Revolutionizing comic coloring: Federated learning-based neural network for efficiency and privacy

Lunshuo Tian

The Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, 518055, China

12110203@mail.sustech.edu.cn

Abstract. The conventional method of coloring comics can be quite arduous and time-consuming, particularly within the realm of animation, where each frame necessitates individual coloring. Over the past few years, the advent of deep learning technology has introduced fresh prospects for comic coloring. Nonetheless, it's worth noting that deep learning typically demands a substantial volume of data to effectively train models, giving rise to legitimate concerns regarding data privacy. This study proposes an innovative approach that applies federated learning to comic coloring to improve efficiency and address privacy issues. In this research, an existing neural network European Conference on Computer Vision (ECCV16) model was employed using federated learning to partition and train the model on segmented databases. To better quantify the color differences between the original images and the colored images, this study designed a custom loss function. Experimental results demonstrate that this approach has been effective in comic coloring, demonstrating the feasibility of applying federated learning to the task of comic coloring.

Keywords: Deep learning, Federated learning, Comic colorizing.

1. Introduction

Traditionally, comic artists face the task of adding color to black and white sketches, a labor-intensive process that demands substantial time and effort. This task becomes even more resource-intensive when applied to animation, requiring colorization for every frame. To enhance the efficiency of comic creation, attempts have been made to integrate deep learning with the coloring process [1,2].

In this rapidly evolving digital era, artificial intelligence has gained significant prominence across numerous domains, owing to its powerful computational capabilities and the potential to simulate human intelligence. From self-driving cars [3] to natural language processing [4], Artificial Intelligence (AI) has made a great success. Deep learning, a pivotal component of AI, has achieved groundbreaking results in fields such as image recognition [5], speech recognition, and natural language generation, thanks to its multi-layered neural network architecture [6]. However, deep learning relies heavily on models trained on vast amounts of data, and where data is involved, privacy probably needs to be concerned.

In 2023, AI comic generator tools [7] are widely developed and used, but these are frequently used for entertaining purposes rather than commercial purposes. Logically, these tools don’t have necessity to bother the issue of privacy. Thus, it is necessary to focus on the colorization of handwritten comics.

Through extensive data training, models capable of automatically colorizing black and white sketches have been developed. Presently, popular comic coloring AIs like Petalica Paint for Manga [8] and Style2paints [9] exist. Petalica Paint for Manga allows color specification for specific areas, yet color spreading to wrong areas remains an issue, necessitating extensive post-processing. Style2paints offers a few style options, but the choices are limited, failing to match perfectly with an artist's perspective. Moreover, these solutions lack effective methods for safeguarding the privacy of creators' works, potentially resulting in commercial losses [10] when widely used.

To solve the limitation mentioned above, this study has chosen to explore the application of federated learning [11] to comic coloring. Federated learning, as a decentralized training approach, offers a new method for model training across multiple participants. By conducting training locally, federated learning avoids privacy risks associated with centralized datasets, at the same time benefiting individuals from the collective model. By applying federated learning to comic coloring, this study aims to provide personalized coloring while preserving privacy. Allowing comic artists to locally train models, capturing their unique artistic style and color preferences, my approach aims to maintain creators' uniqueness while leveraging the privacy advantages of federated learning. In this research, the ECCV16 [12] neural network model crafted by Richard for training purposes were employed, dividing and training weights using federated learning on segmented databases. During the model optimization process, this study also designed a tailored loss function to quantify color discrepancies between original and colorized images. The experimental results demonstrated the effectiveness of the proposed approach.

2. Method

2.1. Data preparation

In order to meet the anticipated requirements, this study needs to collect a sufficient number of comic book collections for training and validation purposes. Comics come in many genres, including Superhero, Slice-of-Life, Humor, Non-fiction, Science-Fiction/Fantasy, and Horror [13]. Each genre of comics has significant differences in art style and coloring characteristics. Therefore, this study will select comic book collections of the same genre as the database, allowing for training with similar features. After extensive collection and filtering, this study will use the comic books images provided by the Kaggle [14] platform as the image dataset.

