SUPPLEMENTARY OF "COOPERATIVE HARDWARE-PROMPT LEARNING FOR SNAPSHOT COMPRESSIVE IMAGING"

Anonymous authors
Paper under double-blind review

1 SOCIAL IMPACT

This work develops a federated learning treatment to enable the collaboration of the CASSI systems with different hardware configurations. The proposed method will practically encourage the cross-institution collaborations with emerging optical system designs engaged. By improving the robustness of the pre-trained reconstruction software backend toward optical encoders, this work will help expedite the efficient and widespread deployment of the deep models on sensors or platforms.

2 PERFORMANCE ON MORE CLIENTS

Table 1: Performance of FedAvg and FedHP under different number of clients.

#Clients	FedAvg (McMahan et al., 2017)		FedHP	
	PSNR	SSIM	PSNR	SSIM
3	$31.21_{\pm 0.10}$	$0.8959_{\pm 0.0017}$	$31.35_{\pm 0.10}$	$0.9033_{\pm 0.0014}$
4	$31.06_{\pm 0.10}$	$0.8955_{\pm 0.0018}$	$31.33_{\pm 0.13}$	$0.9023_{\pm 0.0018}$
5	$31.05_{\pm 0.10}$	$0.9025_{\pm 0.0014}$	$31.32_{\pm 0.19}$	$0.9029_{\pm 0.0019}$

In this section, we discuss the effectiveness of the proposed FedHP on more clients (*i.e.*, $C = 4, 5$). Specifically, we adopt the same training dataset for different settings and evenly split it according to the number of clients. As shown in Table 1, FedHP consistently outperforms the FedAvg, which indicates that the proposed collaborative learning solution is more robuts to the number of clients. Besides, we observe that FedHP remains a relative stable performance on different number of clients. By comparison, FedAvg suffers from an obvious performance descent when $C = 4, 5$. Intuitively, FedHP collaboratively learns a global prompter for a client-specific input data space adaptation, which can effectively solve the distribution gap induced by different hardware instances. However, FedAvg learns a shared backbone for different data distributions, which inevitably suffers from the client drift. This issue will be strengthened when more clients participate in the learning. We leave the exploitation of very large number of clients (*e.g.*, $C = 100$) into the future works.

3 ALGORITHM

The learning procedure of proposed FedHP is provided in Algorithm 1. Let us take one global round for example, the learning can be divided into four stages. (1) Initializing the global prompt network from scratch and then distributing it to local clients. Then instantiating the client backbones with the pre-trained models upon the local training dataset. The adaptors are also randomly initialized for a better adaptation of the pre-trained backbones to the aligned input data representation. (2) Local updating of the prompt network, during which all the other learnable parameters in the system are kept fixed. (3) Local updating of the adaptors. Notably, the parameters of the adaptors is only updated and maintained in local. (4) Global aggregation of the local prompt networks.

Algorithm 1 FedHP Training Algorithm

Input: Number of global rounds T; Number of clients C; Number of client subset C'; Pre-trained models θ_c^p, $c = 1, ..., C$; Number of local update iterations S_p, S_b; Random initialized parameter of prompt network ϕ_G; Random initialized parameter of adaptors of c-th client ϵ_c; Learning rate α_p of prompt network; Learning rate α_b of adaptors;

Output: $\phi_G, \epsilon_c, c = 1, ..., C$;

Server Executes;
Randomly choose a set of clients of number C';
for $t = 1, ..., T$ **do**
  for $c \in C'$ in parallel **do**
    Send global prompt network ϕ_G to ϕ_c;
    $\phi_c \leftarrow$ LocalTraining(θ_c^p, ϵ_c, ϕ_c);
  end for
  $\phi_G \leftarrow \sum_{c=1}^{c=C'} \frac{|\mathcal{D}_c|}{|\mathcal{D}|} \phi_c$;
end for
return ϕ_G;
LocalTraining(θ_c^p, ϵ_c, ϕ_c);
for $s = 1, ..., S_p$ **do**
  $\phi_c \leftarrow \phi_c - \alpha_p \nabla \ell(\theta_c^p, \epsilon_c, \phi_c)$ using $\ell_c = \frac{1}{N} \sum_{i=1}^{N} ||f(\theta_c^p, \epsilon_c; \mathbf{Y}_i^{\mathbf{M}^c} + \Phi(\mathbf{M}^c)) - \mathbf{X}_i||_2^2$;
end for
for $s = 1, ..., S_b$ **do**
  $\epsilon_c \leftarrow \epsilon_c - \alpha_b \nabla \ell(\theta_c^p, \epsilon_c, \phi_c)$ using $\ell_c = \frac{1}{N} \sum_{i=1}^{N} ||f(\theta_c^p, \epsilon_c; \mathbf{Y}_i^{\mathbf{M}^c} + \Phi(\mathbf{M}^c)) - \mathbf{X}_i||_2^2$;
end for
return ϕ_c to server;

4 Visualization Results

In this section, we provide more visualization results of different methods. In Figs. 1∼2, we present the reconstruction results of different methods under the scenario of `hardware shaking`, *i.e.*, the data heterogeneity is naively induced from the different CASSI instances across clients. FedHP enables more fine-grained details retrieval. Besides, we compare the spectral density curves on selected representative spatial regions. The higher correlation to the reference, the better spetrum consistency with the ground truth. In Figs. 3∼4, we show additional real reconstruction results of FedAvg and FedHP on selected wavelengths. By comparison, FedAvg fails to reconstruct some content, while the proposed FedHP allows a more granular result.

