FL-CycleGAN: Enhancing Mobile Photography with Federated Learning-Enabled CycleGAN

Ramindu Walgama
School of Computer Science and Engineering
University of Westminster
London, United Kingdom
w1790183@my.westminster.ac.uk

K.T. Yasas Mahima
School of Engineering and Technology
University of New South Wales
Canberra, ACT, Australia
yasas.mahima@unsw.edu.au

2024 Moratuwa Engineering Research Conference (MERCon) | 979-8-3315-2904-8/24/$31.00 ©2024 IEEE | DOI: 10.1109/MERCon63886.2024.10688983

***Abstract*—Mobile image photography is continuously emerging as an area of interest, yet achieving professional-level camera quality remains a challenge due to hardware limitations. In order to improve the images taken from mobile phones, deep learning-based image processing techniques such as convolutional neural networks are proposed. However, these networks are typically trained using large amounts of paired data and lack continuous training using images captured from mobile phone users. This is because, in reality, creating paired image datasets from user-captured images is challenging and may lead to user privacy issues. As a solution to this challenge, this research proposes FL-CycleGAN, a novel federated learning-based CycleGAN designed to improve the colors of mobile images continuously using user-captured images in an unpaired manner. The evaluations on the ZurichRAW to RGB dataset reveal that FL-CycleGAN reconstructs the colors of mobile images with an average PSNR value of 18.46 and SSIM value of 0.707, demonstrating comparable results to state-of-the-art networks based on paired images. Furthermore, FL-CycleGAN reconstructs high-resolution images with a size of 3968 $\times$ 2976 in under 0.005 seconds.**

Keywords—Federated Learning, GANs, Distributed Learning, Image Enhancement, Mobile Photography

I. Introduction

At present, the use of mobile devices has experienced significant growth in the field of photography due to their portability. However, integrating advanced large sensors used in professional cameras (e.g. Digital Single Lens Reflex (DSLR) camera) into mobile phone cameras poses a challenge due to their limited size [1]. This phenomenon has resulted in various limitations in mobile photography, such as increased noise in captured low-light images, a smaller dynamic range, and fixed aperture and zoom levels. Currently, some mobile companies are partially addressing this issue by integrating multiple cameras [2].

In order to further mitigate the aforementioned limitations, mobile phones have been designed to engage in the post-processing of images, which involves processing images after they are captured. However, this could result in high computational constraints and performance issues. The primary technology behind these processing tasks involves complex image processing and computer vision algorithms, which are typically regarded as proprietary secrets of smartphone manufacturers [2]. As a result, it's evident that images captured by mobile phones from various manufacturers have distinct color schemes, leading to competition between smartphones from different manufacturers [3].

Fig. 1. The original Huawei P20 RAW image and the corresponding FL-CycleGAN reconstructed image.

These post-image processing methods use Deep Learning techniques such as Convolutional Neural Networks (CNNs), which operate on dedicated Graphics Processing Unit (GPU)s available in smartphones [4], [5]. However, training these CNNs requires large paired image datasets [6], which are challenging to collect. Further, with continuous advancements in smartphone camera technologies, it is essential to continuously train the network to prevent smartphone camera quality from being outdated. Furthermore, collecting images from the user's end to train such post-processing networks may lead to potential violations of user privacy, considering the likelihood of including sensitive and personal content [6].

To address the aforementioned challenges in current post-image processing methods on mobile phones, this paper introduces a CycleGAN network [7] based on Federated Learning (FL) [8] which is a distributed unpaired image network to enhance the colors of the mobile images while preserving privacy and facilitating continuous learning. The experimental results demonstrate that the proposed approach can enhance the colors of mobile images (Fig. 1), achieving PSNR values of **18.4665** and SSIM value of **0.7072**, with a processing time around **0.005** seconds under the high-resolution mobile images (3968 × 2976) available in ZurichRAW to RGB Dataset [9]. The main contributions of this study are as follows:

1) Propose an unpaired mobile image enhancement model, which, to the best of our knowledge, is the first to apply CycleGAN in the mobile image enhancement domain to enhance the image colors.
2) Utilize FL based distributed training for mobile image color enhancement for the first time, enabling continuous

learning while protecting user privacy.

