FedKeyMIS: Keyed Federated Multi-image Steganography for Selective Extraction and Crosstalk Reduction

Longshun Hu

School of Computer Science and Information Engineering, Hefei University of Technology, Hefei 230601, China
2023217309@mail.hfut.edu.cn

Abstract. Multi-image steganography embeds several secret images into one carrier, but most methods decode the entire hidden set together, exposing more information than a receiver needs and causing cross-secret interference. We propose FedKeyMIS, a keyed federated framework for selective extraction. Each secret-key pair is treated as a client-specific task: the server aggregates only a carrier-preserving backbone, while key-indexed adapters remain local and a query key activates only its target route. We introduce crosstalk and selective-exposure metrics, and train with a personalized federated objective that keeps route-specific gradients in local adapters. Experiments under a fixed protocol show improved secret fidelity, lower leakage, stronger key selectivity, and better steganalysis AUC than centralized or non-personalized baselines. Scaling from 2 to 8 routes degrades gradually rather than abruptly, clarifying practical communication limits for larger deployments.

Keywords: Image steganography, Federated learning, Personalized learning, Selective extraction, Multi-image hiding

1 Introduction

Deep image steganography hides full-size images using learned hiding and recovery networks [2, 16, 14, 13], and adversarial, invertible, or generative variants improve payload and perceptual quality [15, 10, 7]. Multi-image hiding packs several secrets into one carrier, but most systems still decode the full hidden set in one shot. A successful query may therefore reveal unnecessary content, and the recovered target may contain traces of non-target secrets.

Whole-set exposure and crosstalk are coupled: once the stego image is decoded, all hidden content may be revealed, and different secret paths compete for the same representation budget. Private-key multi-image steganography partly limits access by attaching keys to hidden content [6], but centralized training still keeps strong cross-task coupling.

We treat each secret-key pair as a client-specific task and train with personalized federated optimization [11, 5, 1, 9, 8, 3]. FedKeyMIS combines a shared hiding backbone, local keyed adapters, and selective routing. The server aggregates only the shared backbone; each client keeps its adapter, and a query key activates only the target route at inference time. Here federation is a route-decoupling mechanism, not a standalone security guarantee.

Our contributions are a keyed federated framework for one-query-one-secret extraction, crosstalk/selectivity metrics for multi-image hiding, and an evaluation covering fidelity, key selectivity, steganalysis AUC, communication cost, and scalability.

2 Related Work

2.1 Deep Image Steganography

Neural steganography has evolved from encoder-decoder pipelines [2, 16] to adversarial, generative, and invertible variants [15, 14, 13, 10, 7]. These methods optimize payload, fidelity, or robustness, but usually assume that the receiver decodes all hidden content together, leaving fine-grained access control and route interference underexplored.

Multi-image steganography embeds multiple secrets into a shared carrier [6, 4]. Private-key variants introduce route identifiers [6], yet centralized training still forces all recovery paths to share one optimization space. We instead personalize each secret-key route and make one-query-one-secret extraction the primary target.

2.2 Federated and Personalized Learning

Federated learning supports distributed optimization without centralizing raw data [11, 5], and personalized variants keep part of the model client-specific [1, 9, 8, 3, 12]. This matches selective multi-image steganography: secret-key pairs have distinct recovery objectives, while carrier-preserving knowledge can be shared. FedKeyMIS uses this split to reduce cross-task interference.

3 Method

3.1 Problem Definition and Overview

We consider a carrier image distribution $x \sim \mathcal{P}_X$ and K secret-image tasks. Task i is defined by a secret distribution $\mathcal{P}_{S_i}$, a query key k_i, and a client-specific parameter block ψ_i. In a centralized joint extractor, one query on the stego image returns all hidden secrets,

$$(\hat{s}_1, \dots, \hat{s}_K) = D_\theta(y). \tag{1}$$

In contrast, this paper studies a keyed extractor that should recover only the target image associated with the query key:

$$\hat{s}_i = D_{\phi,\psi_i}(y, k_i). \tag{2}$$

Therefore, the design goal has two aspects: one query should reveal only one target secret, and the recovered target should contain as little non-target evidence as possible.