2.1.1. Data introduction

This dataset consists entirely of superhero comics from Marvel and DC. It contains 52,155 RGB images. The train and test datasets are divided into 86 classes with a 0.8 ratio, meaning the train dataset contains exactly 41,724 images, and the test dataset contains exactly 10,431 images. The author downloaded the books from different webpages. The original images had dimensions of 1988x3056. They were reduced to 288x432 using OpenCV to ensure that there is enough data while keeping the memory usage of the database manageable.

2.1.2. Preprocessing

To adapt the images to the Convolutional Neural Network (CNN) used in the European Conference on Computer Vision (ECCV) 16 model and enable batch processing, several data preprocessing steps are performed. The key steps in data preprocessing are as follows:

Image Reading: For each image, it is first read from the specified data directory. Python's PIL library (Python Imaging Library) is used to open the image, and it is converted to the RGB mode to ensure that all images have the same channel order.

Color Space Transformation: Subsequently, a color space transformation is applied. The original images are in RGB format, but this research utilizes the CIELAB (Commission Internationale de l'éclairage L* a* b*) color space, also known as the Lab color space. This color space is chosen due to its perceptual uniformity with respect to human color vision, where similar numerical transformations in these values correspond to approximately the same visually perceived adjustments. The conversion from RGB to Lab color space is performed using the color.rgb2lab function.

Channel Separation: Lab information comprises the L channel (luminance channel) and ab channels (color channels). In this step, Lab information is separated into L and ab channels for separate processing. The L channel is normalized to the [0, 1] range to enhance training stability.

Channel Concatenation and Reordering: Finally, the normalized L channel and ab channels are recombined into a Lab image. This Lab image is converted into tensor format and its channel order is adjusted to match the input requirements of deep learning models. During this process, the Lab information is transformed into a PyTorch tensor.

Ultimately, a data loader is created to load and batch-process the preprocessed data. The data loader allows us to provide data to the deep learning model in small batches for training and validation. The data loader shuffles the data in the dataset and batches it according to predefined batch size hyperparameters.

2.2. Federated learning

2.2.1. Introduction of federated learning

Federated Learning (FL) is a concept first introduced by Google in 2016. Federated Learning is a machine learning framework that enables the training of models across multiple decentralized devices or data sources while keeping the raw data localized. In essence, FL allows collaborative model training without the necessity of sharing sensitive or private data. Instead, model updates are computed locally on each device using locally available data and then aggregated to produce a global model. In traditional centralized machine learning shown in Figure 1, data is often transferred to a central server for model training, raising privacy concerns.

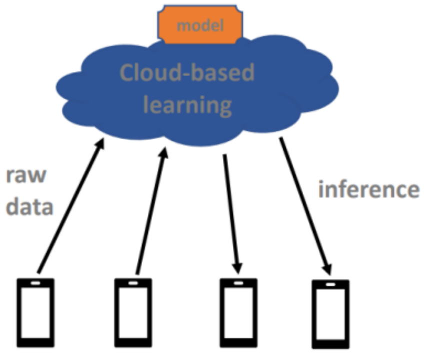

Figure 1. The procedure of traditional model training.

FL achieves this by only exchanging the models trained on the client during communication to ensure that the original data never leaves the device storing it, thus eliminating privacy concerns inherent in traditional machine learning. At the same time, FL leverages data distributed across various devices or servers shown in Figure 2, making a more comprehensive dataset available for model training. This increased data diversity can lead to more robust and accurate models. Moreover, it reduces the need for large-scale data transfers, saving bandwidth and computational resources.