3) Evaluate and demonstrate that the proposed approach can achieve state-of-the-art performance when compared to solutions trained based on paired images.

The rest of the paper is organized as follows: Section II presents a review of the state-of-the-art solutions. In Section III, the proposed solution is presented, followed by the problem formulation. Then, Section IV summarizes the experimental setup. The evaluation of the proposed approach, along with benchmark results, is presented in Section V. Finally, Section VI provides a discussion of the research findings and concludes the paper.

II. Related Works

A. Paired Image Enhancement

Recent image enhancement techniques introduced for mobile devices have relied on CNNs [5], [9]–[11].

The DPED network [11] was proposed along with a paired image dataset captured using three different smartphones (iPhone 3GS, BlackBerry Passport, Sony Xperia Z) and a Canon 70D professional camera. Specifically, this study introduced three models, each trained on one of the smartphone image datasets alongside the professional camera images. The model trained on Sony images had the best PSNR and SSIM values compared to the other two models. However, it is worth noting that by default photo quality on the Sony smartphone surpasses that of the other two mobile devices. The PyNET [9] follows a similar architecture but is trained with ZurichRAW to RGB dataset, which contains images taken using a Huawei P20 smartphone and a Canon 5D Mark IV camera. It processes a 3958×2944 image in 3.8 seconds on an Nvidia Tesla V100 GPU. The PyNET model is compatible with smartphone GPU accelerators [9]. PyNET-V2 [5] is an improved lightweight deep CNN model with a latency of 0.5 seconds outperforming PyNET [9]. The PyNET-V2 model is trained on RAW RGB image pairs captured with both a professional medium-format 102MP Fujifilm camera and a Sony mobile camera. Further, studies presented in the mobile image denoising challenge [10] have employed U-net-like architectures.

B. Unpaired Image Enhancement

An unpaired image Generative Adversarial Network (GAN) model was introduced in [12] to enhance low-light images. Further, several unpaired GAN networks [13] have been proposed for underwater image enhancement, specifically focusing on removing green/blue color tint from images. Notably, study [13] utilized a CycleGAN [7]-based approach. Furthermore, an approach was introduced in [14] that combines CycleGAN [7] with VGG-16-based perceptual loss to convert grayscale medical images into colored ones. To the best of our knowledge, there is no unpaired image-based approach for enhancing the colors of mobile images. In reality, it is challenging to prepare paired image datasets from user-captured mobile images to train a mobile image enhancement network. This difficulty makes it impossible to continuously improve such networks.

C. Distributed Unpaired Image Processing Networks

At present, FL is considered as one of the main distributed learning techniques which help to preserve the privacy of the data [15], [16]. In particular, distributed unpaired image processing based on GAN is a relatively unexplored research topic. Studies, [17], [18] are two closely resembled studies with ours that use CycleGAN based FL approach for processing images. Study, [17] introduced a network named FedMedGAN aiming to perform domain translation for unsupervised brain image synthesis. Here, the training process of the FL limits to synchronize only the generator, where the discriminators do not participate in FL. However, study [19] suggested synchronizing both the generator and the discriminator in the FL process for better results unless there are network bandwidth constraints. Following this, in study [18] both generators and discriminators of the CycleGAN are synchronized. Specifically, this network was proposed to address denoising CT scans in the medical domain, as well as style transfer such as summer-to-winter, photo-to-Van Gogh and horse-to-zebra. However, this method separated two domain images into two clients. For example, client 1 has domain A images, while client 2 has domain B images. In contrast, the proposed solution in this study provides a complete continuous learning architecture that combines FL and CycleGAN, allowing training with data directly taken from the user in an unpaired manner.