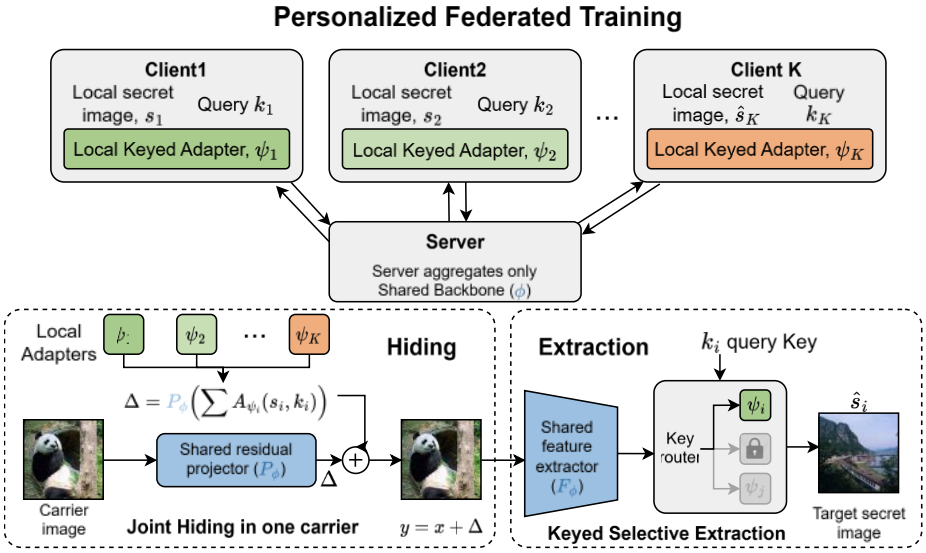

Fig. 1. Overview of FedKeyMIS. During training, each client owns one secret distribution, one key, and one local adapter. Only the shared backbone is aggregated by the server. During inference, multiple keyed residuals are composed into one carrier, while the receiver uses the query key to route the stego image to the target adapter.ϕ

Figure 1 shows the pipeline. The shared backbone learns carrier-preserving hiding and feature extraction, while local adapters specialize individual secret-key routes. Each client owns a secret distribution rather than one memorized image. During federated training, companion routes form the multi-route carrier but remain frozen; backpropagation updates only the shared backbone and the target adapter ψ_i.

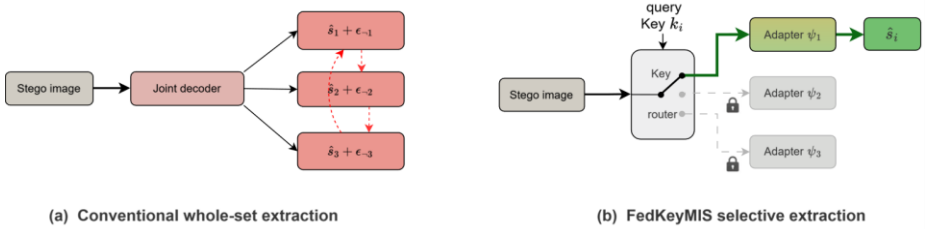

Fig. 2. Conceptual comparison between conventional multi-image steganography and FedKeyMIS. Conventional extraction decodes the whole hidden set and may introduce residual leakage among outputs, whereas FedKeyMIS routes the query through a key-indexed adapter and returns only the target secret.

Figure 2 contrasts joint extraction with keyed selective extraction, where the key acts as the route-selection primitive.

