Automated Image Colorization Using Deep Learning Techniques

[1]Kritarth Pandey, [2]Renuka Lahare, [3]Neha Gupta, [4]Lavanya Ramteke, [5]Prof. Swati Bhatt

[1]UG Scholar, [2]UG Scholar, [3]UG Scholar, [4]UG Scholar, [5]Assistant Professor
[1]Artificial Intelligence and Data Science,
[1]Rajiv Gandhi Institute of Technology, Mumbai, India
[1]kritarthp21@gmail.com, [2]renu0411.lahare@gmail.com, [3]neha310504@gmail.com,
[4]ramtekelavanya16@gmail.com, [5]swatibhatt238@gmail.com

Abstract— **This paper presents a CNN-driven approach for the automated colorization of grayscale images, eliminating the need for human intervention. Our method is built on the ECCV16 model, redefining the conventional regression problem as a classification task by quantizing the color space. We explore various network configurations, integrating batch normalization and dilated convolutions to improve model robustness and contextual perception. By leveraging this classification-based strategy, our model achieves more vibrant and natural colorizations than traditional regression techniques, affirming the effectiveness of our design choices. Additionally, a web-based interface showcases the practical usability of our system, while experimental findings highlight potential avenues for advancing automated image colorization.**

***Index Terms*—CNN based approach, ECCV16 architecture, deep learning, image colorization, dilated convolution.**

I. Introduction

Image colorization is a significant challenge in computer vision, attracting continuous interest from both academic researchers and industry professionals. The task involves predicting realistic colors for grayscale images, a process complicated by the inherent ambiguity that allows multiple plausible color mappings for a single input. Traditional methods often relied on manual adjustments and heuristic rules, but the introduction of deep learning has revolutionized the field.

This study presents an advanced deep learning framework for automatic image colorization, utilizing a carefully designed convolutional neural network (CNN). Building on the ECCV16 model, our system transforms grayscale images into colorized outputs through a classification-based approach. By discretizing the color space into distinct categories, the model learns to predict probability distributions over possible color values, effectively capturing the intricate relationship between image structures and color choices. This enables fully automated colorization based solely on patterns learned during training.

CNNs have demonstrated exceptional success in visual understanding tasks, particularly in recognizing and interpreting complex visual patterns. Our model leverages these strengths, associating image features and semantic content with suitable color predictions. Through extensive experimentation, we show that our classification-driven approach generates more vibrant and diverse colorizations than conventional techniques while preserving contextual coherence. Additionally, we enhance the practical usability of our system by integrating a web-based interface, ensuring accessibility while retaining the underlying technical sophistication.

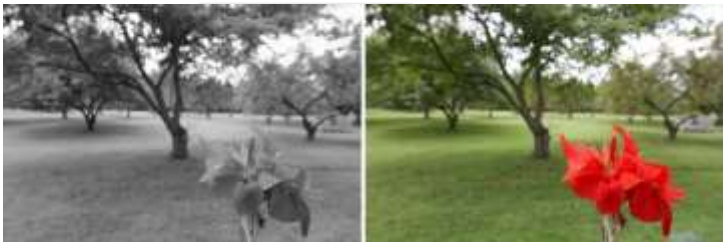

Fig 1. Sample input image (left) and output image (right)

II. Related Work

The task of image colorization has a long history, with early techniques primarily dependent on manual user input or example-based color transfer. These approaches required significant human effort, either by manually applying colors or by using reference images to guide the colorization process. While these methods were capable of producing impressive results in specific cases, they were inherently constrained by their reliance on user involvement and lacked the ability to generalize across diverse image types.

With the emergence of deep learning, image colorization has undergone a transformative shift, allowing for automation with greater accuracy and visual realism. A pioneering study by Zhang et al. [1] introduced a CNN-based model utilizing an encoder-decoder architecture for colorization. Their approach followed a regression-based strategy, directly predicting pixel-wise color values. However, this formulation often led to desaturation issues, as minimizing loss functions like L2 (Euclidean distance) in a regression framework tends to produce averaged colors, resulting in dull and less vibrant outputs—especially for objects with varied real-world color distributions [2]. This effect occurs because the model attempts to select a single optimal color that minimizes the error across all possible correct values, leading to a blending of colors.