III. Methodology

A. Problem Formulation

Let $\mathcal{I}$ be the unpaired image dataset containing domain X RAW mobile images and domain Y professional camera images, where the scene S represented by each corresponding image in two domains are district as $S_{x_i} \neq S_{y_i}$ and $i \leq |\mathcal{I}|$. As mentioned previously, in reality, it is challenging to capture the images of the same scene from two domains where $S_{x_i} = S_{y_i}$ as smartphone users only capture images in domain X. Thus, we intend to train an image enhancement network $\mathcal{M}$ that enables continuous learning using unpaired datasets ($\mathcal{I}_k$) generated from k different smartphone users (Assume this as domain X) and a pre-collected professional camera image dataset (Assume this as domain Y) in a distributed manner. Specifically, the dataset in each device is unique because $\mathcal{I}_n(X) \neq \mathcal{I}_p(X)$ where $p, n \in \{1, ..., k\} \wedge p \neq n$. We denote the data distribution as $x \sim p_{data}(x)$ and $y \sim p_{data}(y)$. The ultimate objective of $\mathcal{M}$ in the inference is to convert raw mobile images to professional camera images as $\mathcal{M}(x) \rightarrow y$, regardless of the smartphone type.

B. FL-CycleGAN

To achieve the aforementioned objective, we introduce FL-CycleGAN, a CycleGAN network trained in a distributed manner based on FL. We choose the CycleGAN model because it allows continuous training of the model with new data using the cycle-consistency loss function in an unsupervised and unpaired manner.

Diving into the proposed solution, FL-CycleGAN contains a CycleGAN network deployed in a centralized location denoted as $\mathcal{M}_S$ and k number of CycleGAN networks similar to the $\mathcal{M}_S$ employed in each k number of user devices as $\mathcal{M}_{Cn}$ where $n \in \{1, ..., k\}$. Here we assume that each $\mathcal{M}_C$ is trained using the domain X images collected from the particular user and provided professional camera domain Y images. After training each $\mathcal{M}_C$ for r amount of rounds the weights and biases of each network are sent synchronized with the $\mathcal{M}_S$ as $\mathcal{M}_S = \mathcal{F}(\mathcal{M}_{C1}, .., \mathcal{M}_{Ck})$ where $\mathcal{F}$ represents the federated algorithm. As the $\mathcal{F}$, we use the FedAvg algorithm [8], which averages the weights and biases of each $\mathcal{M}_C$, as in Eq 1.

$$\mathcal{F}(\mathcal{M}_{C1}, .., \mathcal{M}_{Ck}) = \sum_{n \in \{1,...,k\}} \frac{d_n}{\sum_{n' \in \{1,...,k\}} d_{n'}} w_n, \quad (1)$$

where d_n represents the number of data points in each user and w_n represents the trained weights from each user.

The aggregated weights of the $\mathcal{M}_S$ are again redistributed back to update each $\mathcal{M}_{Cn}$. In particular, the CycleGAN networks in $\mathcal{M}_C$ and $\mathcal{M}_S$ consist of two GANs, and following the suggestion of [19], both discriminators and generators of each GAN pass through the synchronization process to enhance the colors of mobile images which differentiates it from [17], which involves only the generators in the FL process.

The adversarial loss is used to train each GAN as presented in Eq 2. Here, each generator G aims to minimize the objective against an adversary discriminator D, that tries to maximize it, i.e. $min_G max_D \mathcal{L}_{GAN}(G, D, X, Y)$.

$$\begin{aligned}\mathcal{L}_{GAN}(G, D, X, Y) = \mathbb{E}_{x \sim p_{data}(y)}[logD(y)] \\ + \mathbb{E}_{x \sim p_{data}(x)}[log(1 - D(G(x))].\end{aligned} \quad (2)$$

Additionally, the cycle-consistency loss function is used to train the generators. This loss function involves the ℓ_1 function between the recreated image $G_B(G_A(x))$ and the original image x.

$$\begin{aligned}\mathcal{L}_{cyc}(G_A, G_B) = \mathbb{E}_{x \sim p_{data}(x)}[||G_B(G_A(x)) - x||_1] + \\ \mathbb{E}_{y \sim p_{data}(y)}[||G_A(G_B(y)) - y||_1].\end{aligned} \quad (3)$$

The final objective function is represented by Eq. 4.

$$\begin{aligned}\mathcal{L}(G_A, G_B, D_X, D_Y) = \mathcal{L}_{GAN}(G_A, D_Y, X, Y) \\ + \mathcal{L}_{GAN}(G_B, D_X, Y, X) \\ + \lambda \mathcal{L}cyc(G_A, G_B).\end{aligned} \quad (4)$$

The training process of the FL-CyleGAN network is presented in Algorithm 1 and a high-level architecture of the solution is presented in Fig. 2.

