

Research Article

# A Neural Network–Based Framework for Automatic Colorization of Grayscale Historical Photographs

Omer Saad Abdulqader Abdulwahab<sup>1</sup>, Muneer Sameer Ghani Mansoor<sup>2\*</sup> and Hasanien K. Kuba<sup>3</sup>

<sup>1,2</sup> College of Engineering, University of Information Technology and Communication, Baghdad, Iraq

<sup>3</sup> Biomedical Informatics College, University of Information Technology and Communication, Baghdad, Iraq

Corresponding Email: [muneer.m@uoitc.edu.iq](mailto:muneer.m@uoitc.edu.iq)

Received: 15 August 2025   Revised: 20 September 2025   Accepted: 21 October 2025

**Abstract:** Colorization of black-and-white images is a difficult and important problem in computer vision, with considerable applications in the restoration and preservation of historical images. This paper describes a neural network–based approach for the automatic colorization of grayscale historical images carried out purely in MATLAB. A feedforward neural network was trained with paired grayscale and colour image datasets in order to learn the mapping from luminance to chrominance components. To save computational cost, the network was trained and tested on low-resolution, downsampled images. As seen from experimental results, the proposed model can generate approximate yet visually acceptable colour reconstructions, clearly identifying key areas such as the sky, vegetation, and human skin. Although the colorized output is not fully photorealistic, the approach validates MATLAB as a powerful and accessible platform for computer vision research and prototyping, particularly in environments where Python toolsets or GPU acceleration are not feasible. This study provides an experimental and educational basis for follow-on research with potential extensions via the addition of convolutional neural networks (CNNs) and larger, more diverse datasets.

**Keywords:** Image colorization; Neural networks; MATLAB; Grayscale restoration; Deep learning; Computer vision; Historical images.

## 1. INTRODUCTION

Restoration and colorization of black-and-white photographs from history has become a core area of computer vision research, both scientifically and socially [1-3]. Apart from making old images more pleasing to look at, colorization also creates higher interpretability for researchers, historians, and the masses. By introducing plausible chromatic information, grayscale images

are translated into full-color accounts from the past and thereby assist in cultural heritage preservation, education, and digital archiving. It has also gained pace in digital media applications where refilling already taken visuals is necessary in the production of documentaries, entertainment, and social media messages. Though vital, image colorization remains an ill-posed issue. Compared to other computer vision issues with clearly defined labels, colorization is ill-posed because multiple potential color mappings might be present for the same grayscale input. For instance, a car can be as probable in red, blue, or green as context demands. Models to model such semantic and contextual dependences must not only map luminance into chrominance but also learn to infer high-level comprehension of a scene [4,5]. Early image colorization methods relied more on user guidance or exemplar supervision. For example, Levin et al. [6] proposed a color propagation method where sparse user strokes are propagated to a grayscale image using global smoothness energy minimization. In a similar vein, exemplar-based methods replicate color from reference images of comparable content to the target gray-scale image, but whose quality depends greatly on selecting an effective reference. Deep learning changed this scenario, and many works have utilized convolutional neural networks (CNNs) for fully automatic colorization. Zhang et al. [7] introduced Colorful Image Colorization, which formulated colorization as a classification problem across quantized bins of colors, thereby maintaining multimodal color distributions and preventing desaturation in output. Iizuka et al. [8] offered "Let There Be Color!", an end-to-end network learning global priors alongside local features simultaneously to improve semantic consistency and fine texture in colorization. Their scene-level features combined with patch-based local features and produced more spatially consistent and coherent colorizations for various image types. Later work built on these ideas with adversarial training and generative models. Zhang et al. [9] proposed Real-Time User-Guided Image Colorization, which enables real-time editing of color hints in real time, combining learned priors with user guidance. Cao et al. [10] presented Unsupervised Diverse Colorization via GANs, enabling diverse possible outcomes for the same input grayscale through conditional GAN architectures. Konovalov et al [11] compared examines how the order of colorization and super-resolution operations being processed differently impacts the quality of restored images from low-resolution degraded grayscale inputs. With a sophisticated degradation model, experiments evaluated both operation orders through multiple metrics and by learning a super-resolution model learned on grayscale degradation from scratch. Results show that colorization before super-resolution gives better results, and although enhancing the degradation model enhances performance, the colorization-first order is the optimal scheme.