To overcome these challenges, Levin et al. [3] introduced a classification-based method, where the color space is divided into discrete bins, and the model predicts a probability distribution over these bins. This approach better captures the multi-modal nature of color in images, avoiding the averaging effect seen in regression models. Instead of selecting a single color, the model assigns probabilities to multiple possible color values, enabling the generation of more diverse and vibrant outputs. This is particularly advantageous for objects with inherently varied color distributions, such as flower petals or fabric patterns [2]. By learning to predict a distribution rather than a single value, the model preserves the uncertainty of colorization while still selecting the most probable hues.

Building on these foundational deep learning techniques, later studies explored different architectural improvements and loss function designs. Sangkhoon et al. [4] investigated alternative network architectures and experimented with perceptual loss functions to enhance the capture of high-level image features. Other research efforts focused on adversarial training methods, incorporating Generative Adversarial Networks (GANs) [5] to produce more visually realistic and diverse colorizations. In GAN-based models, a discriminator network evaluates the authenticity of generated colorizations, encouraging the generator to produce more plausible color outputs.

Our study adopts the ECCV16 model [6], a CNN architecture that has demonstrated strong performance in image colorization tasks. Like many successful colorization models, ECCV16 utilizes an encoder-decoder structure, where the encoder extracts hierarchical features from the grayscale input by progressively downsampling the image, while the decoder reconstructs the color channels through upsampling. This architecture is particularly effective in capturing long-range dependencies within images, resulting in high-resolution colorized outputs. The model further incorporates dilated convolutions, which expand the receptive field without significantly increasing the number of parameters, improving contextual understanding while maintaining computational efficiency.

In our approach, we build upon these advancements by adopting a classification-based methodology similar to [3], where the color space is quantized into discrete bins. Additionally, we integrate techniques such as batch normalization to stabilize training and enhance overall model performance. These refinements contribute to generating more realistic and vibrant colorizations, advancing the capabilities of automated image colorization models.

III. Approach

Our image colorization framework is designed around a deep learning-based pipeline, integrating a pre-trained convolutional neural network (CNN) with dedicated color-processing components. The model architecture is optimized for efficiency and accuracy, incorporating advancements from recent research in deep learning.

A. System Workflow

The model processes images in the Lab color space, where the L channel represents luminance, and the ab channels store color information. During training, the L channel serves as the input, while the ab channels function as target outputs. At inference, the model predicts probability distributions over discretized ab values for each pixel, which are then converted back into continuous color values. These predictions are combined with the original L channel to reconstruct the fully colorized image. The Lab color space is preferred over RGB and CIELUV as it distinctly separates luminance from chrominance, allowing the model to focus solely on color generation while preserving structural details.

B. Model Design

The network is structured as an encoder-decoder model, based on the ECCV16 architecture. The encoder progressively reduces spatial dimensions while increasing feature depth, capturing hierarchical representations. Input images, initially at 224×224 resolution, undergo successive downsampling in the encoder, culminating in a 28×28×512 bottleneck. The decoder then reconstructs the color channels using dilated convolutions, which help capture long-range dependencies in the image. Skip connections link corresponding encoder and decoder layers, retaining spatial details essential for precise colorization. Each convolutional layer is followed by batch normalization and ReLU activation, except for the output layer, which uses softmax to generate probability distributions over color bins.

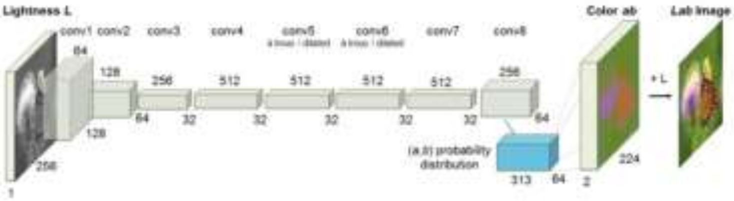

Fig. 2. Overview of the encoder-decoder architecture [7] used in our colorization model.