Algorithm 1 FL-CycleGAN Algorithm

```
Initialize M_S on central server and M_C on each clients
Developer Send I(y) to each client
for fl_round = 0 to ∞ do
    # add more clients (Optional)
    for n = 1 to k + 1 in parallel do
        # local training
        I_n = gather_images_n()
        M_{C_n}(I_n)
        w_n = get_parameters(M_{C_n})
    end for
    W = F(M_{C1}, .., M_{Ck}) # client weights aggregation
    for n = 1 to k + 1 in parallel do
        # client weights update
        w_n = W
        set_parameters(w_n)
        # developer randomly send domain y data update
        c = randint(0 − 5)
        if c == 0 then
            Send and Update I(y)
        end if
    end for
end for
```

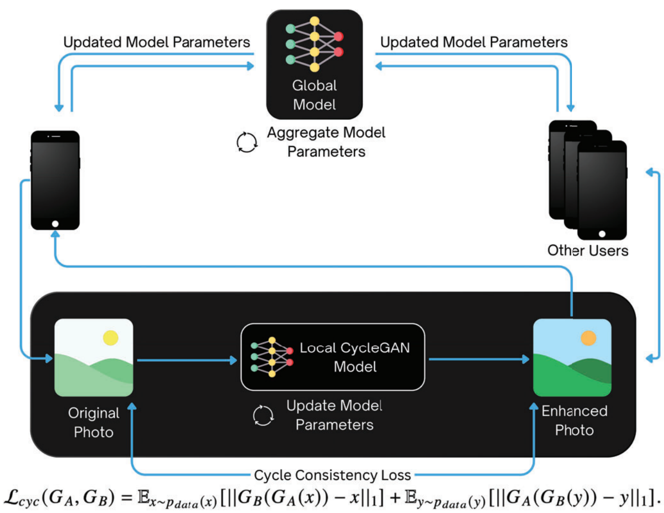

Fig. 2. FL-CycleGAN Architecture: Each client consists of a CycleGAN model that continuously trains with newly captured images. The trained parameters will then go through the federated learning process.

IV. Experimental Setup

A. Datasets

The training is conducted on the ZurichRAW to RGB dataset [9]. This dataset consists of cropped patches from larger images captured by both a professional Canon 5D Mark IV camera and a Huawei P20 camera, totaling 46,839 cropped image pairs. The training utilized Huawei P20 RAW visualized input images to generate images similar to those captured by a professional camera. Notably, though this dataset is a paired image dataset, we feed the data to the model in an unpaired manner by randomly slicing it into two clients.

B. Implementation Details

The generators of the proposed FL-CycleGAN network were implemented using UNet-256 and ResNet-9 architectures. Each model was trained with two clients, and each client ran for 6 epochs per FL round. Additionally, a single

ResNet-9 client model carried out further training using the VGG perceptual loss function along with the SSIM loss. We refer to this as ResNet-9-C1.

The federated architecture was implemented using the Flower Framework [20] along with PyTorch. This Flower Framework enables training on different machines. The models were trained on a single RTX-3080 10GB GPU. Each client with one batch size only took around 4 GB of memory, totaling up to 8 GB out of 10 GB for both clients. We further, hosted the FL server on an AWS EC2 8GB Ubuntu machine. It was noted that the RAM of the central server is mostly idle and uses around 6 GB or less, with maximum usage occurring during the aggregation process.