5 Coded Aperture Distributions

In Figs. 5∼7, we visualize the different distributions of coded apertures in distinct clients under the scenario of the distribution shift of coded apertures among different clients leads to the data heterogeneity among different local input dataset. This mimics a very challenging scenario where in different clients (*e.g.*, research institutions), the corresponding CASSI systems source from different manufacturers. The proposed FedHP allows a potential collaboration among different institutions for the hyperspectral data acquisition for the first time despite the large distribution gap. By comparison, classic methods of FedProx (Li et al., 2020) or SCAFFOLD (Karimireddy et al., 2020) fail to provide reasonable retrieval results.

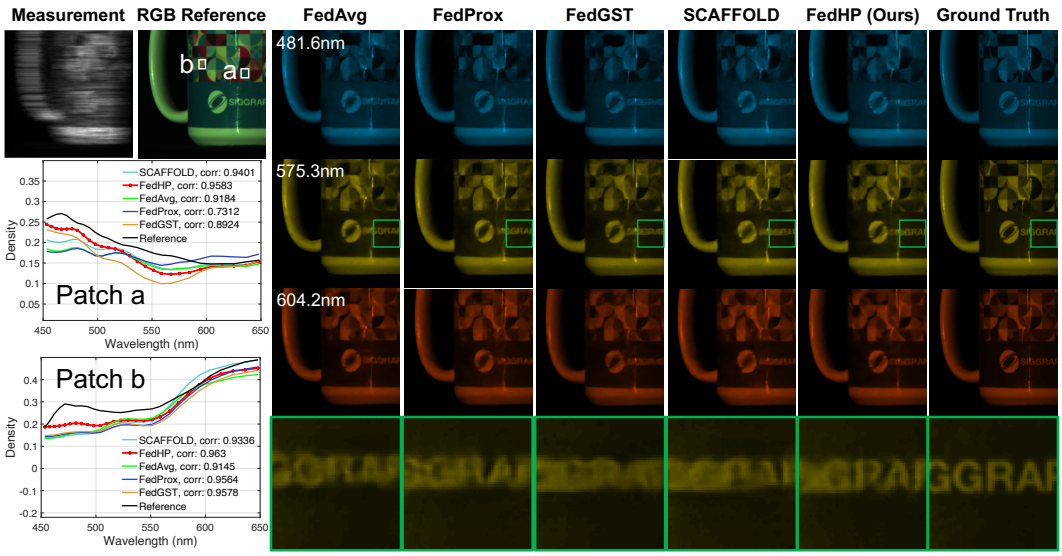

Figure 1: Reconstruction results on simulation data. The density curves compares the spectral consistency of different methods to the ground truth. We use the same coded aperture for all methods.

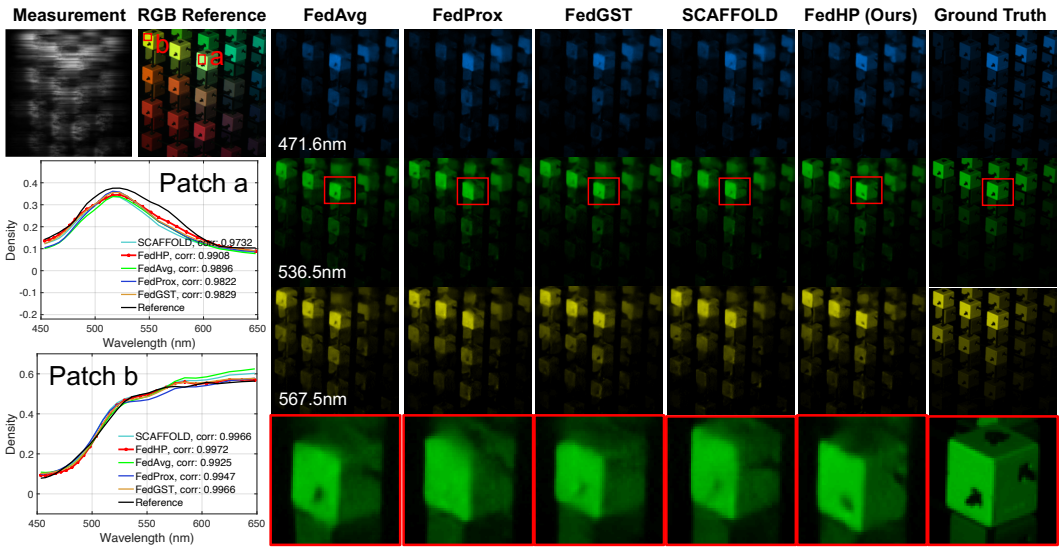

Figure 2: Reconstruction results on simulation data. The density curves compares the spectral consistency of different methods to the ground truth. We use the same coded aperture for all methods.

