433	**0.8470**	0.8439	0.8448
		C6	PCam	ResNet-18	0.8068	0.8094	**0.8361**	0.8123	0.8178	0.8164	0.8264	0.8215	0.8246	0.8268	0.8291	0.8326	0.8340
Det.	mAP@50	C7	RENA	YOLO11n	0.2797	0.2833	**0.3094**	0.2821	0.2860	0.2896	0.2917	0.2990	0.2941	0.2978	0.3022	0.3064	0.3026
		C8	BRAIN	YOLOv8	0.4948	0.5045	**0.5267**	0.5016	0.5054	0.5102	0.5108	0.5140	0.5149	0.5188	0.5180	0.5225	0.5220
Δ (%)					0.00	+2.52	**+5.65**	+1.91	+2.70	+3.06	+3.63	+4.11	+3.88	+4.38	+4.82	+5.25	+4.96

Table 3: Ablation on perturbation factor ϵ.

ϵ	Task		
	Seg.	Cls.	Det.
w/o	0.6452	0.8503	0.3873
0.01	0.6544	0.8592	0.3946
0.05	0.6737	0.8783	0.4115
0.1	**0.6777**	**0.8843**	**0.4181**
0.5	0.6690	0.8787	0.4071
1.0	0.6482	0.8545	0.3893

Table 4: Ablation on generator of TGPL.

Gen.	Task		
	Seg.	Cls.	Det.
w/o	0.6452	0.8503	0.3873
NAFNet	0.6753	0.8820	0.4146
Pix2pix	0.6697	0.8761	0.4074
RDDM	0.6762	0.8807	0.4151
Tiramisu	**0.6777**	**0.8843**	**0.4181**

Table 5: Ablation on optimizer mode of TGPL.

Opt.	Task		
	Seg.	Cls.	Det.
w/o	0.6452	0.8503	0.3873
Adam	**0.6777**	**0.8843**	**0.4181**
PGD	0.6758	0.8829	0.4149
MI-FGSM	0.6746	0.8816	0.4127

4.3 Ablation and Analysis

Perturbation Factor Ablation. The perturbation factor ϵ defines the maximum magnitude of the task-guided reference perturbation and is important for balancing task-beneficial input adaptation against preservation of the original image semantics. Table 3 reports task-level averages, where Seg., Cls., and Det. denote the mean performance over all clients conducting segmentation, classification, and detection, respectively. Performance consistently improves as ϵ increases from 0.01 to 0.1, with $\epsilon = 0.1$ achieving the best results across all three task types. Further increasing ϵ degrades performance because overly strong perturbations may distort semantic information.

Perturbation Generator Ablation. Table 4 compares NAFNet Chen et al. (2022), Pix2pix Isola et al. (2017), RDDM Liu et al. (2024), and Tiramisu Jégou et al. (2017) as TGPL generators. All outperform the setting without a generator across the three task types, confirming the architectural flexibility of TGPL. Tiramisu achieves the best average results and is therefore adopted as the default generator for parameterizing transferable input adaptations.

Perturbation Optimization Ablation. Table 5 evaluates different methods for optimizing the task-guided reference perturbations. Adam Kingma & Ba (2015), PGD Madry et al. (2018), and MI-FGSM Dong et al. (2018) all substantially outperform the setting without perturbation optimization across the three task types, confirming that task-beneficial reference perturbations provide effective supervision for TGPL. Adam achieves the best average performance on segmentation, clas-

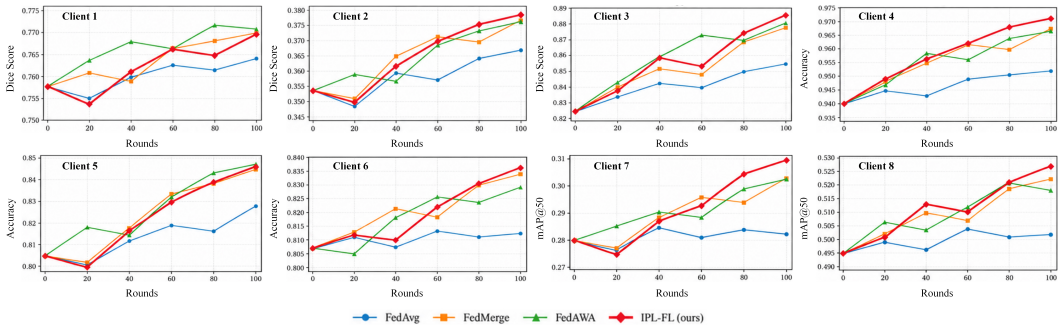

Figure 3: Performance over communication rounds for all eight clients.

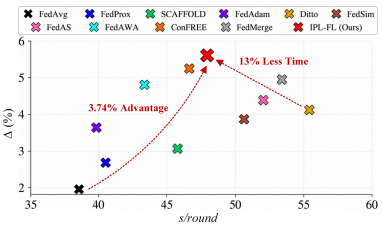

Figure 4: Overall performance gain versus per-round training time.

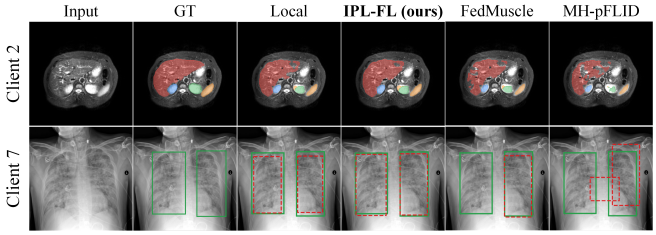

Figure 5: Qualitative comparison on segmentation clients C1 and detection clients C7.

sification, and detection, while the competitive results of PGD and MI-FGSM demonstrate that our framework remains effective across different perturbation optimization strategies. We therefore use Adam as the default optimizer. Appendix A.1 reports more ablations.

Iteration Analysis. To examine the training dynamics, we save model weights at fixed communication rounds and evaluate each checkpoint on the corresponding client test set. Following the standard convergence analysis in federated learning, Figure 3 compares the client-wise performance trajectories of IPL-FL with FedAvg, FedMerge, and FedAWA. FedAvg generally converges to lower performance, while FedMerge and FedAWA exhibit client-dependent fluctuations. In contrast, IPL-FL shows a clear overall upward trend and achieves the best or competitive final performance across clients, demonstrating stable and effective cross-task knowledge transfer during federated training.

Gain and Efficiency Analysis. Figure 4 compares the overall performance gain and per-round training time of different aggregation methods. IPL-FL achieves the highest gain of 5.65% at approximately 48 seconds per round, exceeding FedAvg by 3.74% while requiring about 13% less time than Ditto. These results demonstrate that IPL-FL provides a favorable trade-off between cross-task collaboration benefit and training efficiency.

Visualization Analysis. Figure 5 visualizes the final results on segmentation clients C1–C2 and detection clients C7–C8. Compared with local training and existing FMTL methods, IPL-FL produces more accurate segmentation and lesion localization, demonstrating its ability to effectively exploit knowledge across heterogeneous visual tasks. Appendix A.4 provides more visualization.

5 Conclusion

In this work, we study medical federated multi-task learning under heterogeneous tasks and model architectures. Existing task-model-centric sharing tightly couples transferable and task-specific knowledge, leading to conflicts in knowledge selection during aggregation. To address this issue, we propose IPL-FL, which preserves task-specific knowledge in local models while shifting cross-client sharing to input-adaptation generators. TGPL learns task-beneficial input adaptations, while RAPA selectively aggregates compatible and complementary generator knowledge. Experiments across medical classification, segmentation, and detection tasks show that IPL-FL consistently improves cross-task collaboration, achieving a 5.65% overall relative gain over local training. These results demonstrate the effectiveness of input-space knowledge decoupling for medical FMTL.