C. Evaluation Metrics

In order to evaluate the quality of the generated samples, PSNR and SSIM metrics available in [21] are used as depicted in Eq 5 and Eq 6 respectively.

$$\mathrm{PSNR} = 10 \log_{10} \left(\frac{\mathrm{peakval}^2}{\mathrm{MSE}} \right), \tag{5}$$

where $peakval$ refers to the Peak Value which is the maximal in the image data.

$$\mathrm{SSIM} = [l(x,y)]^{\alpha}.[c(x,y)]^{\beta}.[s(x,y)]^{\gamma}, \tag{6}$$

where l: luminance, c: contrast, s:structure while α, β, γ are parameters greater than 0.

V. Results

A. Quantitative Results

Followed by the previous works [9], [11], the FL-CycleGAN network is also evaluated on 1176 cropped images using the ZurichRAW to RGB dataset. It is noticed that some of the images taken from the mobile camera are not focused, while the targeted professional camera shots are focused in the testing dataset. Thus, to reduce the impact of the quality of the dataset for model results, we sort the PSNR values of all testing images and take each quartile as listed in Table I.

In this experiment, we observed that UNet-256 trains faster than the ResNet-9 architecture. However, according to Table I it is clear that ResNet provides better results in terms of visual quality as it outperforms the UNet-256 model demonstrating higher PSNR and SSIM values after 10 FL rounds. Thus, we choose the ResNet-9 model as our primary generator. Moreover, comparing the performance of the primary ResNet-9 models with that of ResNet-9-C1, both models demonstrate nearly similar PSNR and SSIM values, even though the ResNet-9-C1 trained with additional perceptual loss functions.

We further benchmarked FL-CycleGAN against several state-of-the-art image enhancement studies trained on paired images, as to the best of our knowledge, FL-CycleGAN is the first study to propose an unpaired distributed image training approach for mobile image enhancement. Benchmarking results on 1176 images from the ZurichRAW to RGB Dataset

TABLE I
AVERAGE EVALUATION METRICS FOR OUR MODELS ON CROPPED IMAGES PER QUARTILE.

CycleGAN Generator Type	Client Count	Quartile	Test Length	PSNR(↑)	SSIM(↑)
ResNet-9-C1	1	1^{st}	294	23.9550	0.9080
	1	2^{nd}	588	22.1211	0.8430
	1	3^{rd}	882	20.6010	0.7812
	1	4^{th}	1176	18.8845	0.7083
ResNet-9	2	1^{st}	294	23.7414	0.8986
	2	2^{nd}	588	21.8940	0.8373
	2	3^{rd}	882	20.3081	0.7798
	2	4^{th}	1176	18.4665	0.7072
UNet-256	2	1^{st}	294	21.6334	0.8074
	2	2^{nd}	588	19.4438	0.7306
	2	3^{rd}	882	17.7490	0.6714
	2	4^{th}	1176	16.0251	0.6018

are presented in Table II. These results reveal that although FL-CycleGAN uses an unpaired training approach, it demonstrates nearly the same performance as PyNET [9] and outperforms both DPED [11] and EnlightenGAN [12] approaches, which were trained on paired image datasets. Further, none of these benchmarked approaches use FL-based distributed training approach. In general, it is challenging to train a network with unpaired image data as the network has to learn the distribution between two domains without having a pixel-to-pixel mapping. Additionally, since FL-CycleGAN is trained in a distributed manner, it is possible to experience a slight performance drop after each FL round. Hence, the performance of FL-CycleGAN is at an acceptable level. Fig. 3 demonstrates a comparison of reconstructed images from each benchmarked approach.

TABLE II
AVERAGE PSNR AND SSIM VALUES OF DIFFERENT MODELS TESTED ON ZURICHRAW DATASET.

Model	PSNR(↑)	SSIM(↑)	Params(↓)	Unpaired	FL
PyNET [9]	**21.0685**	**0.7582**	47.55 M	✗	✗
DPED [11]	13.8149	0.6105	—	✗	✗
EnlightenGAN [12]	14.1650	0.6259	8.64 M	✓	✗
FL-CycleGAN	**18.4665**	**0.7072**	11.38 M	✓	✓

The CycleGAN model utilized in FL-CycleGAN only uses a single generator during inference, comprised of 11.378 million parameters and 45 MB in size. This is able to process high-resolution images (3968 × 2976 - Huawei P20 RAW Visualized Images) in approximately 0.005 seconds using 7.4 GB of GPU on an RTX-3080 10 GB GPU.