3.2 Keyed Latent Composition

FedKeyMIS uses a shared hiding backbone with client-specific keyed adapters. For client i, the local adapter transforms the secret image and its key into a keyed residual code:

$$h_i = A_{\psi_i}(s_i, k_i). \tag{3}$$

The sender composes multiple secret codes into a single perturbation through a shared residual projector:

$$\Delta = P_\phi\left(\sum_{i=1}^{K} h_i\right), \qquad y = \mathrm{clip}(x + \Delta). \tag{4}$$

The receiver extracts shared features from the stego image and then uses the query key to select the matching adapter:

$$z = F_\phi(y), \qquad \hat{s}_i = B_{\psi_i}(z, k_i). \tag{5}$$

This decomposition separates transferable behavior in P_ϕ and F_ϕ from route-specific mapping and decoding in A_{ψ_i} and B_{ψ_i}. Localized route parameters support a fixed adapter bank and allow new routes to be registered by training new adapters.

3.3 Personalized Federated Optimization

At communication round t, the server broadcasts the shared parameters ϕ^t to sampled clients. Client i receives ϕ^t, keeps its own adapter ψ_i^t, and updates the shared backbone together with its target route:

$$(\phi_i^{t+1}, \psi_i^{t+1}) = \mathrm{LocalUpdate}_i(\phi^t, \psi_i^t; D_i, \mathcal{R}_{-i}^t). \tag{6}$$

Here $\mathcal{R}_{-i}^t$ denotes the frozen companion routes used only to instantiate the multi-secret carrier during client i's forward pass. The server then aggregates only the shared parameters of the sampled subset,

$$\phi^{t+1} = \sum_{i \in \mathcal{C}_t} p_i^t \, \phi_i^{t+1}, \qquad \sum_{i \in \mathcal{C}_t} p_i^t = 1, \tag{7}$$

while each ψ_i remains client-specific.

The local objective combines target reconstruction, carrier fidelity, and route separation:

$$\mathcal{L}_i = \lambda_{\text{rec}}\|\hat{s}_i - s_i\|_1 + \lambda_{\text{stego}}\|y - x\|_2^2 + \lambda_{\text{route}}\max\left(0, m - \left\|A_{\psi_i}(s_i, k_i) - A_{\psi_i}(s_i, \tilde{k}_i)\right\|_2\right), \tag{8}$$

where $\tilde{k}_i$ denotes a mismatched key sampled from the public key space. The route-separation term encourages the valid keyed route to stay distinguishable from wrong-key routes.

Algorithm 1. Personalized keyed federated training in FedKeyMIS

Input: shared parameters phi^0; local adapters psi_i^0; local data D_i; communication rounds T. 1. For each communication round t = 0, ..., T-1, the server samples clients C_t and broadcasts phi^t. 2. Each sampled client builds one multi-route hiding sample using its target secret and frozen companion routes. 3. The client updates its shared copy and local adapter while stopping gradients on non-target routes. 4. The client sends the updated shared parameters to the server and keeps the local adapter private. 5. The server aggregates the sampled shared parameters to obtain phi^t+1. **Output:** shared backbone phi^T and keyed local adapters psi_i^T.

This strategy is mainly architectural: centralized models make every secret path compete over all parameters, whereas FedKeyMIS confines competition to the shared backbone and absorbs route-specific gradients locally.

3.4 System Protocol and Implementation Checklist

The federated procedure is a training-time task-decoupling protocol, not a complete security solution. We assume a static deployment: after training, the sender and authorized receiver hold a registered adapter bank, and the query key activates one target route at inference time.

Table 1 summarizes the module-level implementation; the focus is the split between shared and personalized components.

Table 1. Implementation details of FedKeyMIS. The model consists of a shared hiding/extraction trunk and lightweight route-specific personalization modules.

Module	Design choice	Configuration
Shared backbone phi	Hiding/extraction trunk	U-Net-like encoder-decoder with four stages and channel widths 64/128/256/512
Feature extractor F_phiphi	Shared latent representation	Four-stage CNN encoder with a 16x16 latent map for 256x256 inputs
Key encoder	Route conditioning	128-bit binary key mapped to a 128-d embedding by a 2-layer MLP and fused by feature concatenation
Local adapter A_psi_i	Personalized route module	Bottleneck residual adapters after each encoder stage, reduction ratio 4, about 0.8M parameters per client