AI USE STATEMENT

In this work, we used generative AI tools for language polishing and for providing suggestions on how to organize and present experimental results. We have not used generative AI tools for generating research ideas, designing the methodology, implementing the proposed method, conducting experiments, generating experimental data, or producing the final presentation of experimental results, and other required disclosure tasks are not applicable to this work. Additionally, generative AI tools were used to improve the clarity, grammar, and readability of the manuscript and to suggest possible ways of organizing experimental results. We have reviewed all AI-assisted work. All technical content, methodological descriptions, experimental results, numerical values, figures, tables, and their final presentation were independently produced or verified by the authors. All AI-assisted language revisions and presentation suggestions were manually reviewed to ensure that they did not alter the intended scientific meaning or claims. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies medical federated multi-task learning using publicly available medical imaging datasets and does not involve the collection of new patient data or direct interaction with human subjects. Our framework is designed to support privacy-preserving collaboration by keeping local data and task models at individual clients. Nevertheless, medical AI systems may inherit biases from the underlying datasets or exhibit uneven performance across populations, tasks, and imaging conditions. The proposed method is intended for research purposes rather than direct clinical decision-making. We encourage careful validation with representative data and appropriate clinical oversight before any real-world deployment.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we provide detailed descriptions of the proposed method, experimental settings, datasets, evaluation metrics, and implementation details in the main paper and supplementary material. The source code and configurations used in our experiments will be publicly released upon acceptance.

REFERENCES

Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. In *European conference on computer vision*, pp. 17–33. Springer, 2022.

Shutong Chen, Tianyi Zhou, Guodong Long, Jing Jiang, and Chengqi Zhang. Fedmerge: Federated model merging for personalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 20253–20261, 2026.

Ziyang Chen, Yongsheng Pan, Yiwen Ye, Mengkang Lu, and Yong Xia. Each test image deserves a specific prompt: Continual test-time adaptation for 2d medical image segmentation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11184–11193. IEEE, 2024.

Noel Codella, Veronica Rotemberg, Philipp Tschandl, M Emre Celebi, Stephen Dusza, David Gutman, Brian Helba, Aadi Kalloo, Konstantinos Liopyris, Michael Marchetti, et al. Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic). *arXiv preprint arXiv:1902.03368*, 2019.

Liam Collins, Hamed Hassani, Aryan Mokhtari, and Sanjay Shakkottai. Exploiting shared representations for personalized federated learning. In *International conference on machine learning*, pp. 2089–2099. PMLR, 2021.

Enmao Diao, Jie Ding, and Vahid Tarokh. Heterofl: Computation and communication efficient federated learning for heterogeneous clients. *International Conference on Learning Representations*, 2020.

Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9185–9193. IEEE, 2018.

Gamaleldin F Elsayed, Ian Goodfellow, and Jascha Sohl-Dickstein. Adversarial reprogramming of neural networks. *International Conference on Learning Representations*, 2018.

Simon Graham, Quoc Dang Vu, Mostafa Jahanifar, Shan E Ahmed Raza, Fayyaz Minhas, David Snead, and Nasir Rajpoot. One model is all you need: multi-task learning enables simultaneous histology image segmentation and classification. *Medical Image Analysis*, 83:102685, 2023.

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

Chaoyang He, Murali Annavaram, and Salman Avestimehr. Group knowledge transfer: Federated learning of large cnns at the edge. *Advances in neural information processing systems*, 33:14068–14080, 2020.

Zhi-An Huang, Yao Hu, Rui Liu, Xiaoming Xue, Zexuan Zhu, Linqi Song, and Kay Chen Tan. Federated multi-task learning for joint diagnosis of multiple mental disorders on mri scans. *IEEE Transactions on Biomedical Engineering*, 70(4):1137–1149, 2022.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 5967–5976. Ieee, 2017.

Simon Jégou, Michal Drozdzal, David Vazquez, Adriana Romero, and Yoshua Bengio. The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation. In *2017 IEEE conference on computer vision and pattern recognition workshops (CVPRW)*, pp. 1175–1183. IEEE, 2017.

Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European conference on computer vision*, pp. 709–727. Springer, 2022.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, pp. 5132–5143. PMLR, 2020.

A Emre Kavur, N Sinem Gezer, Mustafa Barış, Sinem Aslan, Pierre-Henri Conze, Vladimir Groza, Duc Duy Pham, Soumick Chatterjee, Philipp Ernst, Savaş Özkan, et al. Chaos challenge-combined (ct-mr) healthy abdominal organ segmentation. *Medical image analysis*, 69:101950, 2021.

Daniel S Kermany, Michael Goldbaum, Wenjia Cai, Carolina CS Valentim, Huiying Liang, Sally L Baxter, Alex McKeown, Ge Yang, Xiaokang Wu, Fangbing Yan, et al. Identifying medical diagnoses and treatable diseases by image-based deep learning. *cell*, 172(5):1122–1131, 2018.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Tao Li, Yingqi Gao, Kai Wang, Song Guo, Hanruo Liu, and Hong Kang. Diagnostic assessment of deep learning algorithms for diabetic retinopathy screening. *Information Sciences*, 501:511–522, 2019.

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.

Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In *International conference on machine learning*, pp. 6357–6368. PMLR, 2021.

Tianchi Liao, Lele Fu, Jialong Chen, Zhen Wang, Zibin Zheng, and Chuan Chen. A swiss army knife for heterogeneous federated learning: Flexible coupling via trace norm. *Advances in Neural Information Processing Systems*, 37:139886–139911, 2024.

Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. *Advances in neural information processing systems*, 33:2351–2363, 2020.

Jiawei Liu, Qiang Wang, Huijie Fan, Yinong Wang, Yandong Tang, and Liangqiong Qu. Residual denoising diffusion models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2773–2783. IEEE, 2024.

Yuxiang Lu, Suizhi Huang, Yuwen Yang, Shalayiding Sirejiding, Yue Ding, and Hongtao Lu. Fedhca 2: Towards hetero-client federated multi-task learning. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5599–5609. IEEE, 2024.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

Disha Makhija, Xing Han, Nhat Ho, and Joydeep Ghosh. Architecture agnostic federated learning for neural networks. In *International conference on machine learning*, pp. 14860–14870. PMLR, 2022.

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*, pp. 1273–1282. Pmlr, 2017.

Binh Nguyen, Minh-Duong Nguyen, Jinsun Park, Viet Pham, and Won-Joo Hwang. Federated domain generalization with data-free on-server matching gradient. In *International Conference on Learning Representations*, volume 2025, pp. 77488–77513, 2025.

Changdae Oh, Hyeji Hwang, Hee-young Lee, YongTaek Lim, Geunyoung Jung, Jiyoung Jung, Hosik Choi, and Kyungwoo Song. Blackvip: Black-box visual prompting for robust transfer learning. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24224–24235. IEEE, 2023.

Jaehoon Oh, Sangmook Kim, and Se-Young Yun. Fedbabu: Towards enhanced representation for federated image classification. *International Conference on Learning Representations*, 2021.

Krishna Pillutla, Kshitiz Malik, Abdel-Rahman Mohamed, Mike Rabbat, Maziar Sanjabi, and Lin Xiao. Federated learning with partial model personalization. In *International Conference on Machine Learning*, pp. 17716–17758. PMLR, 2022.

Sashank Reddi, Zachary Burr Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečný, Sanjiv Kumar, and Brendan McMahan. Adaptive federated optimization. In *International Conference on Learning Representations*, 2021.

Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletari, Holger R Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N Galtier, Bennett A Landman, Klaus Maier-Hein, et al. The future of digital health with federated learning. *NPJ digital medicine*, 3(1):119, 2020.

Christophe Roux, Max Zimmer, and Sebastian Pokutta. On the byzantine-resilience of distillation-based federated learning. In *International Conference on Learning Representations*, volume 2025, pp. 43105–43130, 2025.

Mehdi Setayesh, Mahdi Beitollahi, Yasser Khalil, and Hongliang Li. Toward enhancing representation learning in federated multi-task settings. In *International Conference on Learning Representations*, volume 2026, pp. 7961–7990, 2026.

Changlong Shi, He Zhao, Bingjie Zhang, Mingyuan Zhou, Dandan Guo, and Yi Chang. Fedawa: Adaptive optimization of aggregation weights in federated learning using client vectors. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 30651–30660. IEEE, 2025.

George Shih, Carol C Wu, Safwan S Halabi, Marc D Kohli, Luciano M Prevedello, Tessa S Cook, Arjun Sharma, Judith K Amorosa, Veronica Arteaga, Maya Galperin-Aizenberg, et al. Augmenting the national institutes of health chest radiograph dataset with expert annotations of possible pneumonia. *Radiology: Artificial Intelligence*, 1(1):e180041, 2019.

Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet S Talwalkar. Federated multi-task learning. *Advances in neural information processing systems*, 30, 2017.