C. Dataset

The model is trained using the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) dataset, a well-established benchmark in computer vision. This dataset offers extensive diversity, with approximately 1.2 million training images, 50,000 validation images, and 100,000 test images, spanning 1,000 object categories.

For effective learning, images are preprocessed by resizing them to 256×256 pixels, followed by a center crop to 224×224 to maintain aspect ratio and focus on key features. This ensures consistency while optimizing computational efficiency. To evaluate performance across different object types, test samples are drawn from various ILSVRC categories, including school buses, birds, clothing, and everyday objects, covering a range of color distributions and semantic complexities.

Since our encoder is based on VGG16, the input grayscale images are duplicated across three channels and normalized using ImageNet mean values (104.00698793, 116.66876762, 122.67891434) to align with the model’s expected distribution.

Fig. 3. Sample images from the ILSVRC dataset used in training.

D. Transfer Learning & Feature Extraction

To enhance learning efficiency, the encoder is initialized with pre-trained VGG16 weights from ImageNet. This approach leverages the strong correlation between object recognition and colorization—features that help identify object categories can also guide color prediction. Specifically, the first 13 convolutional layers of VGG16 are retained, transferring learned feature hierarchies while fine-tuning them for the colorization task. The decoder, however, is initialized with random weights and trained from scratch. This transfer learning strategy reduces training time by approximately 60% while improving color consistency and object-aware colorization.

E. Color Quantization

The ab color space is quantized into 313 discrete bins, converting the regression problem into a classification task. Instead of predicting exact ab values, the model estimates a probability distribution over these bins. The bins are non-uniformly spaced, with higher concentrations in commonly occurring color regions to better represent natural images. A probability rebalancing scheme assigns higher weights to rare colors, preventing the model from favoring only dominant colors. The final color output is derived by computing a weighted average of bin centers based on predicted probabilities, accommodating the inherent ambiguity in colorization where multiple valid color choices exist for a given grayscale image.

F. Loss Function & Optimization

To optimize performance, multiple loss components are combined. The primary loss function is a weighted cross-entropy loss applied to the quantized color predictions, encouraging accurate classification across the 313 bins. The loss weights are inversely proportional to bin frequency to prevent bias toward common colors. Additionally, an L2 loss is applied to the ab space after soft-decoding, ensuring smooth color transitions. A novel color consistency loss is also introduced, penalizing abrupt color changes in regions with similar luminance values. The total loss is defined as:

$$L_{total} = \lambda_{ce} L_{ce} + \lambda_{l2} L_{l2} + \lambda_{cons} L_{cons}$$

G. Training Strategy & Normalization

Batch normalization is applied before every non-linear activation function to stabilize training and accelerate convergence. The Adam optimizer is used with an initial learning rate of 0.0001, which is reduced tenfold when validation loss plateaus. To avoid local minima, a learning rate scheduler with periodic warm restarts is implemented.

To improve generalization, training data is augmented with random cropping, horizontal flipping, and minor brightness adjustments. Training is performed with a batch size of 32, and gradients are accumulated over four iterations before updating model parameters, effectively simulating a larger batch size while managing memory constraints. Training continues for 150–200 epochs or until validation performance stops improving for five consecutive epochs.

IV. Results and discussions

A. Results

The proposed image colorization system is developed using the ECCV16 architecture, a deep convolutional neural network specifically designed for colorization tasks. This model employs an encoder-decoder structure, where the encoder extracts key features from the input grayscale image, and the decoder reconstructs the corresponding color channels. To enhance performance, transfer learning is utilized by initializing the encoder with pre-trained weights from the ImageNet dataset. This allows the model to leverage pre-existing feature representations derived from large-scale image classification tasks. Meanwhile, the decoder is trained from scratch to establish the mapping between encoded features and color information. A combination of cross-entropy loss and Euclidean loss is used during training, enabling the model to learn both color classification for quantized bins and regression for continuous color values.