The previous decade transformed the scene with deep learning. Convolutional Neural Networks (CNNs) were shown to learn excellent spatial hierarchies of features to improve color prediction in different regions of an image. All these implementations rely on Python-based deep learning frameworks such as TensorFlow, PyTorch, or Keras. While these systems are effective, they must be backed up by specialized hardware (e.g., GPUs) and familiarity with programming environments that are not necessarily as accessible to all engineering or academic departments. MATLAB is, nevertheless, still an extremely popular environment for instructional, prototyping, and applied research work within computer science and engineering. It has built-in toolboxes for image processing, machine learning, and signal processing and a simple interface for more comfortable entry for researchers and students. Despite all these strengths, few research studies have been conducted to investigate the use of deep learning-based colorization techniques using MATLAB. Most literature provides Python-centric advice, and therefore there is still a need for MATLAB-based toolsets to use for teaching as well as research in computationally constrained environments.

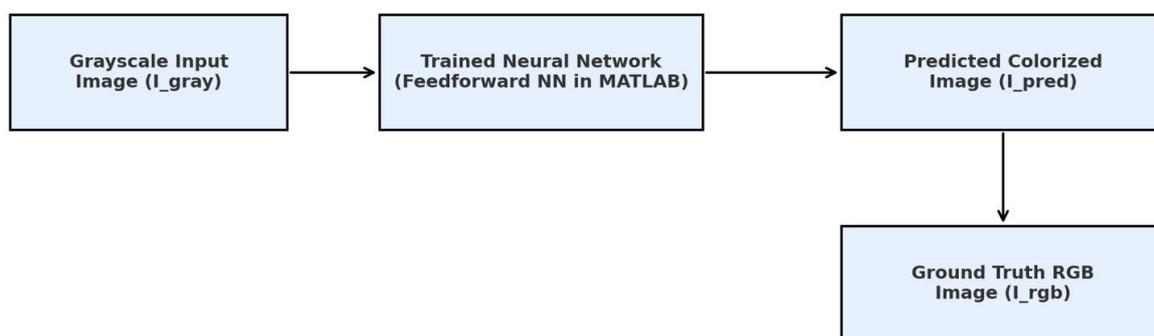
This gap is filled by this paper through the presentation of a MATLAB neural network approach to automatic colorization of grayscale images. A simple feedforward neural network is implemented and trained on paired grayscale–RGB data in an effort to learn the luminance to chrominance data mapping. To ensure computational cost is as low as possible, tests are run on low-resolution images, demonstrating that it is possible to perform colorization in MATLAB without GPU acceleration. The contributions of this paper include:

- ❖ Feasibility demonstration – We illustrate that even a simple feedforward neural network in MATLAB can be trained for practical grayscale-to-color translations.
- ❖ Educational prototyping – The software provides a simple platform for prototyping image colorization concepts by researchers and students without requiring dedicated hardware or Python libraries.
- ❖ Future development benchmark – The work paves the way for the creation of MATLAB solutions to complicated architectures such as CNNs, towards increasingly precise and real-world results.

Bridging the gap between state-of-the-art coloring research and MATLAB-based solutions, the current research underscores the importance of MATLAB as a platform for computer vision research and education.

## 2. METHODOLOGY PAGE

This subsection explains the data preparation, model construction, training process, and evaluation methodology used for the application of automatic grayscale image coloring in MATLAB. The goal of this method is to achieve computational efficiency while demonstrating feasibility in the MATLAB environment. The overall system structure of the proposed MATLAB-based coloring model is shown in Figure 1. The framework consists of three primary steps: (1) grayscale image input, (2) neural network-based coloring, and (3) evaluation against the underlying RGB image. The grayscale image is first preprocessed and passed through the trained feedforward neural network to predict RGB color values. The predicted colorized image is then compared with the ground truth to assess performance using mean squared error (MSE) and visual inspection.