Zhaobin Sun, Nannan Wu, Junjie Shi, Li Yu, Kwang-Ting Cheng, and Zengqiang Yan. Fedmlp: Federated multi-label medical image classification under task heterogeneity. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 394–404. Springer, 2024.

Saeid Asgari Taghanaki, Kumar Abhishek, and Ghassan Hamarneh. Improved inference via deep input transfer. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 819–827. Springer, 2019.

Ultralytics. Brain tumor detection dataset, 2023. URL `https://docs.ultralytics.com/datasets/detect/brain-tumor`.

Bastiaan S Veeling, Jasper Linmans, Jim Winkens, Taco Cohen, and Max Welling. Rotation equivariant cnns for digital pathology. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 210–218. Springer, 2018.

Fan Wu, Xinyu Yan, Jiabei Liu, and Wei Yang Bryan Lim. Fedram: Federated reweighting and aggregation for multi-task learning. *Advances in Neural Information Processing Systems*, 38: 102379–102406, 2026.

Luyuan Xie, Manqing Lin, Tianyu Luan, Cong Li, Yuejian Fang, Qingni Shen, and Zhonghai Wu. MH-pFLID: Model heterogeneous personalized federated learning via injection and distillation for medical data analysis. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 54561–54575. PMLR, 2024.

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*, pp. 35–44. Springer, 2025.

Xiyuan Yang, Wenke Huang, and Mang Ye. Fedas: Bridging inconsistency in personalized federated learning. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11986–11995. IEEE, 2024.

Wenwen Zhang, Zhenyu Tang, Hao Zhang, Shaohao Rui, Z Jane Wang, and Xiaosong Wang. Modality-agnostic federated learning with adaptive updates for heterogeneous medical image tasks. *IEEE Transactions on Medical Imaging*, 2026.

Hao Zheng, Zhigang Hu, Liu Yang, Meiguang Zheng, Aikun Xu, and Boyu Wang. Confree: Conflict-free client update aggregation for personalized federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 22875–22883, 2025.

A Appendix

A.1 Supplementary Ablation Analysis

Table 6: Ablation study of the perturbation factor ϵ on each clients.

ϵ	Seg. (Dice)			Cls. (Acc)			Det. (mAP@50)		Δ (%)
	C1	C2	C3	C4	C5	C6	C7	C8	
w/o	0.7577	0.3536	0.8243	0.9399	0.8041	0.8068	0.2797	0.4948	0.00
0.01	0.7621	0.3604	0.8408	0.9492	0.8156	0.8128	0.2865	0.5027	+1.46
0.05	**0.7702**	0.3740	0.8768	0.9641	0.8407	0.8302	0.3035	0.5194	+4.66
0.1	0.7695	**0.3786**	**0.8849**	**0.9709**	**0.8458**	0.8361	**0.3094**	**0.5267**	**+5.65**
0.5	0.7665	0.3695	0.8710	0.9617	0.8373	**0.8370**	0.2980	0.5162	+4.05
1.0	0.7569	0.3551	0.8326	0.9432	0.8097	0.8105	0.2789	0.4996	+0.44

Details of the Perturbation Factor ϵ Ablation Analysis. Table 6 reports the client-level sensitivity to the perturbation factor ϵ. Performance generally improves as ϵ increases to 0.1, which achieves the best overall gain of 5.65% and the highest results on six of eight clients. Although C1 and C6 slightly favor $\epsilon = 0.05$ and 0.5, respectively, larger perturbations reduce the overall benefit, confirming that a moderate budget better balances task adaptation and semantic preservation.

Table 7: Ablation study of the perturbation generator on each clients.

Gen.	Seg. (Dice)			Cls. (Acc)			Det. (mAP@50)		Δ (%)
	C1	C2	C3	C4	C5	C6	C7	C8	
w/o	0.7577	0.3536	0.8243	0.9399	0.8041	0.8068	0.2797	0.4948	0.00
NAFNet	**0.7701**	0.3764	0.8795	0.9681	0.8441	0.8338	0.3060	0.5232	+5.16
Pix2pix	0.7662	0.3720	0.8708	0.9627	0.8376	0.8279	0.2984	0.5164	+4.03
RDDM	0.7686	**0.3792**	0.8807	0.9674	0.8419	0.8328	0.3074	0.5228	+5.24
Tiramisu	0.7695	0.3786	**0.8849**	**0.9709**	**0.8458**	**0.8361**	**0.3094**	**0.5267**	**+5.65**

Details of the Perturbation Generator Ablation Analysis. Table 7 shows that all generator architectures improve every client over local training, confirming that TGPL can effectively parameterize task-beneficial perturbations with different architectures. Tiramisu achieves the best overall gain of 5.65% and the highest performance on six of eight clients, while NAFNet and RDDM perform best on C1 and C2, respectively. These results demonstrate the architectural flexibility of TGPL and support Tiramisu as the default generator.

Table 8: Ablation study of the perturbation optimizer on each client.

Opt.	Seg. (Dice)			Cls. (Acc)			Det. (mAP@50)		Δ (%)
	C1	C2	C3	C4	C5	C6	C7	C8	
w/o	0.7577	0.3536	0.8243	0.9399	0.8041	0.8068	0.2797	0.4948	0.00
Adam	0.7695	**0.3786**	**0.8849**	**0.9709**	0.8458	**0.8361**	**0.3094**	**0.5267**	**+5.65**
PGD	0.7688	0.3769	0.8817	0.9688	**0.8465**	0.8334	0.3058	0.5239	+5.23
MI-FGSM	**0.7699**	0.3748	0.8792	0.9671	0.8427	0.8349	0.3036	0.5218	+4.93

Details of the Perturbation Optimization Algorithm Ablation Analysis. Table 8 shows that Adam, PGD, and MI-FGSM consistently improve all clients over local training, confirming the effectiveness of task-guided perturbation optimization. Adam achieves the best overall gain of 5.65% and the highest performance on six of eight clients, while MI-FGSM and PGD perform slightly better on C1 and C5, respectively. These results demonstrate that TGPL is robust to different optimization algorithms and support Adam as the default choice.

Details of the Generator Loss Weight λ_1 Ablation Analysis. Table 9 shows that IPL-FL is relatively insensitive to the generator reconstruction weight λ_1. Different clients exhibit different preferences because the balance between structural consistency and pixel-level reconstruction affects heterogeneous tasks differently. For example, several segmentation clients favor smaller weights,

Table 9: Ablation study of the generator loss weight λ_1 on each client.

λ_1	Seg. (Dice)			Cls. (Acc)			Det. (mAP@50)		Δ (%)
	C1	C2	C3	C4	C5	C6	C7	C8	
$0.25\times$	0.7704	0.3722	**0.8858**	0.9678	0.8415	0.8345	0.3052	0.5226	+5.02
$0.5\times$	**0.7707**	0.3756	0.8853	0.9698	0.8438	0.8356	0.3070	0.5250	+5.36
$1\times$	0.7695	0.3786	0.8849	**0.9709**	0.8458	**0.8361**	0.3094	**0.5267**	**+5.65**
$2\times$	0.7685	0.3795	0.8825	0.9705	0.8466	0.8350	0.3100	0.5256	+5.61
$4\times$	0.7662	**0.3801**	0.8790	0.9697	**0.8469**	0.8318	**0.3103**	0.5218	+5.41

whereas some classification and detection clients benefit slightly from stronger pixel-level constraints. Despite these client-specific variations, $\lambda_1 = 1\times$ provides the best overall balance and achieves the highest average gain of 5.65%.

Table 10: Ablation study of the perturbation optimization steps P on each client.

P	Seg. (Dice)			Cls. (Acc)			Det. (mAP@50)		Δ (%)
	C1	C2	C3	C4	C5	C6	C7	C8	
20	0.7638	0.3625	0.8580	0.9650	0.8280	0.8220	0.2910	0.5170	+2.93
50	0.7670	0.3720	0.8750	0.9690	0.8390	0.8310	0.3015	0.5230	+4.56
100	0.7695	0.3786	0.8849	0.9709	0.8458	0.8361	0.3094	0.5267	**+5.65**
150	**0.7711**	0.3756	**0.8861**	0.9692	**0.8468**	0.8336	**0.3104**	0.5245	+5.53
200	0.7681	**0.3800**	0.8812	**0.9720**	0.8435	**0.8372**	0.3061	**0.5272**	+5.48

Details of the Perturbation Optimization Steps P Ablation Analysis. Table 10 shows that increasing P from 20 to 100 substantially improves the quality of task-guided perturbations. The improvement is client dependent, reflecting different optimization difficulties across tasks and modalities. After $P = 100$, the overall performance becomes saturated. Although individual clients may fluctuate slightly upward or downward at 150 or 200 steps, the average gain decreases marginally to 5.53% and 5.48%, respectively. This suggests that additional optimization beyond 100 steps provides limited benefit while increasing computational cost.