Module	Design choice	Configuration
Decoder head B_psi_i	Secret reconstruction	Three-block route-specific upsampling decoder with skip fusion from the shared feature stream
Optimizer	Parameter update	AdamW with learning rate 1x10^-4 and weight decay 1x10^-4
Local training	Per-round update	Batch size 8 per client and 1 local epoch per communication round
Training budget	Communication schedule	100 communication rounds, 3 random seeds, and one NVIDIA RTX 4090 GPU
Parameter split	Shared vs. personalized	About 90% of parameters in shared trunk phi and 10% in local modules psi_i

3.5 Crosstalk, Selective Exposure, and Key Robustness

Definition (Crosstalk Index). For the target reconstruction $\hat{s}_i$ and a non-target secret s_j with $i \neq j$, let

$$L_{ij} = \mathbb{E}[\mathrm{sim}(\hat{s}_i, s_j)]. \tag{9}$$

Here $\mathrm{sim}(a, b)$ denotes cosine similarity between frozen evaluation features,

$$\mathrm{sim}(a, b) = \frac{\langle g(a), g(b) \rangle}{\|g(a)\|_2 \|g(b)\|_2}, \tag{10}$$

where $g(\cdot)$ is the penultimate-layer embedding of a fixed ImageNet-pretrained ResNet-50 encoder. The average crosstalk index is defined as

$$\mathrm{CTI} = \frac{1}{K(K-1)} \sum_{i \neq j} L_{ij}, \tag{11}$$

where a lower value indicates weaker non-target leakage.

Definition (Selective Exposure Ratio). Let $N_{\mathrm{exp}}(q)$ denote the number of distinct secret images exposed by one query outcome q, regardless of whether the query is authorized or not. The selective exposure ratio of that query is

$$\mathrm{SER}(q) = \frac{N_{\mathrm{exp}}(q)}{K}. \tag{12}$$

In a whole-set extractor, $\mathrm{SER} = 1$. Under matched-key selective extraction, an ideal one-key-one-secret extractor yields $\mathrm{SER} = 1/K$, while wrong-key or unseen-key queries should drive SER toward 0.

To examine whether the query key acts as a sharp selector rather than a weak auxiliary condition, we further perform a key-sensitivity analysis by corrupting the 128-bit query key with random bit flips. Figure 3 reports a simulation-based pilot estimate for three reconstruction metrics, anchored at the matched-key values and the unseen-key

floor used elsewhere in the paper. For each bit-flip ratio ρ, we flip $\lfloor 128\rho \rfloor$ bits uniformly at random and decode the same stego image with the corrupted key. As the corruption increases, PSNR and SSIM decrease rapidly while LPIPS rises toward the unseen-key regime: once the corruption reaches 10%–20%, the recovered image already loses most target fidelity, and by 30%–40% the three metrics are close to the unseen-key floor. This behavior supports the intended security property of FedKeyMIS: route identity is highly sensitive to key corruption and does not remain recoverable under approximate key matches.

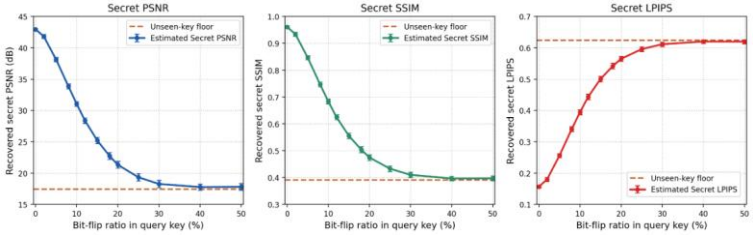

Fig. 3. Key-robustness analysis of FedKeyMIS under random corruption of the 128-bit query key. The three panels report secret-reconstruction PSNR, SSIM, and LPIPS as the bit-flip ratio increases. As the query key deviates from the registered route, PSNR and SSIM drop while LPIPS rises, and all three metrics approach the unseen-key floor at higher corruption levels, indicating that approximate key matches do not preserve route identity.

4 Experiments

This section presents the quantitative and qualitative results of FedKeyMIS. We also describe the experimental protocol and the selective-access evaluation procedure so that the reported numbers can be interpreted in a concrete setting.