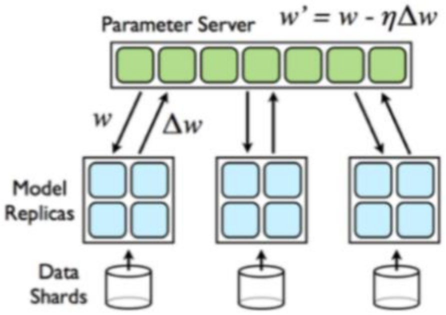

Figure 2. The process of communications between server and clients.

2.2.2. Introduction of algorithm

The Convolutional Neural Network (CNN) model used in this study is the ECCV16 model [12], designed for image colorization. The primary objective of this model is to map grayscale images (black and white images) to their corresponding color versions, thus achieving automatic image colorization. The ECCV16 model employs a deep CNN architecture, which comprises multiple convolutional and deconvolutional layers to facilitate the mapping from grayscale input images to color output images. The network's architectural design enables the effective capture of colorization information within the images.Each conv layer refers to a block of 2 or 3 repeated conv and ReLU layers, followed by a BatchNorm [15] layer. The net has no pool layers. All changes in resolution are achieved through spatial downsampling or upsampling between conv blocks. The specific implement of ECCV16 CNN is shown in Figure 3.

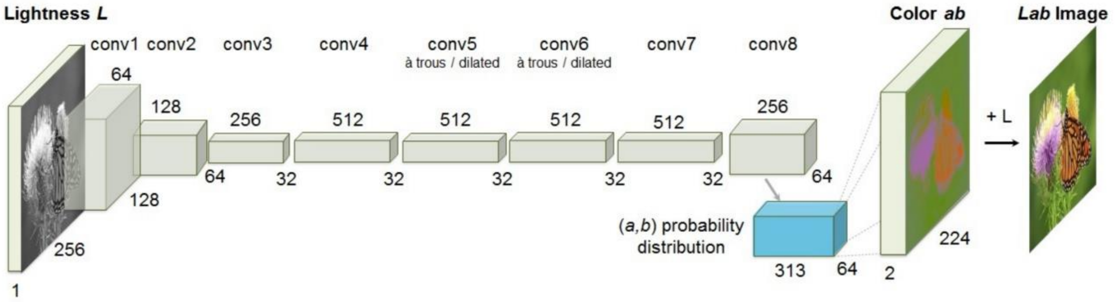

Figure 3. The structure of CNN used in the ECCV16 model [12].

2.2.3. Strategy

The objective of this experiment is to explore the potential application of Federated Learning in the field of comic book coloring. Therefore, the process simulates the concepts of distributed thinking, simulating synchronous distributed training, and model communication.

In this experiment, this study will be simulated with only two clients. The total dataset is divided into 4:4:2, representing the local data of User 1, the local data of User 2, and the test data, which is shown as Figure 4.

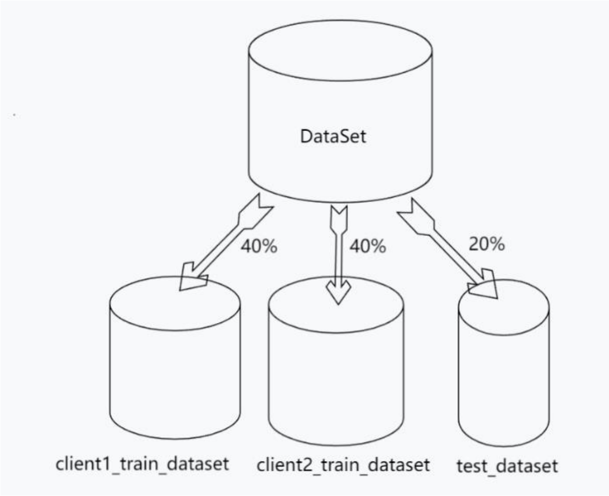

Figure 4. The procedure about the splitting of the dataset.

Next, this study utilizes model weights that have been trained on over a million of images using ECCV16 as the cloud-based global model. Although the global model at this point performs well in coloring real-world images, it struggles to match colors with corresponding features in 2D comics effectively. However, it can perform coloring on backgrounds and common objects in comics.