Table 11: Ablation study of the SVD energy threshold τ on each client.

τ	Seg. (Dice)			Cls. (Acc)			Det. (mAP@50)		Δ (%)
	C1	C2	C3	C4	C5	C6	C7	C8	
70%	0.7630	0.3660	0.8620	0.9665	0.8320	0.8290	0.2940	0.5200	+3.50
80%	0.7678	0.3728	0.8785	0.9690	0.8395	0.8330	0.3020	0.5240	+4.74
90%	**0.7695**	0.3786	**0.8849**	0.9709	0.8458	**0.8361**	**0.3094**	0.5267	**+5.65**
95%	0.7683	**0.3801**	0.8826	0.9720	0.8470	0.8338	0.3073	**0.5274**	+5.56
99%	0.7655	0.3760	0.8760	**0.9730**	**0.8490**	0.8305	0.3040	0.5220	+4.98

Details of the SVD Energy Threshold τ Ablation Analysis. Table 11 evaluates how much singular-value energy should be retained in the candidate public subspace. A small τ removes useful shared directions and leads to clear performance degradation, particularly for clients that require richer cross-client information. Increasing τ to 90% provides the best overall performance. Further increasing the threshold benefits several clients but degrades others, indicating that weak singular directions contain a mixture of useful and client-specific variations rather than uniformly transferable information. Therefore, $\tau = 90\%$ provides a favorable balance between shared-structure coverage and selective knowledge sharing.

Details of the RAPA Maximum Rank $q_{\max}$ Ablation Analysis. Table 12 shows that an overly small $q_{\max}$ restricts the candidate public subspace and removes useful shared structure. Increasing $q_{\max}$ from 1 to 4 therefore produces substantial improvements across most clients. When the rank cap is further relaxed, the response becomes client dependent. Some clients slightly benefit from additional directions, whereas others deteriorate because these directions contain less compatible or more client-specific variation. Consequently, $q_{\max} = 4$ achieves the best overall gain of 5.65% and

Table 12: Ablation study of the maximum rank $q_{\max}$ on each client.

$q_{\max}$	Seg. (Dice)			Cls. (Acc)			Det. (mAP@50)		Δ (%)
	C1	C2	C3	C4	C5	C6	C7	C8	
1	0.7618	0.3691	0.8534	0.9672	0.8258	0.8273	0.2892	0.5197	+3.13
2	0.7662	0.3756	0.8722	0.9697	0.8391	0.8338	0.2996	0.5235	+4.62
4	0.7695	0.3786	**0.8849**	0.9709	0.8458	**0.8361**	**0.3094**	0.5267	**+5.65**
6	**0.7702**	0.3776	0.8794	**0.9718**	**0.8479**	0.8329	0.3052	**0.5270**	+5.35
8	0.7681	**0.3788**	0.8728	0.9700	0.8437	0.8311	0.3026	0.5241	+4.95

provides a suitable balance between retaining shared structure and limiting unnecessary high-rank variation.

Table 13: Comparison under homogeneous task models.

Methods	C1	C2	C3	Δ (%)
	CHAOS-CT	CHAOS-MRI	ISIC	
	U-Net	U-Net	U-Net	
Local	0.7577	0.7375	0.8243	0.00
FedAvg	0.7985	0.7769	0.8678	+5.33
Ditto	0.8010	0.7796	0.8704	+5.67
ConFREE	0.8065	0.7869	0.8795	+6.61
FedMerge	0.8081	0.7856	0.8812	+6.69
IPL-FL (Ours)	**0.8135**	**0.7930**	**0.8867**	**+7.49**

Table 14: Comparison under heterogeneous task models.

Methods	C2	C5	C8	Δ (%)
	CHAOS-MRI	DDR	BRAIN	
	U-Net	Swin-T	YOLOv8	
Local	0.7375	0.8041	0.4948	0.00
FedHeNN	0.7308	0.7964	0.4876	-1.11
MH-pFLID	0.7261	0.7905	0.4841	-1.80
FedMuscle	0.7422	0.8086	0.4976	+0.59
IPL-FL (Ours)	**0.7618**	**0.8335**	**0.5147**	**+3.66**

A.2 Performance Analysis under Different Client Settings

Performance Comparison with Fully Task-Homogeneous Clients. Table 13 evaluates the homogeneous segmentation setting, where all clients use U-Net models. The increased compatibility of tasks and model architectures enables all federated methods to obtain substantial gains. IPL-FL achieves the best performance on every client and the highest overall improvement of 7.49%, demonstrating that input-space knowledge sharing remains effective even when direct parameter aggregation is feasible.

Performance Comparison with Fully Task-Heterogeneous Clients. Table 14 considers a constrained collaboration setting with only three clients performing heterogeneous segmentation, classification, and detection tasks. FedHeNN and MH-pFLID suffer from negative transfer, while FedMuscle provides only a marginal gain of 0.59%. In contrast, IPL-FL achieves the best result on every client and a substantially larger overall improvement of 3.66%, highlighting its advantage under limited and heterogeneous collaboration.

A.3 Client Correlation Analysis

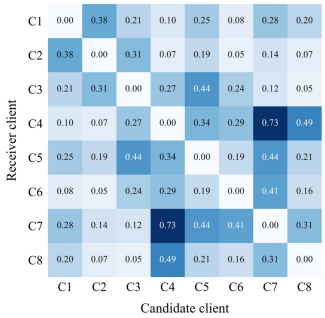

Figure 6: Client relation matrix constructed by RAPA across the eight heterogeneous clients. Larger values indicate stronger compatibility between the public generator updates of two clients.