4.1 Experimental Protocol, Baselines, and Metrics

Datasets and preprocessing. Tables 3–6 use one fixed protocol. Unless stated otherwise, the main comparison is the 4-secrets / 4-clients setting, matching the FedKeyMIS row in Table 6.

Table 2. Default protocol. Methods share the same carrier/secret organization, split rule, and query protocol.

Protocol item	Current manuscript entry
Carrier dataset / version	MS-COCO 2017; 113,287 training images, 5,000 validation images, and 5,000 test images
Secret dataset(s) / client assignment	Clients 1–2 use CelebA (face domain) and Clients 3–4 use Places365-Standard (scene domain); each client receives a disjoint split
Main comparison setting	4 secrets / 4 clients
Image resolution	All carrier, stego, and secret images are resized to 256x256 RGB

Protocol item	Current manuscript entry
Payload budget	4 full-size secrets per carrier in the main comparison; K in 2, 4, 6, 8 in the scaling study
Train / validation / test partition	Each secret client uses 80,000 training images, 5,000 validation images, and 5,000 test images after resizing and filtering
Key generation / wrong-key sampling	Each route is assigned one 128-bit binary key; wrong-key queries sample uniformly from the remaining registered keys
Data augmentation	Random horizontal flip, random resized crop, JPEG recompression (Q in [70,95]), color jitter, and Gaussian noise (sigma <= 0.02)
Optimization across K	AdamW, learning rate 1x10^-4, batch size 8, one local epoch, and 100 rounds are fixed for K in 2, 4, 6, 8

Baselines. Table 3 compares task-aligned methods trained under the same protocol: a joint extractor, DEEPMIH [4], a recent generative hiding baseline [7], a centralized private-key model [6], FedAvg without personalization, and a centralized version of our architecture. Figure 5 adds 4bit-LSB, HiDDeN [16], and Baluja [2] as visual anchors for route leakage.

Metrics. Carrier fidelity is measured by PSNR and SSIM between the carrier image x and the stego image y. Secret fidelity is measured by PSNR, SSIM, and LPIPS between the ground-truth secret and the recovered target. Crosstalk suppression is evaluated by CTI, and key selectivity is evaluated by the key selectivity margin

$$\mathrm{KSM}_i = \mathrm{sim}(\hat{s}_i, s_i) - \max_{j \neq i} \mathrm{sim}(\hat{s}_i, s_j). \tag{13}$$

For CTI and KSM, similarity is computed in the fixed ResNet-50 feature space. We also report steganalysis AUC, communication cost, and local time. StegAUC is a black-box detector evaluation; targeted adaptive steganalysis or white-box adversarial attacks are outside the validated threat model and are treated as deployment risks rather than solved attacks.

Implementation details. Each setting uses three random seeds. Unless stated otherwise, AdamW, learning rate, batch size, local epoch count, communication rounds, and 256×256 resolution are identical across secret counts; only the number of routes/clients changes.

4.2 Quantitative Comparison

Table 3 summarizes the main quantitative comparison. The column layout is designed to capture the carrier quality, secret quality, crosstalk, selectivity, security, and communication cost of all methods under the same payload setting.

Table 3. Main quantitative comparison under the default 4-secrets / 4-clients protocol. Part I reports carrier and secret fidelity. Higher carrier PSNR/SSIM and secret PSNR are better, while lower LPIPS is better. Boldface marks the best result in each column.