The core idea of Federated Learning involves iterative communication between the global model and client models. In each communication round, the following basic steps are performed:

Model Distribution: The global model parameters are duplicated twice, with each duplicate used as the initial model for two client devices.

Local Training: After receiving the global model, each client device independently trains the model using its local data. This is a local operation and does not require sending data back to the global server.

Model Update: Following local training, each client device generates a model update, representing changes in model parameters. This update typically includes gradient information and weight adjustments.

Model Aggregation: Client devices send their model updates back to the global server, where the server collects all model updates.

Model Averaging: The global server calculates the average of all model updates and applies this average to the global model.

Iterative Process: These steps are iteratively executed in multiple communication rounds until the global model converges to a satisfactory state.

In each round, the global model is shared with clients, and client devices use their local data to refine the model. The model updates from all clients are then aggregated and averaged to improve the global model. This process continues iteratively until the global model reaches the desired level of performance or convergence.

2.3. Implementation details

2.3.1. Torch

The entire experiment relies on the use of the torch library. The model construction utilizes the nn module from torch. The optimization function employs the SGD function from the optim module. Furthermore, extensive usage of the torch library is made for dataset processing and handling.

2.3.2. Loss function

The calculation of similarity between the predicted image and the original image (1-loss) is also an existing problem. During the training process, only the ab channels change, while the L channel remains constant. Therefore, this study only considers the loss of the ab channel. Initially, the Euclidean distance of the three channels to compute the loss for Lab's different colors will be considered, ultimately iterating through each pixel of the image to calculate the average loss.

However, such a calculation would overestimate the differences between highly saturated colors, and the results obtained would not effectively match human visual experience. Therefore, this study adopts the CIEDE2000 algorithm [16], resulting in a loss function tailored to the image colorization problem. After calculating the differences for each pixel, the list of differences is converted into a tensor, facilitating gradient calculations for parameter optimization to better fit the training data.

As this experiment aims to simulate federated learning, the color difference loss function consumes a significant amount of time due to the computation of a large number of pixels. To save time, this experiment employs a weighted cross-entropy loss function that allows for faster training.

3. Results and Discussion

3.1. The performance of the model

Directly using the pre-trained SIGGRAPH17 model for colorization, the results may only provide colors for common objects and backgrounds. It may have difficulty accurately coloring specific superhero features or details.

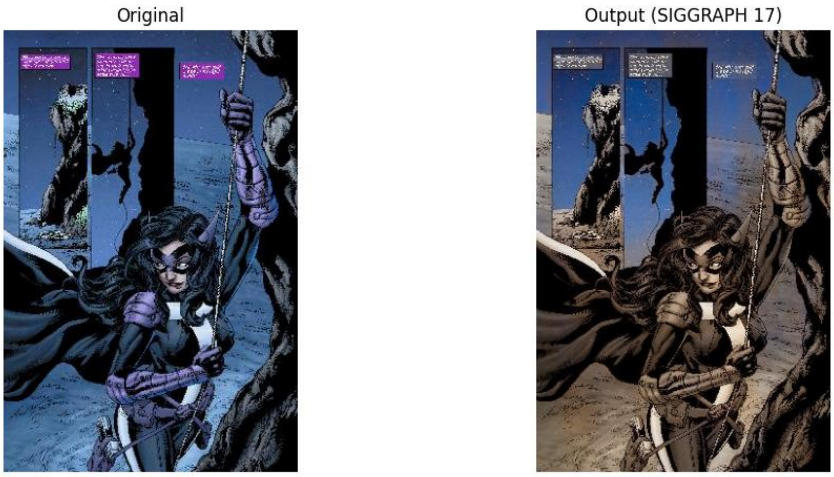

Figure 5. The original model’s performance on comic in Huntress.