B. Qualitative Evaluation

We conducted a human subjective evaluation among 40 different individuals using the reconstructed images of FL-CycleGAN, PyNET [9], and the Huawei P20 processed images. Some samples are included in Fig. 3 (Columns 2-4). The evaluation contained 8 images from each device, and participants were requested to rate each image (total

Fig. 3. Visual results obtained from different models: Huawei P20 RAW images, Huawei P20 processed images, reconstructed images from FL-CycleGAN (Ours), PyNET [9], EnlightenGAN [12], and ground truth images from Canon 5D Mark IV.

TABLE III
HUMAN SUBJECTIVE EVALUATION.

Rank	Image source	Total Score (↑)
1	PyNET [9]	1092
2	**FL-CycleGAN**	**1088**
3	Huawei P20 Smartphone Processed Image	979

24 images) from 1 to 5 (1 - Poor Quality, 5 - Excellent Quality) based on the color quality of the images. Rankings were determined by calculating the sum of each rating for each image (rating 5 received 5 points for each response, and rating 1 received 1 point for each response). As presented in Table III, FL-CycleGAN achieved 1088 points, securing the second position with 4-points difference from PyNET. These results further confirm that FL-CycleGAN images are visually more appealing compared to Huawei P20 processed images and are much closer in quality to PyNET images.

Based on the examples presented in Fig. 3, it's evident that FL-CycleGAN excels in preserving saturation and contrast compared to other methods. As an example in the second-row images, PyNET lacks improvements in red and blue colors in the traffic sign while FL-CycleGAN significantly improves this region. It also noticed that images processed from EnlightenGAN are overexposed and in some instances, PyNET increases brightness more than the ground truth. Additionally, PyNET model images in the first row exhibit a vintage effect with darkness in the corners, while FL-CycleGAN images demonstrate balanced exposure. Moreover, Huawei P20 processed images are more crispier.

VI. CONCLUSION AND FUTURE WORKS

We propose a novel continuous learning architecture encompassing a CycleGAN model into the FL architecture, which preserves the privacy of the user while maintaining performance in in-device training and image enhancement. To the best of our knowledge, there are no CycleGAN-based mobile image enhancement methods proposed so far, and this is the first to enhance mobile images in a distributed manner.

The proposed solution can be further enhanced by improving the generators/discriminators, adding attention maps [12], and further enhancing images after processing as in [22]. Moreover, we plan to work on compressing and optimizing the solution for deployment on smartphones in different geo-locations, as the solution has been tested on desktop GPUs. Additionally, it is essential to experiment with different FL algorithms to improve the quality of the aggregated weights.

References

[1] K. Singh, A. Seth, H. S. Sandhu, and K. Samdani, “A comprehensive review of convolutional neural network based image enhancement techniques,” in *2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN)*. IEEE, 2019, pp. 1–6.

[2] M. Delbracio, D. Kelly, M. S. Brown, and P. Milanfar, “Mobile computational photography: A tour,” *Annual Review of Vision Science*, vol. 7, pp. 571–604, 2021.

[3] W. Zhu, G. Zhai, Z. Han, X. Min, T. Wang, Z. Zhang, and X. Yangand, “A multiple attributes image quality database for smartphone camera photo quality assessment,” in *2020 IEEE International Conference on Image Processing (ICIP)*, 2020, pp. 2990–2994.

[4] C. Morikawa, M. Kobayashi, M. Satoh, Y. Kuroda, T. Inomata, H. Matsuo, T. Miura, and M. Hilaga, “Image and video processing on mobile devices: a survey,” *The Visual Computer*, vol. 37, no. 12, pp. 2931–2949, 2021. [Online]. Available: https://doi.org/10.1007/s00371-021-02200-8

[5] A. Ignatov, G. Malivenko, R. Timofte, Y. Tseng, Y.-S. Xu, P.-H. Yu, C.-M. Chiang, H.-K. Kuo, M.-H. Chen, C.-M. Cheng *et al.*, “PyNet-V2 Mobile: Efficient on-device photo processing with neural networks,” in *2022 26th International Conference on Pattern Recognition (ICPR)*. IEEE, 2022, pp. 677–684.