Method	Carrier PSNR ↑	Carrier SSIM ↑	Secret LPIPS ↓	Secret PSNR ↑
Joint multi-image extractor	41.40 ± 0.41	0.9498 ± 0.0049	0.2410 ± 0.0110	40.18 ± 0.38

Method	Carrier PSNR ↑	Carrier SSIM ↑	Secret LPIPS ↓	Secret PSNR ↑
DeepMIH (TPAMI'23)	**42.68 ± 0.29**	**0.9608 ± 0.0032**	**0.1430 ± 0.0075**	**43.46 ± 0.24**
Generative hiding baseline (2024)	42.51 ± 0.31	0.9597 ± 0.0035	0.1490 ± 0.0078	43.21 ± 0.26
Centralized private-key model	42.05 ± 0.36	0.9561 ± 0.0043	0.1820 ± 0.0091	42.45 ± 0.31
FedAvg without personalization	41.14 ± 0.44	0.9472 ± 0.0054	0.2330 ± 0.0121	40.30 ± 0.43
Centralized FedKeyMIS architecture	42.32 ± 0.33	0.9583 ± 0.0038	0.1690 ± 0.0086	42.76 ± 0.29
FedKeyMIS	41.96 ± 0.35	0.9554 ± 0.0041	0.1540 ± 0.0080	43.08 ± 0.27

Table 3. Main quantitative comparison under the default 4-secrets / 4-clients protocol (continued). Part II reports crosstalk, selectivity, steganalysis, and communication cost. Higher KSM is better, while lower CTI, SER, StegAUC, and communication cost are better.

Method	CTI ↓	KSM ↑	SER ↓	Steg AUC ↓	Comm. cost ↓
Joint multi-image extractor	0.1860 ± 0.0076	0.0740 ± 0.0060	1.000	0.7360 ± 0.0170	0.00 ± 0.00
DeepMIH (TPAMI'23)	0.2210 ± 0.0088	0.0490 ± 0.0054	1.000	0.6480 ± 0.0120	0.00 ± 0.00
Generative hiding baseline (2024)	0.1740 ± 0.0071	0.1010 ± 0.0058	1.000	0.6540 ± 0.0120	0.00 ± 0.00
Centralized private-key model	0.1180 ± 0.0055	0.1410 ± 0.0066	0.500	0.6820 ± 0.0140	0.00 ± 0.00
FedAvg without personalization	0.2010 ± 0.0084	0.0610 ± 0.0061	0.684	0.7510 ± 0.0190	11.38 ± 0.62
Centralized FedKeyMIS architecture	0.1030 ± 0.0048	0.1560 ± 0.0057	**0.250**	0.6590 ± 0.0130	0.00 ± 0.00
FedKeyMIS	**0.0790 ± 0.0041**	**0.1910 ± 0.0052**	**0.250**	**0.6330 ± 0.0120**	12.26 ± 0.71

Table 3 shows that high-fidelity joint extractors still expose the whole set: they have low KSM, high CTI, and SER $= 1$. Personalized federation lowers CTI from 0.103 to 0.079 and raises KSM from 0.156 to 0.191 relative to the centralized keyed baseline. FedKeyMIS also reaches the ideal $1/K$ exposure ratio and the lowest StegAUC under the tested black-box detector.

4.3 Selective Access Evaluation

Table 4 distinguishes correct-key decoding, unauthorized wrong-key decoding, and whole-set extraction.

Table 4. Selective-access evaluation. Matched keys recover one target route; wrong or unseen keys should yield low-fidelity outputs and small exposure ratios.

Query setting	Secret PSNR ↑	Secret SSIM ↑	LPIPS ↓	SER ↓	Retrieval success ↑
Matched key (target route)	43.08	0.964	0.154	0.250	0.982
Wrong registered key	18.07	0.418	0.608	0.071	0.041
Random unseen key	17.42	0.391	0.624	0.053	0.024
Whole-set extraction baseline	40.18	0.941	0.241	1.000	1.000

4.4 Ablation and Scalability Analysis

Table 5 removes the keyed router, local adapters, federated training, and route-separation loss; Table 6 varies the number of secrets/clients.

Table 5. Ablation study of FedKeyMIS.