Figure 5 and Figure 6 presents the model’s performance on comic and the loss over 50 epochs on the Huntress dataset, respectively. In this study, the comics within the Huntress folder of the training dataset, consisting of 100 images, were split into two equal subsets. Each subset underwent individual training utilizing the ECCV16 model, resulting in the creation of two new models. Subsequently, these two models were assessed using the test images from the Huntress dataset. The findings revealed that both models exhibited varying degrees of enhancement in their ability to color the images.

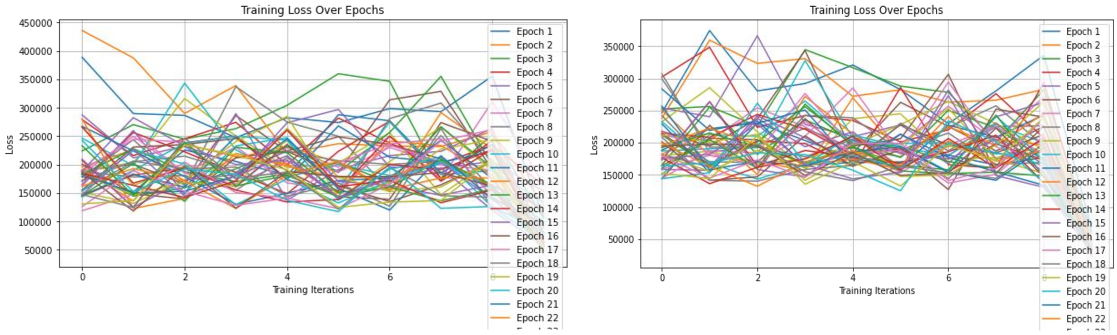

Figure 6. The loss over 50 epochs on the Huntress datasets.

Finally, the weights of the two newly trained models were averaged to obtain a single combined model. This combined model, tailored for Huntress, was once again tested on the testing dataset, and the result is as following Figure 7. Obviously, the prediction loss on previously unseen datasets is significantly lower, demonstrating the success of model weight averaging.

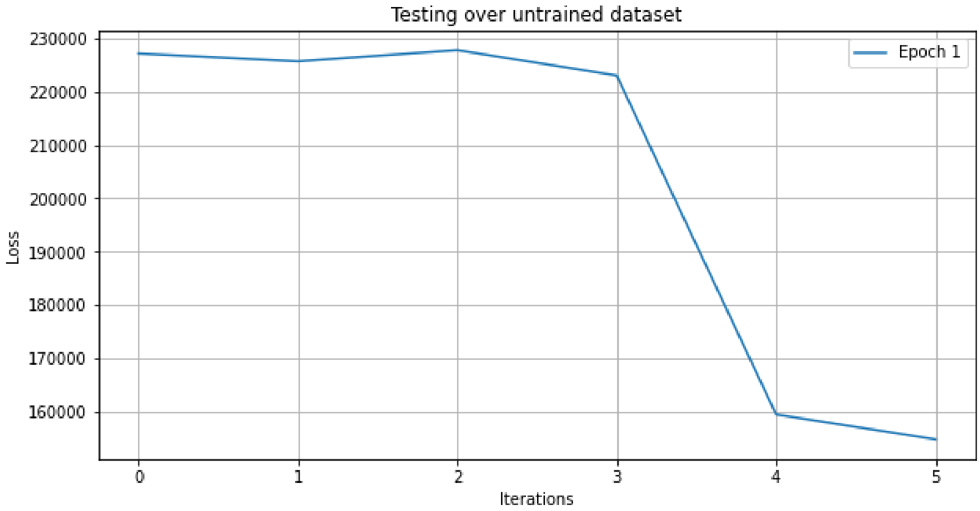

Figure 7. The loss testing over untrained dataset.