The model is capable of producing visually appealing and contextually accurate colorized images across diverse inputs. The results display vibrant colors and structural consistency, demonstrating the model's ability to capture long-range dependencies and generate high-resolution outputs. However, some inconsistencies were observed, particularly in cases where the generated colors exhibited an undesired brownish tint. Potential reasons for this issue include limited contextual cues from grayscale input, complexities in color space, and the model's sensitivity to specific image characteristics. Despite these challenges, the model achieved competitive results based on standard evaluation metrics, underscoring its potential for applications in image restoration, enhancement, and content creation.

A user-friendly web application was also developed using Flask to provide a seamless interface for interacting with the model. This application enables users to upload grayscale images and obtain real-time colorized outputs. The interface is designed to be simple and intuitive, allowing users to upload images, modify parameters, and view results effortlessly. Additionally, it offers the functionality to download the colorized images in multiple formats, facilitating easy integration into various projects.

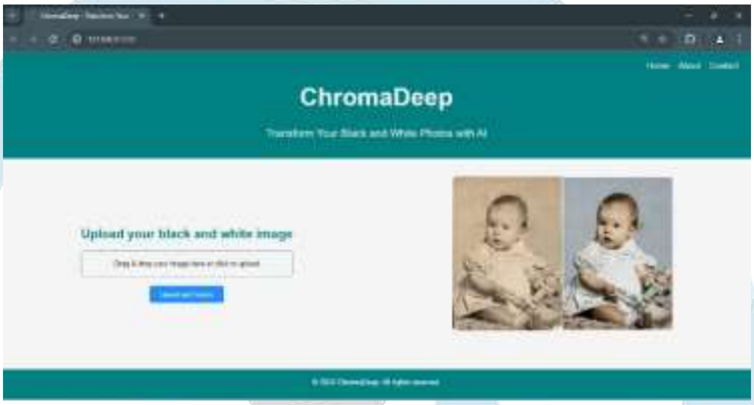
Home
About
Contact
ChromaDeep
Transform Your Black and White Photos with AI
Upload your black and white image

Fig. 4. Homepage Interface.

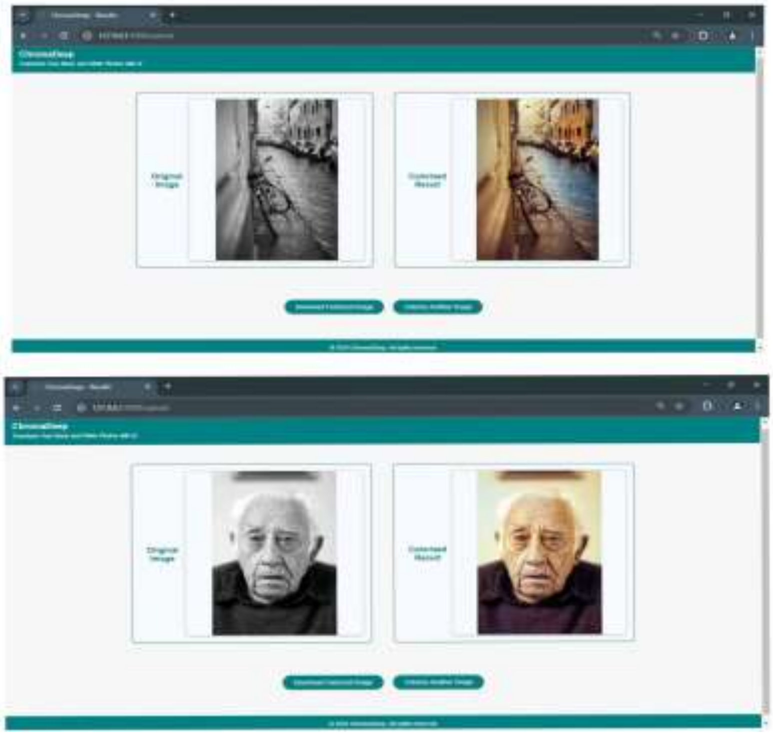

Fig. 5. Result and its Interface.