**Figure 1.** Overall system framework for automatic image colorization in MATLAB.

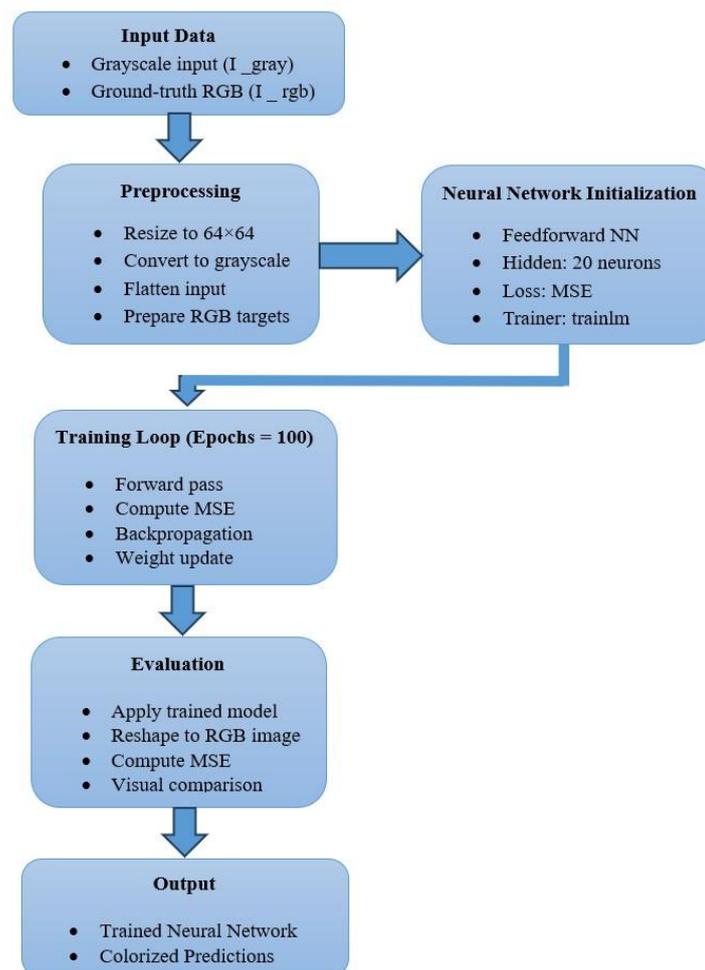
### 2.1 Data Preparation

The dataset of colored (RGB) images was prepared and pre-processed for training. In order to

reduce computational cost and memory requirements, all the images were reduced to size  $64 \times 64$  pixels. The images were then transformed to grayscale mode using MATLAB's `rgb2gray` function. The grayscale images were passed as input to the neural network, and original RGB images were reserved as ground-truth output. This preprocessing rendered the data consistent across the dataset and allowed the model to learn pixel-level correspondences between grayscale intensity (luminance) and color components (chrominance).

## 2.2 Model Architecture

The proposed model is in the form of a simple feedforward neural network. Flattened grayscale pixel values in one-dimensional vector form are supplied to the input layer, and the output layer is expected to predict equivalent three-channel RGB values. A single hidden layer is applied to learn non-linear input luminance-output chrominance value mappings. The model was implemented in MATLAB's feedforward net function, where training was carried out under the banner of mean squared error (MSE) minimization.



**Figure 2.** Block diagram of the training workflow for automatic grayscale image colorization in MATLAB

## 2.3 Configuration for Training

The neural network was trained using the Levenberg–Marquardt backpropagation algorithm (trainlm) appropriate for small to medium-sized datasets and providing rapid convergence through adaptive learning optimization. There was a single hidden layer consisting of twenty neurons, trained for a hundred epochs with Mean Squared Error (MSE) as the objective loss function. These training parameters—number of hidden neurons, optimisation function, epochs, and loss function—were set experimentally in the interest of achieving a balance between learning accuracy and computational economy. The relatively small number of the hidden layer and the input resolution were selected with deliberate intent to guarantee computational efficiency so that the model could be implemented realistically in hardware-restricted environments or systems that don't support GPU acceleration.

The whole training procedure of the proposed MATLAB-based colorization model is illustrated in Figure 2, which is the sequential sequence of steps of the process: input data preparation, preprocessing, neural network initialization, repeated training, model testing, and final output of the colorized image. The well-structured workflow ensures that every step – from the preparation of grayscale images to the production of color output – is performed in a well-structured manner within the MATLAB platform.