3.2. Discussion

This experiment, through the simulation of federated learning, demonstrates the feasibility of applying federated learning to the field of coloring in manga. It enables efficient colorization of grayscale images by avoiding potential privacy leaks through weight exchanges between models. However, there are areas in this experiment that can be improved, such as the quantity of data. With more time and better equipment, more refined models can be trained, resulting in better colorization outcomes. Additionally, the loss function in this experiment requires computing every pixel, which consumes a significant amount of time. In the future, if better loss functions tailored for image colorization can be developed, it will enhance the efficiency of the experiment.

4. Conclusion

The primary aim of this study is to demonstrate the applicability of federated learning to the field of comic coloring. This approach achieves personalized coloring while enhancing the efficiency of comic coloring, with a crucial focus on addressing privacy concerns, which is an unexplored direction in current federated learning applications. In comparison to traditional methods, this novel approach empowers comic creators to conduct local model training while safeguarding privacy, allowing them to capture their unique artistic styles and color preferences. In this research, the ECCV16 neural network model, originally crafted by Richard for training purposes, played a pivotal role. The study employed federated learning techniques to distribute and train model weights across segmented databases. As part of the model optimization process, a tailored loss function was devised to quantitatively measure color disparities between the original and colorized images. Empirical findings have substantiated the efficacy of this approach. Nevertheless, it is worth noting that there exist areas for enhancement in this study, notably with regards to the volume of data utilized. With more time and improved equipment, more refined models could be trained, resulting in enhanced coloring outcomes. Additionally, the current loss function requires pixel-level calculations, which are time-consuming. Future research efforts may focus on developing more suitable loss functions tailored for image coloring, which would enhance experimental efficiency.

References

[1] Wang N Chen G D and Tian Y 2022 Image Colorization Algorithm Based on Deep Learning. Symmetry 14(11) 2295

[2] Deng T 2023 Image Colorization Based on Deep Learning

[3] Fatima A Gowda A K Chandana C U Tauseef M D & Bhavitha Y 2023 Implementation of Driverless Car In 2023 International Conference on Advances in Electronics, Communication, Computing and Intelligent Information Systems (ICAECIS) (pp. 447-452). IEEE

[4] Gruetzemacher R 2023 The Power of Natural Language Processing. Harvard Business Review https://hbr. org/2022/04/the-power-of-natural-language-processing. Accessed, 6.

[5] Wu M and Chen L 2015 Image recognition based on deep learning In 2015 Chinese automation congress (CAC) pp 542-546 IEEE

[6] Wilamowski B M 2009 Neural network architectures and learning algorithms IEEE Industrial Electronics Magazine, 3(4), 56-63

[7] Kamina G 2023 Top 7 AI Comic Generator Tools to Try in 2023 https://ambcrypto.com/blog/top-7-ai-comic-generator-tools-to-try-in-2023/

[8] Petalica paint 2023 https://petalica.com/index_zh.html

[9] Github 2021 Style2paints https://github.com/lllyasviel/style2paints

[10] Matomo 2022https://matomo.org/blog/2022/08/data-privacy-in-business-risks-and-opportunities/

[11] Rieke N et al 2020 The future of digital health with federated learning NPJ digital medicine 3(1) 119

[12] Zhang R 2016 Colorful image colorization. In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, Proceedings, Part III 14 pp 649-666 Springer International Publishing

[13] Triton college library 2023 https://library.triton.edu/c.php?g=1107468&p=8074763

[14] Kaggle 2017 Comic books classification https://www.kaggle.com/datasets/cenkbircanoglu/comic-books-classification

[15] Ioffe S Szegedy C 2015 Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167

[16] Luo M R Cui G Rigg B 2001 The development of the CIE 2000 colour-difference formula: CIEDE2000. Color Research & Application: Endorsed by Inter-Society Color Council, The Colour Group (Great Britain), Canadian Society for Color, Color Science Association of Japan, Dutch Society for the Study of Color, The Swedish Colour Centre Foundation, Colour Society of Australia, Centre Français de la Couleur, 26(5), 340-350