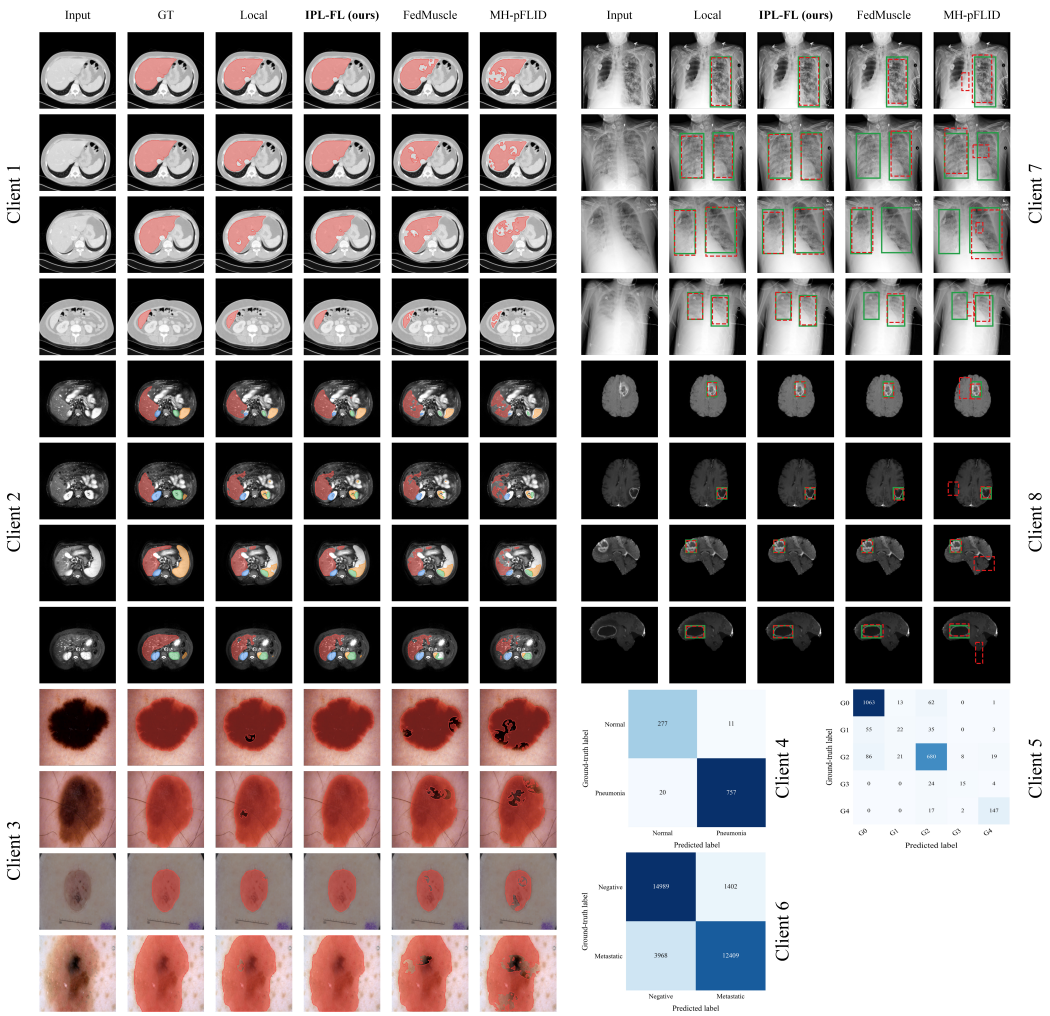

Figure 7: Supplementary qualitative results for individual clients across heterogeneous medical tasks. The figure compares segmentation predictions for C1–C3 and detection predictions for C7–C8 among Local, IPL-FL, FedMuscle, and MH-pFLID, and presents the confusion matrices of IPL-FL for the classification clients C4–C6.

Figure 6 visualizes the client relation matrix constructed by RAPA. Strong relations are not restricted to clients performing the same task, but are broadly observed across different task types. In particular, C4 and C7 exhibit the strongest relation of 0.73, although they perform classification and detection, respectively. Both clients involve thoracic imaging and related pathological patterns, making their public generator updates highly compatible and complementary. Other notable cross-task relations include C4–C8 (0.49), C5–C7 (0.44), C3–C5 (0.44), and C6–C7 (0.41). In contrast, the strongest within-task relations are relatively lower, including C1–C2 (0.38) for segmentation, C4–C5 (0.34) for classification, and C7–C8 (0.31) for detection. These observations show that RAPA does not simply group clients according to task identity, but identifies compatible and complementary knowledge across heterogeneous tasks, thereby enabling effective cross-task information exchange in IPL-FL.

A.4 Supplementary Visualization Analysis for Individual Clients

Figure 7 provides qualitative comparisons across segmentation, classification, and detection tasks. For the segmentation clients C1–C3, IPL-FL produces predictions that more closely match the ground truth, with more complete target regions and fewer boundary errors or fragmented predic-

tions than Local, FedMuscle, and MH-pFLID. Similar improvements are observed for the detection clients C7–C8, where the bounding boxes predicted by IPL-FL align more closely with the target lesions while reducing missed or spurious detections. For classification, the confusion matrices of C4 and C6 exhibit strong diagonal concentration, indicating reliable recognition of the major classes. However, C5 on DDR remains affected by its long-tailed class distribution. In particular, the minority grades G1 and G3 contain substantially fewer samples and are more frequently confused with neighboring grades, including G1 with G0/G2 and G3 with G2/G4. This result suggests that, although IPL-FL improves cross-task collaboration, severe class imbalance remains challenging for minority categories. Overall, these qualitative results further demonstrate that IPL-FL effectively exploits cross-task information to improve predictions across heterogeneous medical tasks.

A.5 Theoretical Properties of Task-Guided Perturbation Learning

TGPL is designed to transfer task supervision from heterogeneous local predictors to a common input-space carrier while keeping the task models fixed. This section establishes three properties that are directly related to this design. First, the perturbation optimization produces bounded input adaptations whose task loss is no worse than that of the original input. Second, gradients of the frozen predictor provide task-beneficial search directions rather than adversarial ones. Third, the task benefit obtained by sample-wise perturbation optimization can be preserved after the optimized inputs are parameterized by the perturbation generator, provided that the generator approximation error is sufficiently controlled.

Task supervision in the input space. For client k and sample $(x_{k,m}, y_{k,m})$, define the task objective induced by the frozen local predictor as

$$\phi_{k,m}(x) = \mathcal{L}_k^{\text{task}}\left(f_k(x; w_k^*), y_{k,m}\right). \tag{6}$$

TGPL optimizes $\phi_{k,m}(\text{clip}(x_{k,m} + \delta))$ under $\|\delta\|_\infty \le \epsilon$, while w_k^* remains unchanged. At differentiable points of the clipping operator, the perturbation gradient satisfies

$$\nabla_\delta \phi_{k,m} = J_{\text{clip}}^\top J_{f_k}^\top \nabla_{\hat{y}} \mathcal{L}_k^{\text{task}}(\hat{y}, y_{k,m}), \tag{7}$$

where $\hat{y} = f_k(\text{clip}(x_{k,m} + \delta); w_k^*)$. Thus, the perturbation update jointly depends on the task supervision and the input sensitivity of the frozen task model, while no gradient is used to modify w_k^*. This gives a precise interpretation of TGPL as transferring task supervision into input adaptation rather than continuing parameter-space optimization of the task model.

The perturbation constraint also guarantees bounded input deviation. Since coordinate-wise clipping is non-expansive, every feasible δ satisfies $\|\text{clip}(x_{k,m}+\delta) - x_{k,m}\|_\infty \le \epsilon$. Therefore, task supervision is expressed through a bounded local modification of the input rather than an unrestricted image transformation.

Task-beneficial property of perturbation optimization. Let $\delta_{k,m}^\star$ denote an exact solution of the constrained perturbation problem and let $x_{k,m}^\star = \text{clip}(x_{k,m} + \delta_{k,m}^\star)$. Because the zero perturbation is feasible, the optimal solution necessarily satisfies

$$\phi_{k,m}(x_{k,m}^\star) \le \phi_{k,m}(x_{k,m}). \tag{8}$$

Hence, the improvement margin $\gamma_{k,m}^\star = \phi_{k,m}(x_{k,m}) - \phi_{k,m}(x_{k,m}^\star)$ is non-negative.

The same property holds for the practical finite-step procedure. Let $\mathcal{C}_{k,m}^{(P)} = \{0, \delta_{k,m}^{(1)}, \ldots, \delta_{k,m}^{(P)}\}$ contain the feasible projected perturbations whose task losses are actually evaluated during P optimization steps. If $\bar{\delta}_{k,m}^{(P)}$ is the candidate with the smallest observed loss and $\bar{x}_{k,m}^{(P)} = \text{clip}(x_{k,m} + \bar{\delta}_{k,m}^{(P)})$, then

$$\phi_{k,m}\left(\bar{x}_{k,m}^{(P)}\right) \le \phi_{k,m}(x_{k,m}). \tag{9}$$

This follows immediately from $0 \in \mathcal{C}_{k,m}^{(P)}$. Therefore, the best-so-far task loss is monotonically non-increasing as more perturbation candidates are evaluated. Importantly, this guarantee does not require Adam, PGD, or another optimizer to reach the global optimum. The optimizer determines the candidate trajectory, whereas retaining the clean input and the lowest-loss candidate provides the non-degradation safeguard.

Why the perturbation gradient is task-beneficial. Equation (9) guarantees that TGPL does not retain a perturbation worse than the original input, but it does not explain why gradient optimization can actively discover a better one. Around an input x, let $g = \nabla_x \phi(x)$. The first-order approximation is $\phi(x+\delta) \approx \phi(x) + g^\top \delta$. Under $\|\delta\|_\infty \le \epsilon$, Hölder duality gives $g^\top \delta \ge -\epsilon \|g\|_1$, with equality attained at $\delta = -\epsilon \operatorname{sign}(g)$. Thus, the first-order TGPL direction maximally decreases the local linearization of the task objective within the admissible ℓ_∞ neighborhood.