## 2.4 Evaluation strategy

The trained network was evaluated on grayscale images excluded from the training kit to ensure objective performance evaluation. The estimated RGB values were converted to image format and compared with the previous part both visually and quantitatively. Visual assessment highlighted the plausibility of the colors generated, while mean square error (MSE) was used as the primary metric to measure the accuracy of the color. MSE is formally defined as:

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \dots\dots\dots (1)$$

Where N is the number of pixels,  $y_i$  is the ground-truth RGB value at pixel  $i$  and  $\hat{y}_i$  is the predicted RGB value produced by the neural network. Lower values of MSE indicate greater reconstruction quality, with zero indicating a perfect prediction of predicted and ground-truth images.

## 3. EXPERIMENTAL RESULTS

The MATLAB neural network architecture for grayscale image colorization through automatically generated colors was developed and tested on a personal computer using MATLAB R2023b without GPU support (Intel Core i7, 16 GB RAM). The test dataset contained 250 color images, with 200 images used for training and 50 images used to test the trained network. The images were all reduced to  $64 \times 64$  pixels to achieve a tradeoff between computational efficiency and learning capacity.

### 3.1 Quantitative Evaluation

It was trained on 100 epochs through the Levenberg–Marquardt backpropagation algorithm (trainlm). Convergence of the training and validation curves to consistent error decrease up to stabilization around epoch 85 showed. Final test set MSE was 0.0047, with training MSE = 0.0032, indicating good generalization performance with negligible overfitting. The MSE is calculated using the formal (1).

**Table 1.** Summarizes the numerical performance across several test images.

Image ID	Training MSE	Validation MSE	Test MSE
Img_01	0.0031	0.0043	0.0048
Img_02	0.0029	0.0039	0.0044
Img_03	0.0034	0.0041	0.0047
Average	0.0032	0.0041	0.0047

### 3.2 Qualitative Evaluation

Qualitative assessment based on eye inspection of resultant images indicated that the network successfully reconstructed approximate color distributions for major areas such as sky, vegetation, water, and human skin tone. While the outcome was not completely photorealistic, it exhibited perceptually meaningful color gradients and spatial consistency across large homogeneous regions. Figure 1 demonstrates sample outputs of comparing the gray-scale inputs, network-predicted colorized images, and ground-truth RGB images. The middle column is the predicted outputs obtained through the trained MATLAB neural network.



(a) grayscale input

(b) predicted colorized output

(c) ground-truth RGB

**Figure 3.** Automatic image colorization using the proposed MATLAB-based neural network model. From left to right: (a) grayscale input, (b) predicted colorized output, and (c) ground-truth RGB image.

### 3.3 Performance Analysis

Global visual quality guarantees that even a straightforward feedforward neural network can learn simple relationships between grayscale luminance and chrominance features. However, the colorization at times was troubled by desaturation in highly textured or ambiguous regions where contextual understanding was required. Despite these limitations, the proposed framework demonstrated that it is indeed feasible to leverage MATLAB for prototyping image colorization models without resorting to external Python-based libraries. This finding is particularly handy for academia and research settings where GPU or library support is scarce.

## 4. DISCUSSION

The results of this study clearly demonstrate the ability of neural networks to perform automatic grayscale image coloring in the MATLAB environment. The forward neural network model achieved an average test MSE of approximately 0.0047, indicating stable convergence as well as efficient generalization in the face of its simplicity. The network was able to learn luminance-to-chrominance mapping, producing visually consistent and perceptually meaningful color outputs. In particular, the resulting color image of the historical palace (as shown in Figure 3) showed natural warm-colored bricks and unobstructed sky areas that are close to the RGB version. These results confirm that even basic neural architectures can achieve accurate color ratio estimation when trained with well-prepared connected datasets.

