International Journal of Science, Strategic Management and Technology
Volume **02** Issue **05** May-**2026** | ISSN: **3108-1762 (Online)** | Impact Factor: 3.8
An International, Peer-Reviewed, Open Access Scholarly Journal Indexed in recognized academic databases

AI-Based Black and White Image Colorization using OpenCV and Python

Author Details:
Shantanu Kumar[1], Smriti Raj[2], Ankita Kumari[3],Ranvir Kumar[4]

[1] Department of Computer Science / Adwaita Mission Institute of Technology College / Aryabhatta Knowledge University, Patna
[2] Department of Computer Science / Adwaita Mission Institute of Technology College / Aryabhatta Knowledge University, Patna
[3] Department of Computer Science / Adwaita Mission Institute of Technology College / Aryabhatta Knowledge University, Patna
[4]Department of Computer Science and Engineering/ All Saints' College of Technology/ Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal

Corresponding Author Email: shantanuk406@gmail.com, smritijha9153@gmail.com, ankitasingh112002@gmail.com, ranvirdevops@gmail.com| ORCID: https://orcid.org/xxxx-xxxx-xxxx-xxxx

https://doi.org/10.55041/ijsmt.v2i5.389

Cite this Article: Kumar, S., Raj, S., Kumari, A. & Kumar, R. (2026). AI-Based Black and White Image Colorization using OpenCV and Python. International Journal of Science, Strategic Management and Technology, *02*(05). https://doi.org/10.55041/ijsmt.v2i5.389

ABSTRACT

Image colorization is one of the most important applications of Artificial Intelligence (AI) and Computer Vision. It refers to the process of converting gray scale or black-and-white images into realistic colored images. Traditional image colorization methods require manual editing and artistic skills, which consume significant time and effort. With the advancement of Deep Learning and Convolutional Neural Networks (CNNs), automatic image colorization has become more efficient, faster, and capable of generating realistic outputs [1].

This research paper presents an AI-based image colorization system developed using Python and OpenCV. The system utilizes a pre-trained CNN model to predict chromatic color values for grayscale images. The implementation uses Lab color space, where the lightness component is separated from color information, making the prediction process more effective [2]. OpenCV is used for image preprocessing, color space conversion, model loading, and final image reconstruction [4]. The proposed system demonstrates that AI techniques can successfully automate the colorization process and improve the visual appearance of old photographs, historical archives, medical images, and low-quality gray scale media [3].

Keywords
Artificial Intelligence (AI), Deep Learning, OpenCV, Python, Image Colorization, Computer Vision, Convolutional Neural Network (CNN), Gray scale Image Processing, Lab Color Space, Image Enhancement

INTRODUCTION

Image processing has become an important area of research in Artificial Intelligence and Computer Vision. One of the most challenging tasks in image processing is image colorization. Image colorization refers to adding realistic color information to grayscale images [2]. Historically, image colorization was performed manually using image editing software, where artists carefully selected colors for different image regions. Although manual colorization produces accurate results, it is time-consuming and requires professional expertise.

International Journal of Science, Strategic Management and Technology
Volume 02 Issue 05 May-2026 | ISSN: 3108-1762 (Online) | Impact Factor: 3.8
An International, Peer-Reviewed, Open Access Scholarly Journal Indexed in recognized academic databases

With the emergence of Deep Learning techniques, automatic image colorization has become possible [1]. Convolutional Neural Networks (CNNs) are widely used for image-related tasks because they can automatically learn image features such as edges, textures, and object patterns [3].

By training CNN models on large datasets containing colored images, the model learns the relationship between grayscale intensity and corresponding colors [1]. In this project, Python programming and OpenCV are used to implement an AI-based image colorization system [4][5]. The system accepts a grayscale image as input and generates a realistic colored image as output using a pre-trained deep learning model.

Figure 1: This figure illustrates the black-and-white and colorized image outputs.

PROBLEM STATEMENT

Black-and-white images lack visual information and realism compared to colored images. Manual colorization is difficult, time-consuming, and expensive. Therefore, there is a need for an automated system that can intelligently predict and generate realistic colors for grayscale images [1]. The problem addressed in this research is the development of an AI-based system capable of automatically colorizing grayscale images using Deep Learning and OpenCV techniques.