[6] L. Tran, D. Kong, H. Jin, and J. Liu, “Privacy-CNH: A framework to detect photo privacy with convolutional neural network using hierarchical features,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, no. 1, Feb. 2016. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/10169

[7] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[8] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, “Communication-efficient learning of deep networks from decentralized data,” in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, A. Singh and J. Zhu, Eds., vol. 54. PMLR, 20–22 Apr 2017, pp. 1273–1282. [Online]. Available: https://proceedings.mlr.press/v54/mcmahan17a.html

[9] A. Ignatov, L. Van Gool, and R. Timofte, “Replacing mobile camera ISP with a single deep learning model,” in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020, pp. 2275–2285.

[10] A. Ignatov, B.-S. Kim, R. Timofte *et al.*, “Fast camera image denoising on mobile GPUs with deep learning, mobile AI 2021 challenge: Report,” in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2021, pp. 2515–2524.

[11] A. Ignatov, N. Kobyshev, R. Timofte, K. Vanhoey, and L. Van Gool, “DSLR-quality photos on mobile devices with deep convolutional networks,” in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[12] Y. Jiang, X. Gong, D. Liu, Y. Cheng, C. Fang, X. Shen, J. Yang, P. Zhou, and Z. Wang, “EnlightenGAN: Deep light enhancement without paired supervision,” 2021.

[13] C. Fabbri, M. J. Islam, and J. Sattar, “Enhancing underwater imagery using generative adversarial networks,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 7159–7165.

[14] Y. Liang, D. Lee, Y. Li, and B.-S. Shin, “Unpaired medical image colorization using generative adversarial network,” *Multimedia Tools and Applications*, vol. 81, no. 19, pp. 26 669–26 683, 2022. [Online]. Available: https://doi.org/10.1007/s11042-020-10468-6

[15] K. H. W. S. M. B. A. V. F. V. P. L. F. Mingzhe Chen, Deniz Gündüz, “Distributed learning in wireless networks: Recent progress and future challenges,” *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 12, pp. 3579–3605, 2021.

[16] Y. Zhuo and B. Li, “FedNS: Improving federated learning for collaborative image classification on mobile clients,” in *2021 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2021, pp. 1–6.

[17] J. Wang, G. Xie, Y. Huang, J. Lyu, F. Zheng, Y. Zheng, and Y. Jin, “FedMed-GAN: Federated domain translation on unsupervised cross-modality brain image synthesis,” *Neurocomputing*, vol. 546, p. 126282, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231223004058

[18] J. Song and J. C. Ye, “Federated CycleGAN for privacy-preserving image-to-image translation,” 2021. [Online]. Available: https://arxiv.org/abs/2106.09246

[19] C. Fan and P. Liu, “Federated generative adversarial learning,” in *Pattern Recognition and Computer Vision*, Y. Peng, Q. Liu, H. Lu, Z. Sun, C. Liu, X. Chen, H. Zha, and J. Yang, Eds. Cham: Springer International Publishing, 2020, pp. 3–15.

[20] D. J. Beutel, T. Topal, A. Mathur, X. Qiu, J. Fernandez-Marques, Y. Gao, L. Sani, K. H. Li, T. Parcollet, P. P. B. de Gusmão, and N. D. Lane, “Flower: A friendly federated learning framework,” Mar. 2022, open-Source, mobile-friendly Federated Learning framework. [Online]. Available: https://hal.science/hal-03601230

[21] C. Chen and J. Mo, “IQA-PyTorch: PyTorch toolbox for image quality assessment,” [Online]. Available: https://github.com/chaofengc/IQA-PyTorch, 2022.

[22] G. Tang, J. Ni, Y. Chen, W. Cao, and S. X. Yang, “An improved CycleGAN based model for low-light image enhancement,” *IEEE Sensors Journal*, pp. 1–1, 2023.