A stronger local statement follows if ϕ is β-smooth. For $\delta_\alpha = -\alpha \operatorname{sign}(g)$ and a step that does not activate clipping,

$$\phi(x+\delta_\alpha) - \phi(x) \le -\alpha \|g\|_1 + \frac{\beta}{2}\alpha^2 \|\operatorname{sign}(g)\|_2^2. \tag{10}$$

Therefore, a sufficiently small positive α yields a strict local decrease whenever $g \neq 0$. This analysis explains why the frozen task model provides a meaningful task-guided search direction. It also distinguishes TGPL from adversarial perturbation generation, which follows the opposite direction to increase the task objective.

Preservation of task benefit by generator learning. Sample-wise perturbation optimization requires task supervision and iterative optimization, so TGPL subsequently parameterizes the optimized inputs using $G_k(\cdot;\theta_k)$. Let $x^*_{k,m}$ denote the optimized target stored in $\mathcal{D}_p$ and define $\gamma_{k,m} = \phi_{k,m}(x_{k,m}) - \phi_{k,m}(x^*_{k,m}) \ge 0$. Let $\hat{x}_{k,m} = G_k(x_{k,m};\theta_k)$.

Assume that $\phi_{k,m}$ is locally $L_{k,m}$-Lipschitz with respect to the ℓ_1 norm around $x^*_{k,m}$ and $\hat{x}_{k,m}$. Then

$$\phi_{k,m}(\hat{x}_{k,m}) - \phi_{k,m}(x_{k,m}) \le -\gamma_{k,m} + L_{k,m} \left\|\hat{x}_{k,m} - x^*_{k,m}\right\|_1. \tag{11}$$

Indeed, adding and subtracting $\phi_{k,m}(x^*_{k,m})$ separates the perturbation gain from the generator approximation error, and the latter is bounded by Lipschitz continuity. Equation (11) shows that generator learning preserves the perturbation benefit whenever $L_{k,m}\|\hat{x}_{k,m} - x^*_{k,m}\|_1 < \gamma_{k,m}$.

This condition is directly connected to the generator objective used in IPL-FL. Let $\Gamma_k = \frac{1}{N_k}\sum_m \gamma_{k,m}$ and assume $L_{k,m} \le L_k$ for all local samples. Since $1 - \mathrm{SSIM}(\hat{x}, x^*) \ge 0$, the generator objective satisfies $\mathcal{L}^G_k \ge \lambda_1 \bar{e}_k$, where $\bar{e}_k = \frac{1}{N_k}\sum_m \|\hat{x}_{k,m} - x^*_{k,m}\|_1$. Averaging Eq. (11) gives

$$\mathcal{R}^G_k - \mathcal{R}^{\text{clean}}_k \le -\Gamma_k + \frac{L_k}{\lambda_1}\mathcal{L}^G_k. \tag{12}$$

Hence, $\mathcal{L}^G_k < \lambda_1 \Gamma_k / L_k$ is a sufficient condition for the generated inputs to retain a positive empirical task benefit.

The same decomposition also clarifies the generalization requirement at inference time. If the expected generator error relative to the conceptual task-guided target on unseen samples is bounded by η_k, then the population risks satisfy

$$\mathcal{R}^{G,\text{pop}}_k - \mathcal{R}^{\text{clean,pop}}_k \le -\Gamma^{\text{pop}}_k + L_k \eta_k. \tag{13}$$

This is a conditional generalization statement rather than an unconditional guarantee, because η_k depends on generator capacity, optimization, and local data coverage.

Together, these results establish the role of TGPL in IPL-FL. TGPL converts supervision from heterogeneous frozen task models into bounded and task-beneficial input adaptations, and generator learning provides a parameterized carrier for these adaptations. TGPL itself does not assume that every encoded component is beneficial to every other client. The resulting generator updates are therefore treated as candidate shareable knowledge, whose cross-client compatibility is further determined by RAPA.

A.6 Optimality of Generator Update Low-Rank Decomposition

TGPL makes generator-level communication possible across heterogeneous task models, but the resulting generator updates may still contain both cross-client common variation and client-specific variation. RAPA therefore does not directly aggregate the complete update. It first extracts a compact common subspace in which potentially shareable information can be analyzed.

Reference-relative update representation. At communication round r, let θ_k^r denote the post-local generator parameters received by the server. RAPA defines $\mu_k^r = \mathrm{vec}(\theta_k^r - \theta_{\mathrm{ref}}^r)$ and stacks them as $U^r = [\mu_1^r, \ldots, \mu_N^r] \in \mathbb{R}^{D\times N}$. The reference removes a common coordinate offset without changing pairwise client differences, since $\mu_i^r - \mu_j^r = \mathrm{vec}(\theta_i^r - \theta_j^r)$. Therefore, the subsequent decomposition describes relative local adaptation rather than absolute generator parameter values.

Let $\bar{U}^r = U^r/s^r$ denote the normalized update matrix for the common positive scale s^r used by RAPA. Its singular value decomposition is

$$\bar{U}^r = A^r \Sigma^r (V^r)^\top, \tag{14}$$

with singular values $\sigma_1^r \geq \cdots \geq \sigma_\rho^r > 0$. For a rank q, let Q^r contain the first q left singular vectors and define $Z^r = (Q^r)^\top \bar{U}^r = [z_1^r, \ldots, z_N^r]$.

Optimality of the shared subspace. The decomposition used by RAPA can be written as

$$\min_{Q,Z} \left\|\bar{U}^r - QZ\right\|_F^2 \quad \text{s.t.} \quad Q^\top Q = I_q. \tag{15}$$

For a fixed Q, differentiating with respect to Z gives the optimal coordinates $Z = Q^\top \bar{U}^r$. Substituting this result into Eq. (15) gives $\|\bar{U}^r\|_F^2 - \|Q^\top \bar{U}^r\|_F^2$. Hence, minimizing reconstruction error is equivalent to maximizing the collective update energy projected onto the common q-dimensional subspace.

By the variational characterization of singular vectors, this maximum is achieved by the first q left singular vectors. Equivalently,

$$\min_{\mathrm{rank}(X)\leq q} \left\|\bar{U}^r - X\right\|_F^2 = \sum_{i=q+1}^{\rho} (\sigma_i^r)^2. \tag{16}$$

Thus, among all rank-q linear representations, the truncated SVD used by RAPA minimizes the total reconstruction error of the client generator updates and captures the maximum possible shared-subspace energy.

Define $U_{\mathrm{pub}}^r = s^r Q^r Z^r$ and $E^r = U^r - U_{\mathrm{pub}}^r$. Once the residual is retained, the decomposition is exact:

$$\boxed{U^r = s^r Q^r Z^r + E^r.} \tag{17}$$

Orthogonality and energy decomposition. Because $Z^r = (Q^r)^\top \bar{U}^r$ and $(Q^r)^\top Q^r = I_q$, the residual satisfies $(Q^r)^\top E^r = 0$. Consequently, the candidate public component and residual are orthogonal under the Frobenius inner product:

$$\langle s^r Q^r Z^r, E^r \rangle_F = 0. \tag{18}$$

The total update energy therefore decomposes as

$$\|U^r\|_F^2 = \|s^r Q^r Z^r\|_F^2 + \|E^r\|_F^2. \tag{19}$$

For an individual client, $\mu_k^r = s^r Q^r z_k^r + e_k^r$ with $(Q^r)^\top e_k^r = 0$. Hence, RAPA separates each update into a component explained by the common subspace and a residual component that cannot be represented by that subspace.

The fraction of update energy explained by rank q is

$$\eta_q^r = \frac{\sum_{i=1}^q (\sigma_i^r)^2}{\sum_{i=1}^\rho (\sigma_i^r)^2}. \tag{20}$$

Accordingly, the target rank is the smallest q satisfying $\eta_q^r \geq 0.9$, subject to the maximum rank used in IPL-FL. If the rank cap becomes active before the 90% threshold is reached, the method intentionally favors a compact public subspace rather than forcing additional heterogeneous variation into the shared component.

This behavior is desirable under task heterogeneity. When client generator updates are strongly correlated, their singular spectrum is concentrated and a small q explains most of the update energy. When client updates are substantially different, the spectrum becomes less concentrated and a larger fraction of variation remains in E^r. Thus, the decomposition does not assume that all generator variation should be shared.