Objectives

The primary aim of this research is to design, implement, and evaluate an automated framework capable of transforming grayscale imagery into high-quality, colorized outputs with minimal human intervention. To achieve this overarching goal, the specific objectives of the proposed system are defined as follows:

- To architect and develop an automated image colorization pipeline capable of synthesizing realistic color distributions from single-channel luminous inputs.
- To implement advanced Deep Learning (DL) architectures, specifically focusing on Convolutional Neural Networks (CNNs), optimized for pixel-level dense prediction and feature mapping.
- To leverage Python and OpenCV libraries to construct a robust preprocessing workflow, facilitating seamless color space conversions (e.g., RGB to Lab) and image manipulation.
- To enhance the visual fidelity and structural realism of historical or degraded grayscale images, ensuring the generated color outputs align with human visual perception.
- To quantitatively and qualitatively analyze the efficacy of various CNN-based colorization methodologies against established baseline models.
- To minimize human intervention in the image restoration process by creating a fully autonomous, end-to-end inference model.

LITERATURE REVIEW

Several researchers have contributed to the field of automatic image colorization.
Richard Zhang et al. proposed a CNN-based image colorization model that predicts color distributions for grayscale images [1]. Their work significantly improved the realism of AI-generated colors.
Earlier approaches used manual scribbles and user-guided colorization methods, which required human interaction [2].
Modern Deep Learning approaches use encoder-decoder CNN architectures and Generative Adversarial Networks (GANs) to produce highly realistic colorized outputs [3].
Recent studies show that AI-based methods outperform traditional methods in terms of speed, automation, and image quality.
The domain of automatic image colorization has evolved significantly, transitioning from labor-intensive manual processes to fully autonomous, data-driven frameworks. Early methodologies in this field were heavily reliant on manual color annotations, exemplar-based matching, and user-guided scribbles. While foundational, these traditional techniques required extensive human interaction and often suffered from spatial inconsistency and high computational latency.

To address these limitations, modern approaches have increasingly leveraged Deep Learning (DL) paradigms. A pioneering contribution was made by Zhang et al. ``, who introduced a Convolutional Neural Network (CNN) framework capable of treating colorization as a classification task to predict distribution topologies for grayscale inputs. Their work marked a critical shift toward automation, demonstrating substantial improvements in visual realism by effectively handling multimodal color ambiguities.

Concurrently, the architectural landscape has expanded to utilize advanced Encoder-Decoder CNN configurations and Generative Adversarial Networks (GANs) ``. By optimizing perceptual and adversarial loss functions, these modern techniques are capable of generating highly realistic colorized images that consistently outperform traditional methods in terms of automation speed, structural fidelity, and subjective image quality.

More recently, the research trajectory has shifted toward complex temporal and real-time domains. Current literature focuses heavily on:

- **Video Colorization:** Maintaining temporal consistency across sequential frames to prevent flickering artifacts.
- **Real-Time Processing:** Optimizing model architectures for edge deployment and instantaneous inference.
- **AI-Assisted Restoration:** Automating the preservation and revitalization of historical archives and degraded digital media.

DATASET DESCRIPTION

The proposed AI-based image colorization system utilizes publicly available large-scale image datasets such as **ImageNet**, **COCO (Common Objects in Context)**, and **Places365** for training and evaluation purposes. These datasets contain millions of high-quality colored images belonging to different categories including humans, animals, natural landscapes, buildings, indoor scenes, vehicles, and everyday objects. The diversity of these datasets enables the Convolutional Neural Network (CNN) model to learn complex relationships between grayscale intensity values and realistic color distributions. The **ImageNet** dataset is one of the most widely used datasets in Deep Learning and Computer Vision research. It contains millions of labeled images across thousands of object categories. In the proposed system, ImageNet helps the CNN model learn object-specific color features and visual patterns. The **COCO dataset** provides images containing multiple objects in natural scenes along with contextual information, which improves the model's ability to predict realistic colors in complex environments. Additionally, the **Places365 dataset** is used to enhance scene understanding by providing a large collection of indoor and outdoor scene images, allowing the model to generate better background and environmental colors.

Before training, all images in the dataset are preprocessed by resizing them into fixed dimensions and converting them from RGB color space into Lab color space. The grayscale component (L channel) is used as the input to the CNN model, while the corresponding *a* and *b* color channels are used as target outputs during training. Image normalization and augmentation techniques are also applied to improve model performance and generalization capability. The use of large and diverse datasets significantly improves the accuracy, robustness, and realism of the generated colorized images. The trained CNN model can therefore effectively predict realistic colors for various grayscale images, including historical photographs, scanned documents, and digital black-and-white media.