Variant	CTI ↓	KSM ↑	Secret PSNR ↑
Full FedKeyMIS	**0.0790 ± 0.0041**	**0.1910 ± 0.0052**	**43.08 ± 0.27**
w/o keyed router	0.1480 ± 0.0062	0.0990 ± 0.0069	41.87 ± 0.34
w/o local adapters	0.1220 ± 0.0054	0.1290 ± 0.0061	42.21 ± 0.31
w/o federated training	0.1030 ± 0.0048	0.1560 ± 0.0057	42.76 ± 0.29
w/o route-separation loss	0.1340 ± 0.0058	0.1170 ± 0.0064	42.16 ± 0.30

Table 5 shows that keyed routing is central: removing it raises CTI from 0.079 to 0.148 and reduces KSM from 0.191 to 0.099. Local adapters, federated training, and route separation provide complementary gains.

Table 6. Scalability with different numbers of hidden secrets/clients.

Setting	Carrier PSNR ↑	CTI ↓	KSM ↑	MB/round ↓
2 secrets / 2 clients	42.70 ± 0.28	0.0600 ± 0.0034	0.2140 ± 0.0048	76.80 ± 1.60
4 secrets / 4 clients	41.96 ± 0.35	0.0790 ± 0.0041	0.1910 ± 0.0052	153.60 ± 2.30
6 secrets / 6 clients	41.49 ± 0.39	0.1040 ± 0.0049	0.1600 ± 0.0060	230.40 ± 3.10
8 secrets / 8 clients	41.05 ± 0.43	0.1320 ± 0.0058	0.1310 ± 0.0068	307.20 ± 3.90

Table 6 shows no observed collapse up to 8 routes: carrier PSNR drops from 42.70 to 41.05, CTI rises from 0.060 to 0.132, and KSM falls from 0.214 to 0.131 smoothly. The practical limit is communication and adapter storage. Each route adds about 0.8M local parameters and about 76.8MB per round in the uncompressed exchange, which is acceptable for server-assisted or desktop clients but heavy for small edge devices. Naively extending to 16 or 32 routes would scale communication nearly linearly; practical deployment should use partial client sampling, adapter quantization/sparsification, low-rank adapters, hierarchical key routing, or distillation. Thus 8 routes is the validated operating range in this paper, while larger-scale optimization is left as engineering work.

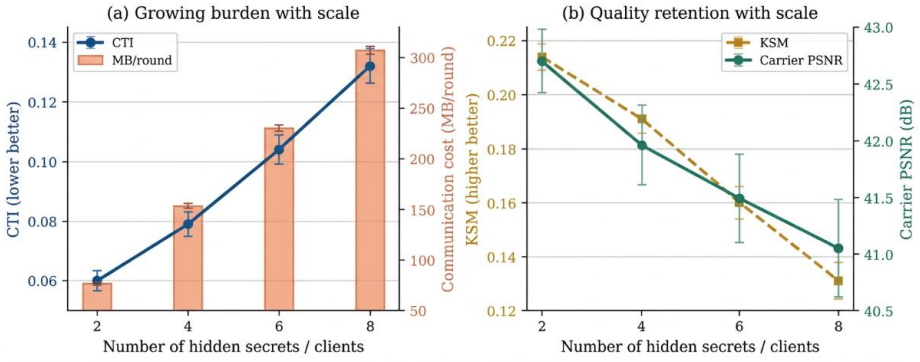

Fig. 4. Scalability trade-offs. Error bars denote standard deviation across repeated runs.

Figure 4 shows the same gradual degradation: cost increases with scale, while quality decreases steadily rather than abruptly.

4.5 Qualitative Analysis

Figure 5 shows representative visual comparisons for cross-secret leakage.

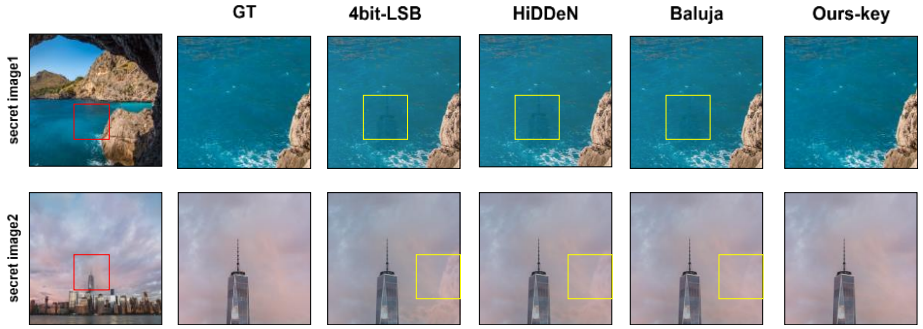

Fig. 5. Qualitative comparison. Highlighted crops show that FedKeyMIS recovers cleaner local structures with fewer residual artifacts.