Geometry preservation in the public coordinates. The subsequent relation modeling is performed on Z^r, so it is important to establish that these low-dimensional coordinates preserve the relevant geometry of the candidate public updates. For client k, define $u^r_{k,\mathrm{pub}} = s^r Q^r z^r_k$. Since $(Q^r)^\top Q^r = I_q$,

$$(u^r_{i,\mathrm{pub}})^\top u^r_{j,\mathrm{pub}} = (s^r)^2 (z^r_i)^\top z^r_j, \qquad \|u^r_{i,\mathrm{pub}} - u^r_{j,\mathrm{pub}}\|_2 = s^r \|z^r_i - z^r_j\|_2.$$

Therefore, whenever both public components are non-zero,

$$\cos\left(u^r_{i,\mathrm{pub}}, u^r_{j,\mathrm{pub}}\right) = \cos\left(z^r_i, z^r_j\right). \tag{21}$$

Thus, Z^r is not an arbitrary compressed representation. It is the orthogonal coordinate representation of the high-dimensional candidate public updates and preserves their directional geometry exactly.

The decomposition should nevertheless be interpreted conservatively. Low rank alone cannot guarantee that every retained direction is beneficial to every client, nor does the residual necessarily contain only task-specific information. The mathematical result is that $s^r Q^r Z^r$ is the optimal jointly representable rank-constrained component, while E^r contains the remaining unexplained variation. RAPA therefore treats the former as *candidate public knowledge* and performs an additional relation-aware selection before knowledge is exchanged.

A.7 Theoretical Analysis of Relation-Aware Knowledge Selection

The low-rank decomposition determines where potentially shareable generator information can be compared, but it does not determine which clients should share knowledge. RAPA addresses this remaining knowledge-selection problem using two complementary criteria. A useful collaborator should be directionally compatible with the target client and should also provide information that is not already represented by the selected collaborators.

Directional compatibility. For target client k and candidate client j, RAPA defines $a^r_{kj} = \max(\cos(z^r_k, z^r_j), 0)$, with the score set to zero when either coordinate vector has zero norm. By Eq. (21), this is exactly the cosine relation between their high-dimensional candidate public updates.

The inner product can be written as $(z^r_k)^\top z^r_j = \|z^r_k\|_2 \|z^r_j\|_2 \cos(z^r_k, z^r_j)$. Therefore, if $\cos(z^r_k, z^r_j) \leq 0$, the candidate direction has a non-positive projection onto the target public direction. Setting $a^r_{kj} = 0$ excludes such directly conflicting directions from relation construction. Positive alignment does not by itself guarantee beneficial transfer, but it provides a necessary compatibility filter before any information from client j is considered for client k.

Conditional information as complementarity. Directional similarity alone cannot distinguish complementary collaborators from redundant ones. To quantify the additional variation provided by a candidate, RAPA constructs a positive semidefinite Gram matrix from the public coordinates: $K^r_{ij} = (z^r_i)^\top z^r_j / q$. We interpret K^r as the covariance surrogate of a zero-mean Gaussian latent vector $\xi^r = [\xi^r_1, \ldots, \xi^r_N]^\top$.

For a selected neighbor set $\mathcal{S}_k$, consider noisy observations $r_{\mathcal{S}_k} = \xi_{\mathcal{S}_k} + \eta_{\mathcal{S}_k}$ with $\eta_{\mathcal{S}_k} \sim \mathcal{N}(0, \rho I)$. The conditional variance of candidate j is then

$$v^r_{j|\mathcal{S}_k} = K^r_{jj} - K^r_{j\mathcal{S}_k} \left(K^r_{\mathcal{S}_k\mathcal{S}_k} + \rho I\right)^{-1} K^r_{\mathcal{S}_k j}. \tag{22}$$

This is the regularized Schur complement used by RAPA. It is non-negative because it is a conditional variance under the Gaussian surrogate.

The quantity has a direct redundancy interpretation. If client j is largely predictable from the already selected clients, then $v^r_{j|\mathcal{S}_k}$ is small. If client j contains variation that cannot be explained by the current neighbor set, the conditional variance remains large.

To convert this residual variation into information gain, let $o^r_j = \xi^r_j + \varepsilon_j$ with $\varepsilon_j \sim \mathcal{N}(0, \sigma^2)$ independent of all other variables. For Gaussian variables,

$$\begin{aligned} I\left(\xi^r_j; o^r_j \mid r_{\mathcal{S}_k}\right) &= h\left(o^r_j \mid r_{\mathcal{S}_k}\right) - h\left(o^r_j \mid \xi^r_j, r_{\mathcal{S}_k}\right) \\ &= \frac{1}{2} \log\left(1 + \frac{v^r_{j|\mathcal{S}_k}}{\sigma^2}\right). \end{aligned} \tag{23}$$

Equation (23) therefore gives a precise information-theoretic interpretation of the conditional information gain used in RAPA.

Redundancy decreases after neighbor selection. The conditional variance also explains why RAPA updates the information gain after each greedy selection. Suppose $\mathcal{S}_1 \subseteq \mathcal{S}_2$. Under the Gaussian surrogate, $v^r_{j|\mathcal{S}}$ is the minimum mean-squared prediction error of ξ^r_j from the noisy observations of the selected clients. Enlarging the observation set cannot increase this minimum error. Therefore,

$$v^r_{j|\mathcal{S}_2} \le v^r_{j|\mathcal{S}_1}, \qquad g^r_{j|\mathcal{S}_2} \le g^r_{j|\mathcal{S}_1}. \tag{24}$$

Thus, once one informative client has been selected, other clients carrying similar public variation naturally become less valuable. This mechanism prevents relation construction from repeatedly selecting highly redundant collaborators.

Compatibility–complementarity relation. RAPA combines the two criteria as $R^r_{kj} = a^r_{kj} g^r_{j|\mathcal{S}_k}$. Both factors are non-negative. Therefore, a high relation score requires both positive directional compatibility and non-redundant conditional information. If the directions conflict, then $a^r_{kj} = 0$ regardless of the information magnitude. If a candidate is directionally aligned but almost completely explained by the selected neighbors, then $g^r_{j|\mathcal{S}_k}$ becomes small.

The product therefore implements the two stages of knowledge selection required by IPL-FL:

$$\text{relation} = \text{compatibility} \times \text{complementarity}.$$

After greedy neighbor selection, RAPA symmetrizes the non-negative relation scores to obtain the symmetric relation matrix used by graph aggregation.

This analysis also clarifies why RAPA is necessary after low-rank decomposition. SVD identifies the dominant jointly representable generator variation, but cannot determine client-specific transfer relationships. Directional consistency excludes directly conflicting public directions, while conditional information gain suppresses redundant knowledge. Together, they transform the candidate public subspace into a client-specific relation structure for personalized aggregation.

A.8 Graph-Regularized Personalized Aggregation

After relation-aware selection, RAPA obtains a non-negative symmetric relation matrix R^r. The remaining question is how these pairwise relations should be converted into personalized client updates without collapsing all clients into one common generator. We show that the aggregation rule used by RAPA is the unique solution of a graph-regularized optimization problem.

Graph-regularized objective. Let $D^r = \mathrm{diag}(R^r \mathbf{1})$ and $L^r = D^r - R^r$ be the graph Laplacian. Let $Y^r = [y^r_1, \dots, y^r_N]$ denote the personalized public coordinates after aggregation. Consider

$$\min_{Y} \frac{1}{2} \|Y - Z^r\|_F^2 + \frac{\lambda}{2} \mathrm{Tr}\left(Y L^r Y^\top\right). \tag{25}$$

The first term preserves each client local public update, while the second term encourages only related clients to approach one another. Since

$$\mathrm{Tr}(Y L^r Y^\top) = \frac{1}{2} \sum_{i,j} R^r_{ij} \|y_i - y_j\|_2^2,$$

strong edges impose stronger consistency, whereas unrelated clients are not directly coupled.

Differentiating Eq. (25) with respect to Y gives $Y - Z^r + \lambda Y L^r = 0$. Therefore,

$$\boxed{Y^r = Z^r \left(I + \lambda L^r\right)^{-1}.} \tag{26}$$

This is exactly the graph smoothing rule used by RAPA. Hence, the aggregation is not an arbitrary interpolation. It is the closed-form minimizer balancing fidelity to each local public update and consistency with selected related clients.