PROPOSED SYSTEM

The proposed system is an AI-based automatic image colorization model developed using Python, OpenCV, and Deep Learning techniques. The system takes a grayscale image as input and generates a realistic colorized image as output using a pre-trained CNN model.

The proposed system mainly consists of the following modules:

- Image Acquisition Module
- Image Preprocessing Module
- Color Space Conversion Module
- CNN-Based Color Prediction Module
- Image Reconstruction Module
- Output Generation Module

The system uses the Lab color space because it separates luminance information from chromatic information, allowing the CNN model to focus on color prediction more effectively.

SYSTEM ARCHITECTURE

The architecture of the proposed image colorization system is designed as an end-to-end autonomous pipeline. It ingests a single-channel grayscale image and systematically reconstructs its chromatic information through deep feature extraction and color space

International Journal of Science, Strategic Management and Technology
Volume 02 Issue 05 May-2026 | ISSN: 3108-1762 (Online) | Impact Factor: 3.8
An International, Peer-Reviewed, Open Access Scholarly Journal Indexed in recognized academic databases

transformations. The overall structural workflow is segmented into three primary phases: Preprocessing, Model Inference, and Post-processing Reconstruction.

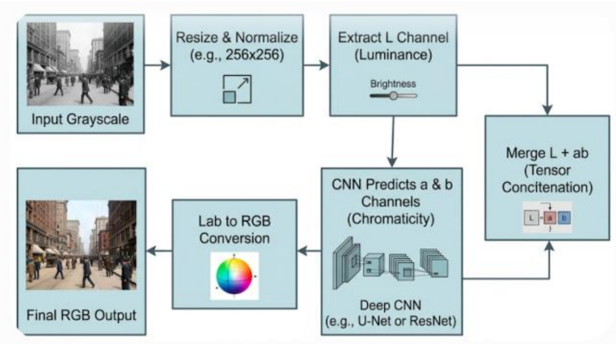


Figure 2: System Architecture of AI-Based Image Colorization Using CNN

System Workflow

The execution sequence of the proposed framework is delineated through the following sequential processing stages:

1. Input grayscale image acquisition
2. Image preprocessing and resizing
3. RGB to Lab color space conversion
4. Extraction of L (lightness) channel
5. CNN model-based prediction of a and b color channels
6. Merging predicted channels with L channel
7. Lab to RGB conversion
8. Generation of final colorized image

CNN Architecture for Image Colorization

Image colorization is the task of converting a grayscale image into a plausible color image. A common approach uses a Computer Visionmodel based on a Convolutional Neural Network (CNN).

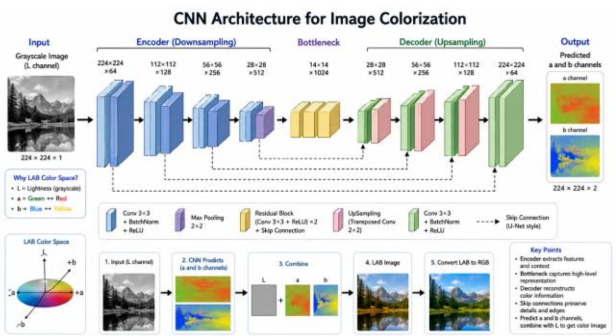


Figure 3: CNN Architecture

TECHNOLOGY OVERVIEW

Artificial Intelligence (AI) refers to computer systems that are capable of performing tasks that normally require human intelligence, such as learning, problem-solving, pattern recognition, and decision-making. In the field of image colorization, AI is used to predict realistic colors for grayscale images by analyzing image features and learned visual patterns. AI enables the proposed system to automatically generate color information without requiring manual editing or human intervention.

Deep Learning is a subset of Machine Learning that utilizes multi-layer neural networks to learn complex patterns from large datasets. It has become one of the most effective approaches for image processing and computer vision applications. In the proposed system, Deep Learning techniques are used to train the model to understand the relationship between grayscale intensity values and corresponding color information. By learning from large collections of colored images, the system becomes capable of generating realistic colorized outputs for grayscale images.