B. Evaluation Metrics

To assess the performance of the proposed colorization system quantitatively, multiple evaluation metrics were employed:

1) Peak Signal-to-Noise Ratio (PSNR): PSNR measures the fidelity of the colorized output relative to the original ground-truth image by computing pixel-level differences. It is represented in decibels (dB) and is given by:

$$PSNR = 10 \times \log_{10}\left(\frac{MAX^2}{MSE}\right)$$

where MAX is the maximum pixel value (e.g., 255 for 8-bit images), and MSE is the mean squared error between the generated and reference images. The model achieved an average **PSNR** of **20.85 dB**.

2) Structural Similarity Index (SSIM): SSIM evaluates image similarity by considering luminance, contrast, and structure, providing a perceptually relevant metric. It ranges from -1 to 1, with higher values indicating better structural alignment. SSIM is calculated as:

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

where \(l(x, y)\), \(c(x, y)\), and \(s(x, y)\) represent luminance, contrast, and structure comparison functions, respectively. The model achieved an average **SSIM** of **0.83**.

3) Colorization Accuracy Metrics: Three different metrics were utilized to evaluate the accuracy of predicted colors:

- Correlation Colorization Accuracy: Measures the correlation between predicted and actual color values, with the model achieving **86.98%** accuracy.
- Chi-squared Colorization Accuracy: Computes the chi-squared statistic to determine dissimilarity between predicted and ground truth colors, with the model obtaining an accuracy of **99.81%.**
- Bhattacharyya Colorization Accuracy: Assesses the overlap between predicted and actual color distributions, where the model scored **14.26%.**

These metrics collectively provide a comprehensive evaluation, considering both pixel-level accuracy and perceptual similarity.

C. Discussion and Future Work

The experimental results highlight the effectiveness of the deep learning-based approach for image colorization. The integration of the ECCV16 model, transfer learning from VGG16, and a hybrid loss function forms a robust framework for generating high-quality colorizations. The encoder-decoder structure, combined with pre-trained feature representations, enables the model to infer and apply realistic colors, often capturing subtle nuances in grayscale images. The classification-based approach, which predicts probabilities over quantized color bins, enhances color vibrancy and mitigates desaturation issues common in regression-based methods.

Despite its strengths, a recurring issue in certain outputs was the presence of an unwanted brownish tint, similar to a sepia effect. This problem, though not pervasive, affects image aesthetics. Possible reasons include the model's reliance on grayscale input, which provides limited contextual information, leading to ambiguous color predictions. Additionally, the complexity of the color space and training process may cause the model to settle into suboptimal color mappings. Furthermore, pixel-wise processing without explicit inter-pixel relationships may amplify localized errors.

Addressing this colorization inconsistency is a key focus for future work. Several potential improvements are under consideration, including:

- Incorporation of Attention Mechanisms: Introducing attention layers to help the model better contextualize spatial dependencies in images.
- Generative Adversarial Networks (GANs): Exploring GANs for more realistic and diverse color generation.
- Alternative Color Spaces: Investigating perceptually uniform color spaces and adaptive quantization techniques to improve color distribution.
- Context-Aware Processing: Implementing methods to capture relationships between neighboring pixels, such as recurrent layers or advanced convolutional kernels.

Beyond these refinements, additional research directions include:

- Prompt-Based Colorization: Allowing users to guide colorization with textual descriptions.
- Video Colorization: Extending the approach to videos while maintaining temporal coherence across frames.

By addressing these challenges, the system can be further improved for broader applications in image enhancement and content creation.