Existence and uniqueness. For any vector v,

$$v^\top L^r v = \frac{1}{2}\sum_{i,j} R^r_{ij}(v_i - v_j)^2 \geq 0.$$

Thus, $L^r \succeq 0$. For $\lambda \geq 0$, $I + \lambda L^r \succ 0$, so the inverse in Eq. (26) always exists. Consequently, Eq. (25) is strictly convex and has a unique minimizer.

This property is important for RAPA because the relation graph may contain multiple connected components or isolated clients. The personalized solution remains well-defined in all such cases.

Exact preservation of the residual component. RAPA reconstructs the personalized update matrix as

$$\widehat{U}^r = s^r Q^r Y^r + E^r. \tag{27}$$

Combining Eq. (17) and Eq. (27) gives

$$\widehat{U}^r - U^r = s^r Q^r \left(Y^r - Z^r\right). \tag{28}$$

Therefore, all cross-client modification occurs inside the candidate public subspace. The residual E^r is exactly invariant under aggregation.

For client k, $\widehat{\mu}^r_k = s^r Q^r y^r_k + e^r_k$, so the original residual e^r_k is directly restored after graph smoothing. This establishes mathematically the preservation mechanism used by RAPA. The method does not merely regularize all generator parameters less aggressively. It completely excludes the residual component from cross-client smoothing.

Mean preservation and the server reference. A graph Laplacian satisfies $L^r\mathbf{1} = 0$. Therefore, $(I + \lambda L^r)^{-1}\mathbf{1} = \mathbf{1}$, which implies

$$Y^r\mathbf{1} = Z^r\mathbf{1}. \tag{29}$$

Thus, graph aggregation redistributes the candidate public information across related clients without changing its global mean. Together with the unchanged residual,

$$\widehat{U}^r\mathbf{1} = U^r\mathbf{1}. \tag{30}$$

This property provides a direct justification for updating the next server reference by averaging the personalized generators. The relation-aware step changes how information is assigned to individual clients, but does not introduce an artificial shift in the mean generator update.

Behavior for weakly related and isolated clients. If client k has no selected relation to any other client, its row and column in L^r are zero. Equation (26) then gives $y^r_k = z^r_k$. Hence, an isolated client retains its original public coordinate in addition to its unchanged residual.

More generally, if the relation graph has several disconnected components, Eq. (25) decomposes across those components. Knowledge can therefore propagate only among clients connected by the relation-selection stage. This prevents RAPA from forcing unrelated heterogeneous tasks toward a single shared solution.

Spectral interpretation. Let $L^r = V\Lambda V^\top$, where $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_N)$ with $\lambda_i \geq 0$. Then

$$(I + \lambda L^r)^{-1} = V\,\mathrm{diag}\left(\frac{1}{1 + \lambda\lambda_i}\right)V^\top. \tag{31}$$

Graph modes with large Laplacian eigenvalues correspond to strong disagreement across relation edges and are more strongly attenuated, while modes in the null space of L^r are preserved exactly. RAPA can therefore be interpreted as a low-pass filter acting only on relation-supported public variation, while the client-specific residual remains untouched.

A.9 Stability and Convergence Properties of RAPA

The complete IPL-FL procedure contains non-convex local generator optimization and an adaptive relation graph that changes across communication rounds. Therefore, claiming unconditional convergence of the entire deep federated system would require assumptions that are substantially stronger than those used by the actual method. Instead, we establish stability and convergence properties of the RAPA aggregation operator itself, which is the component introduced by our framework at the server.

Non-expansiveness of graph aggregation. For a fixed relation graph, define

$$M^r = (I + \lambda L^r)^{-1}$$

and the public aggregation operator $\mathcal{A}_{L^r}(Z) = ZM^r$. From Eq. (31), every eigenvalue of M^r lies in $(0, 1]$. Hence, $\|M^r\|_2 = 1$. For any two public-coordinate matrices Z_1 and Z_2,

$$\|\mathcal{A}_{L^r}(Z_1) - \mathcal{A}_{L^r}(Z_2)\|_F \leq \|Z_1 - Z_2\|_F. \tag{32}$$

Therefore, RAPA does not amplify perturbations in the public coordinates for a fixed graph.

Because multiplication by the orthonormal basis Q^r preserves Euclidean norm, the same property transfers to the reconstructed public generator updates up to the common scale s^r. Thus, the server aggregation itself does not introduce unstable amplification of client-update differences.

Stability to relation-graph perturbations. The relation graph is estimated from finite client updates and may vary slightly under optimization noise. Let L and $\widetilde{L}$ be two positive semidefinite Laplacians and define $M = (I + \lambda L)^{-1}$ and $\widetilde{M} = (I + \lambda \widetilde{L})^{-1}$. The resolvent identity gives

$$M - \widetilde{M} = \lambda M(\widetilde{L} - L)\widetilde{M}.$$

Since $\|M\|_2, \|\widetilde{M}\|_2 \leq 1$,

$$\|M - \widetilde{M}\|_2 \leq \lambda \|L - \widetilde{L}\|_2. \tag{33}$$

Consequently, for a fixed Z,

$$\|ZM - Z\widetilde{M}\|_F \leq \lambda \|Z\|_F \|L - \widetilde{L}\|_2.$$

Thus, small perturbations in the estimated relation graph produce proportionally bounded changes in the personalized public coordinates. The parameter λ therefore controls not only the strength of sharing, but also the sensitivity of the aggregation output to graph estimation errors.

Convergence under a fixed relation graph. To characterize the intrinsic convergence behavior of graph smoothing, consider repeatedly applying the RAPA public operator while holding L fixed and omitting new local training: $Z^{(t+1)} = Z^{(t)}M$. Then

$$Z^{(t)} = Z^{(0)} V \operatorname{diag}\left[\left(\frac{1}{1 + \lambda\lambda_i}\right)^t\right] V^\top. \tag{34}$$

For every $\lambda_i > 0$, the corresponding factor converges to zero, whereas the factors associated with $\lambda_i = 0$ remain one. Therefore,

$$\lim_{t\to\infty} Z^{(t)} = Z^{(0)} P_{\mathrm{null}(L)}, \tag{35}$$

where $P_{\mathrm{null}(L)}$ is the orthogonal projection onto the null space of the graph Laplacian.

For a connected graph, $P_{\mathrm{null}(L)} = \mathbf{1}\mathbf{1}^\top/N$, so the public coordinates converge to their common mean. For a graph with multiple connected components, convergence occurs independently within each component. Importantly, the residual E is not involved in this limiting process. Even if the public component approaches relation-supported consensus, client-specific residual information remains unchanged.

This result explains the personalization behavior of RAPA. Its smoothing dynamics act only on the information explicitly selected as public and only along relation-supported edges. They do not imply convergence of the complete generator parameters to a single global model.

Compatibility with convergent local dynamics. A sufficient condition can also be stated for the composition of local generator optimization and a fixed RAPA operator. Let $\mathcal{T}$ denote a stacked local-update mapping and suppose that, in a neighborhood of a fixed point, it is contractive:

$$\|\mathcal{T}(X) - \mathcal{T}(Y)\|_F \leq \kappa \|X - Y\|_F, \qquad 0 \leq \kappa < 1.$$

Since RAPA is non-expansive by Eq. (32),

$$\|\mathcal{A}_L(\mathcal{T}(X)) - \mathcal{A}_L(\mathcal{T}(Y))\|_F \leq \kappa \|X - Y\|_F.$$

Hence, under this sufficient local condition and a fixed graph, the composed update is also contractive and admits a unique local fixed point.

This statement does not assert that the non-convex neural-network training used in IPL-FL is globally contractive. Instead, it shows that the RAPA operator does not destroy contraction when such stability is provided by the local optimization dynamics.

Scope of the convergence analysis. The above results establish that each RAPA server step is well-posed, uniquely solvable, non-expansive for a fixed graph, stable to bounded graph perturbations, and convergent under repeated fixed-graph smoothing. These properties are sufficient to characterize the stability of the proposed aggregation mechanism.

They should be distinguished from an end-to-end convergence theorem for IPL-FL. In the complete framework, local generator optimization is non-convex, personalized generators change every round, and the relation graph is adaptively reconstructed from the resulting updates. A global convergence theorem would therefore require additional smoothness, stochastic-gradient, bounded-variance, and graph-evolution assumptions that are not essential to the design of IPL-FL. We consequently restrict our theoretical claims to the components introduced by TGPL and RAPA and empirically evaluate the complete federated optimization process in the experiments.