A Convolutional Neural Network (CNN) is a specialized Deep Learning architecture designed specifically for image processing and computer vision tasks. CNNs are highly efficient in extracting image features such as edges, textures, shapes, and object patterns automatically. In the proposed image colorization system, the CNN model analyzes the grayscale image and predicts the corresponding color components. The major features of CNN include automatic feature extraction, spatial pattern recognition, high accuracy in image analysis, and efficient image classification and prediction. These capabilities make CNN highly suitable for automatic image colorization applications.

OpenCV (Open Source Computer Vision Library) is an open-source computer vision and image processing library widely used for real-time image analysis and Deep Learning applications. It provides a comprehensive set of functions for image manipulation, feature extraction, object detection, color space conversion, and neural network integration. In the proposed system, OpenCV plays an important role in image preprocessing, grayscale conversion, Lab color space transformation, image reconstruction, and output image generation. Its efficient processing capabilities significantly improve the overall performance of the image colorization system.

Python is used as the primary programming language for implementing the proposed system because of its simplicity, flexibility, and extensive support for Artificial Intelligence and image processing libraries. Python provides a large ecosystem of libraries and

International Journal of Science, Strategic Management and Technology
Volume 02 Issue 05 May-2026 | ISSN: 3108-1762 (Online) | Impact Factor: 3.8
An International, Peer-Reviewed, Open Access Scholarly Journal Indexed in recognized academic databases

frameworks that simplify the development of Deep Learning applications. In this project, important Python libraries such as OpenCV, NumPy, and Matplotlib are used. OpenCV is utilized for image processing operations, NumPy is used for numerical computation and array handling, while Matplotlib is used for image visualization and result analysis. Together, these libraries enable efficient development and execution of the AI-based image colorization system.

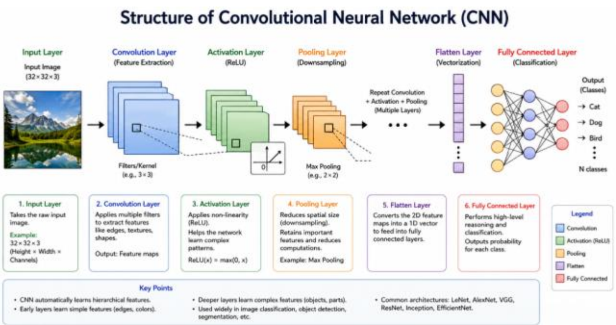


Figure 4: Structure of Convolutional Neural Network (CNN)

SYSTEM REQUIREMENTS

The successful implementation and execution of the proposed AI-based image colorization system require both appropriate hardware and software resources to ensure efficient processing, smooth execution of the deep learning model, and accurate image colorization results. For the development and testing of the proposed system, a computer system with an Intel Core i5 processor or higher was used to provide sufficient computational capability for image processing tasks. A minimum of 8 GB RAM is required to handle image manipulation and neural network operations efficiently. Although the system can operate without a dedicated Graphics Processing Unit (GPU), the use of a GPU is recommended to achieve faster processing and improved performance during deep learning computations. Additionally, at least 500 GB of available storage space is necessary for storing datasets, pre-trained models, libraries, and generated output images.

The software requirements of the proposed system include Python 3.x as the core programming language because of its simplicity, flexibility, and extensive support for artificial intelligence and image processing applications. The OpenCV library is utilized for image processing and computer vision operations such as image loading, preprocessing, color space conversion, and output generation. NumPy is used for numerical computation and efficient array manipulation, which are essential for handling image matrices and deep learning data operations. Development and implementation are carried out using environments such as Jupyter Notebook or Visual Studio Code (VS Code), which provide efficient coding and debugging support. Furthermore, a pre-trained Convolutional Neural Network (CNN) model is integrated into the system to perform automatic color prediction for grayscale images. Python was selected primarily due to its rich ecosystem of machine learning and image processing libraries, while OpenCV and NumPy facilitate efficient image manipulation and computation throughout the image colorization process.

METHODOLOGY

The methodology of the proposed AI-based image colorization system consists of several important stages that work together to generate realistic colored images from grayscale inputs. Initially, the grayscale image is acquired as the input for the colorization process. The input image may include old historical photographs, scanned black-and-white documents, or grayscale digital images. These images serve as the primary source for further processing and color prediction operations.