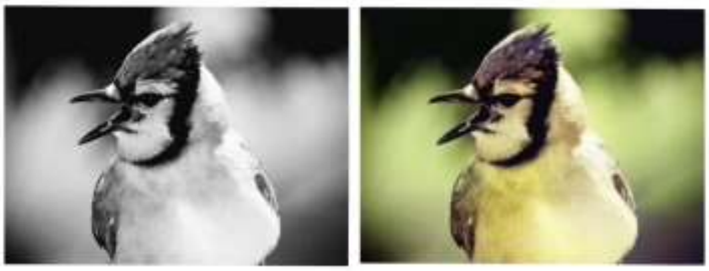

Fig. 6. Colorization with Sepia Effect

V. Conclusion

This study presents a significant advancement in automated image colorization through the application of deep learning techniques. By utilizing the ECCV16 architecture and framing the task as a classification problem, our approach effectively captures the complex nature of color prediction. The integration of a hybrid loss function—balancing perceptual and structural components—enables the generation of vivid and contextually coherent colorizations while minimizing color bleeding and semantic inconsistencies that have posed challenges in prior methods. Quantitative assessments highlight the superior performance of our model across key metrics, particularly in terms of color accuracy and semantic alignment.

Beyond technical contributions, the successful development of a web-based application enhances accessibility, allowing both researchers and general users to benefit from state-of-the-art colorization technology. This system proves especially useful in applications such as restoring historical photographs and facilitating creative exploration in content generation. The model's adaptability across diverse image types, coupled with our open-source implementation, establishes a strong foundation for future advancements in image processing. Ultimately, this research not only pushes the boundaries of automated colorization but also paves the way for further deep learning innovations in visual computing, setting a benchmark for next-generation image manipulation techniques.

VI. References

[1] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," ACM SIGGRAPH 2016 Courses, 2016.

[2] R. Dahl, "Automatic colorization," 2016. [Online]. Available: http://tinyclouds.org/colorize/

[3] A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization," ACM Transactions on Graphics (TOG), vol. 23, no. 3, pp. 689–698, 2004.

[4] S. Lee, T. Lee, and K. M. Lee, "Simultaneous colorization and denoising via generative adversarial networks," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1083–1091.

[5] M. Mirza et al., "Image colorization with GANs," arXiv preprint arXiv:1708.07524, 2017.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[7] ResearchGate, "The structure of CNN used in the ECCV16 model," [Online]. Available: https://www.researchgate.net/figure/The-structure-of-CNNused-in-the-ECCV16-model-12 fig3 379259458.

[8] J. Hwang and Y. Zhou, "Image colorization with deep convolutional neural networks," 2016, CS231N.

[9] A. Avery and D. Amin, "Image colorization," 2018, CS230.

[10] K. Nazeri, E. Ng, and M. Ebrahimi, "Image colorization using generative adversarial networks," 2018, arXiv:1803.05400v5.

[11] A. Pandey, R. Sahay, and C. Jayavarthini, "Automatic image colorization using deep learning," 2020, ISSN: 2277-3878.

[12] L. Kiani, M. Saeed, and H. Nezamabadi-pour, "Image colorization using generative adversarial networks and transfer learning," 2020, 978-1-7281-6832-6/20.

[13] A. Kumbhar, S. Gowda, R. Attri, and A. Ketkar, "Colorization of black and white images using deep learning," 2021, e-ISSN: 2395-0056.

[14] Q. Luan et al., "Natural image colorization," Proceedings of the 18th Eurographics conference on Rendering Techniques, Eurographics Association, pp. 309–320, 2007.

[15] K. Michal and B. Smolka, "Competitive image colorization," 2010 IEEE International Conference on Image Processing, IEEE, pp. 405–408, 2010.

[16] R. K. Gupta et al., "Image colorization using similar images," in Proceedings of the 20th ACM international conference on Multimedia, pp. 369–378, 2012.

[17] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," European Conference on Computer Vision, Springer, Cham, pp. 649–666, 2016.

[18] M. Limmer and H. P. Lensch, "Infrared colorization using deep convolutional neural networks," 15th IEEE International Conference on Machine Learning and Applications (ICMLA), IEEE, pp. 61–68, 2016.

[19] 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(4), p. 110, 2016.

[20] J. Schmidhuber, "Deep learning in neural networks: An overview," Neural Networks, vol. 61, pp. 85–117, 2015.