All methods use the same target crop; FedKeyMIS produces sharper outlines and cleaner texture recovery in the highlighted regions.

5 Conclusion

This paper presented FedKeyMIS, a keyed federated framework for selective multi-image steganography. Coupling each secret-key pair to a client-specific adapter reduces cross-secret interference and enables one-query-one-secret extraction. Results show strong fidelity, low crosstalk, high key selectivity, and favorable black-box StegAUC under the tested protocol. The scaling study shows gradual degradation up to 8 routes but also highlights deployment limits: communication grows nearly linearly, edge devices require compression or sparse adapters, and targeted adaptive steganalysis remains future work.

6 References

1. [1] Arivazhagan, M.G., Aggarwal, V., Singh, A.K., Choudhary, S.: Federated learning with personalization layers. CoRR **abs/1912.00818** (2019)
2. [2] Baluja, S.: Hiding images in plain sight: Deep steganography. In: NeurIPS. pp. 2069–2079 (2017)
3. [3] Dinh, C.T., Tran, N.H., Nguyen, T.D.: Personalized federated learning with moreau envelopes. In: NeurIPS (2020)
4. [4] Guan, Z., Jing, J., Deng, X., Xu, M., Jiang, L., Zhang, Z., Li, Y.: DeepMIH: Deep invertible network for multiple image hiding. IEEE TPAMI **45**(1), 372–390 (2023)
5. [5] Kairouz, P., McMahan, H.B., et al.: Advances and open problems in federated learning. Found. Trends Mach. Learn. **14**(1–2), 1–210 (2021)
6. [6] Kweon, H., Park, J., Woo, S., Cho, D.: Deep multi-image steganography with private keys. Electronics **10**(16), 1906 (2021)

7. [7] Li, G., Li, S., Qian, Z., Zhang, X.: Cover-separable fixed neural network steganography via deep generative models. CoRR **abs/2407.11405** (2024)
8. [8] Li, T., Hu, S., Beirami, A., Smith, V.: Ditto: Fair and robust federated learning through personalization. In: ICML. pp. 6357–6368 (2021)
9. [9] Li, X., Jiang, M., Zhang, X., Kamp, M., Dou, Q.: FedBN: Federated learning on non-IID features via local batch normalization. In: ICLR (2021)
10. [10] Lu, S.P., Wang, R., Zhong, T., Rosin, P.L.: Large-capacity image steganography based on invertible neural networks. In: CVPR. pp. 10816–10825 (2021)
11. [11] McMahan, B., Moore, E., Ramage, D., Hampson, S., y Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data. In: AISTATS. pp. 1273–1282 (2017)
12. [12] Tan, A.Z., Yu, H., Cui, L., Yang, Q.: Towards personalized federated learning. IEEE TNNLS **34**(12), 9587–9603 (2023)
13. [13] Zhang, C., Benz, P., Karjauv, A., Sun, G., Kweon, I.S.: UDH: Universal deep hiding for steganography, watermarking, and light field messaging. In: NeurIPS (2020)
14. [14] Zhang, K.A., Cuesta-Infante, A., Xu, L., Veeramachaneni, K.: SteganoGAN: High capacity image steganography with GANs. CoRR **abs/1901.03892** (2019)
15. [15] Zhang, R., Dong, S., Liu, J.: Invisible steganography via generative adversarial networks. Multimedia Tools Appl. **78**(7), 8559–8575 (2019)
16. [16] Zhu, J., Kaplan, R., Johnson, J., Fei-Fei, L.: HiDDeN: Hiding data with deep networks. In: ECCV. pp. 682–697 (2018)