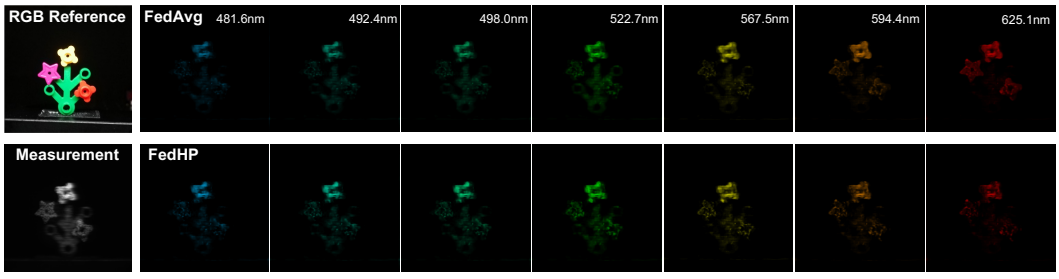

Figure 3: Visualization of reconstruction results on real data. Seven (out of 28) representative wavelengths are selected. We use the same unseen coded aperture for both FedAvg and FedHP.

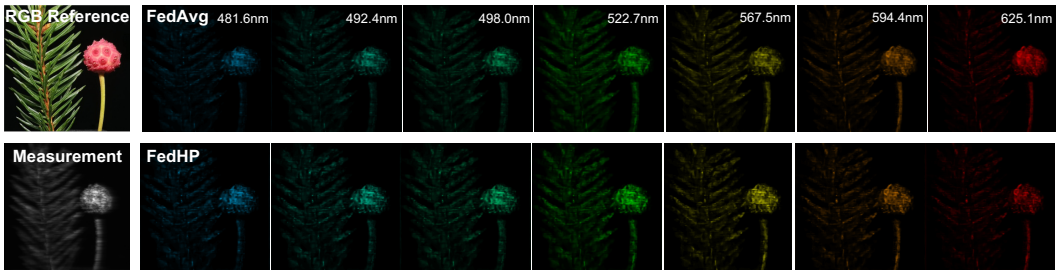

Figure 4: Visualization of reconstruction results on real data. Seven (out of 28) representative wavelengths are selected. We use the same unseen coded aperture for both FedAvg and FedHP.

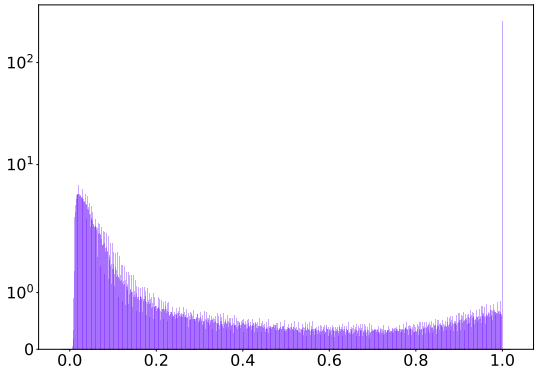

Figure 5: Distribution of Coded apertures in Client 1 ($C = 3$) under the scenario of `manufacturing discrepancy`. The symmetrical logarithm scale is employed for a better visualization.

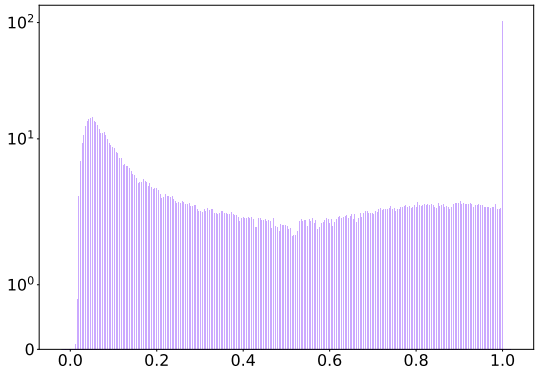

Figure 6: Distribution of Coded apertures in Client 2 ($C = 3$) under the scenario of `manufacturing discrepancy`. The symmetrical logarithm scale is employed for a better visualization.

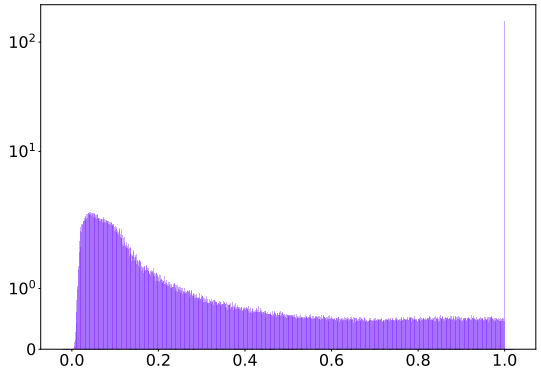

Figure 7: Distribution of Coded apertures in Client 3 ($C = 3$) under the scenario of `manufacturing discrepancy`. The symmetrical logarithm scale is employed for a better visualization.

REFERENCES

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *ICML*, 2020. 2

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. In *MLSys*, 2020. 2

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *AISTATS*, 2017. 1