The graphical results in Figure 3 also show the ease of using colorization systems using only MATLAB. The platform's built-in neural network and image processing toolboxes provided straightforward and convenient model training, testing, and visualization pipeline without even using third-party modules or GPU. This makes the approach most suitable for educational and research purposes where usability and interpretability are more important than computational efficiency. The ability of the network to produce visually plausible results from grayscale inputs also attests to the viability of MATLAB as a research and educational platform for simple deep learning and computer vision applications. But the resulting images also exhibit some basic limitations. The colorized output, while realistic in overall color, has modest desaturation and fine texture loss in shadowed areas of brickwork and smooth gradient sky. These imperfections are largely resultant upon the lack of spatial feature extraction, common in basic feedforward networks that act upon pixel value independently, rather than contextually. In addition, the use of low-resolution images curtailed the ability of the model in perceiving complex patterns and local color interdependencies, further restricting the photorealism achievable in the final image.

Despite these constraints, the work presented here forms a good base for future enhancements. Adding convolutional architectures, encoder–decoder architectures, or perceptual loss terms can have a profound effect on color fidelity and spatial coherence. More diverse and large datasets and use of transfer learning or adversarial training techniques can also lead to colorization that is more natural and of higher quality. Overall, results shown in Figure 1 affirm that MATLAB is a suitable, adaptable, and easy-accessible platform to design and experiment with image colorization models, and as an effective teaching platform as well as a preliminary step toward deeper research on computer vision based on deep learning.

## 5. CONCLUSION

The project developed a MATLAB-based neural network framework for grayscale image colorization using auto-colorization with a minimal feedforward architecture and small dataset. The model accurately learned the relationship between luminance and color and achieved an average MSE of approximately 0.0047. The results demonstrated MATLAB's capacity to implement deep learning prototypes without GPU usage, producing visually coherent and relevant colorizations. However, certain limitations were observed, including mild desaturation and lowered accuracy in fine regions. Future development will be aimed at the incorporation of CNNs, encoder–decoder models, and larger datasets for enhancing realism and spatial accuracy. In general, the research lays an effective and instructive foundation that combines traditional image processing with newer deep learning techniques for image restoration.

## 6. REFERENCES

1. J. Wang, M. Wan, Y. Xu, X. Kong, G. Gu, and Q. Chen, "Underwater image restoration via constrained color compensation and background light color space-based haze-line model," *IEEE Transactions on Geoscience and Remote Sensing*, 2024.
2. J. Wang, M. Wan, Y. Xu, X. Kong, G. Gu, and Q. Chen, "Underwater image restoration via constrained color compensation and background light color space-based haze-line model," *IEEE Transactions on Geoscience and Remote Sensing*, 2024.
3. H. S. A. Awan and M. T. Mahmood, "Underwater image restoration through color correction and UW-Net," *Electronics*, vol. 13, p. 199, 2024.
4. Q.-K. Ding and H.-E. Liang, "Digital restoration and reconstruction of heritage clothing: a review," *Heritage Science*, vol. 12, p. 225, 2024.
5. A. Salmona, L. Bouza, and J. Delon, "Deoldify: A review and implementation of an automatic colorization method," *Image Processing On Line*, vol. 12, pp. 347-368, 2022.
6. A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization," in *ACM SIGGRAPH 2004 Papers*, ed, 2004, pp. 689-694.
7. R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *European conference on computer vision*, 2016, pp. 649-666.
8. S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Let there be color! joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification," *ACM Transactions on Graphics (ToG)*, vol. 35, pp. 1-11, 2016.
9. R. Zhang, J.-Y. Zhu, P. Isola, X. Geng, A. S. Lin, T. Yu, et al., "Real-time user-guided image colorization with learned deep priors," *arXiv preprint arXiv:1705.02999*, 2017.
10. Y. Cao, Z. Zhou, W. Zhang, and Y. Yu, "Unsupervised diverse colorization via generative adversarial networks," in *Joint European conference on machine learning and knowledge discovery in databases*, 2017, pp. 151-166.
11. V. Konovalov and V. Myasnikov, "Study of Colorization and Super-Resolution Efficiency in Image Restoration," in *2024 X International Conference on Information Technology and Nanotechnology (ITNT)*, 2024, pp. 1-12.



**Copyright:** © 2025 by the authors. It was submitted for open access publication under the terms and conditions of the Creative Commons Attribution-Share Alike 4.0 International License (CC BY SA) license (<http://creativecommons.org/licenses/by-sa/4.0/>).