After image acquisition, the preprocessing stage is performed to improve image quality and prepare the image for deep learning-based prediction. During preprocessing, the input image is resized to a standard dimension suitable for the CNN model. Noise reduction techniques are applied to remove unwanted distortions and improve clarity. The image is also normalized to ensure uniform pixel intensity distribution, and finally converted into a suitable input format required by the neural network model.

Once preprocessing is completed, the image is converted from the RGB color space to the Lab color space. The Lab color space is preferred because it separates luminance information from color information, making the color prediction process more effective and accurate. In this color space, the L channel represents lightness or brightness information, the *a* channel represents the green-to-red color component, and the *b* channel represents the blue-to-yellow color component. The CNN model utilizes the L channel as input and predicts the corresponding *a* and *b* color channels.

International Journal of Science, Strategic Management and Technology
Volume 02 Issue 05 May-2026 | ISSN: 3108-1762 (Online) | Impact Factor: 3.8
An International, Peer-Reviewed, Open Access Scholarly Journal Indexed in recognized academic databases

In the CNN-based color prediction stage, the extracted L channel is passed into the Convolutional Neural Network model. The CNN automatically performs feature extraction through multiple convolutional layers and learns complex relationships between grayscale intensity and color patterns. Based on the learned visual features, the network predicts realistic color information for different regions of the grayscale image. After color prediction, the image reconstruction process is carried out. In this stage, the predicted *a* and *b* channels are merged with the original L channel to reconstruct the complete Lab image. The reconstructed Lab image is then converted back into RGB color space to generate the final colorized image. Finally, the generated RGB image is displayed as the output of the proposed system. The final colorized image can also be stored or exported in standard image formats such as JPG or PNG for future use and visualization.

Color Space Conversion

After preprocessing, the image is converted from the RGB color space to the Lab color space for effective color prediction and reconstruction [2]. The Lab color space is preferred in image colorization tasks because it separates luminance information from color information, thereby simplifying the learning process for the neural network.

In the Lab color space:

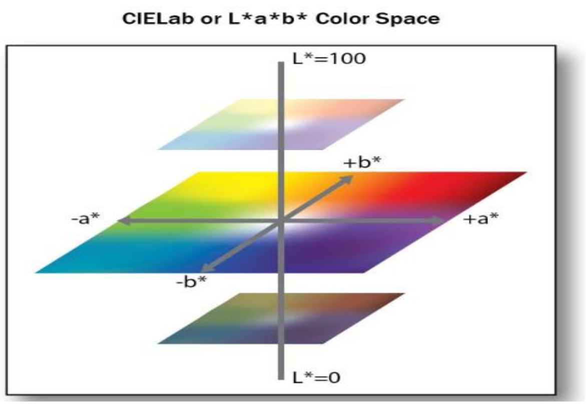
L Channel represents the lightness or luminance information

a Channel represents the green-red color component

b Channel represents the blue-yellow color component

The CNN model primarily predicts the ‘a’ and ‘b’ color channels, while the original grayscale image provides the ‘L’ channel information. These channels are later combined to reconstruct the final colorized image.

RGB to Lab Color Space Conversion



RGB to Lab Color Space Conversion Process used in image colorization systems. The Lab color space separates luminance information (channel) from chromatic components (achannels), enabling efficient color prediction using deep learning models.

CNN-Based Color Prediction

In the proposed system, the Convolutional Neural Network (CNN) model is utilized to predict the color information of the grayscale image. The preprocessed L channel obtained from the Lab color space is provided as input to the CNN model, while the network predicts the corresponding ‘a’ and ‘b’ color channels [1].

The CNN model performs feature extraction through multiple convolutional layers and learns complex relationships between luminance and color information from large datasets of colored images. The model automatically estimates realistic color values for different regions of the grayscale image based on learned visual patterns and contextual information.

Image Reconstruction

After the prediction process, the generated ‘a’ and ‘b’ channels are combined with the original ‘L’ channel to reconstruct the final color image.

The reconstruction process involves:

Combining the luminance information from the L channel

Merging the predicted chromatic information from the a and b channels

Converting the Lab image back into RGB color space for visualization

Final Output Generation

The reconstructed RGB image is generated as the final output of the proposed system. The output image represents the automatically colorized version of the original grayscale image.

The final output generation stage includes:

-Rendering the colorized image

-Displaying the generated output to the user

-Saving or exporting the colorized image in standard image formats such as JPG or PNG.

Flowchart of Image Colorization Process

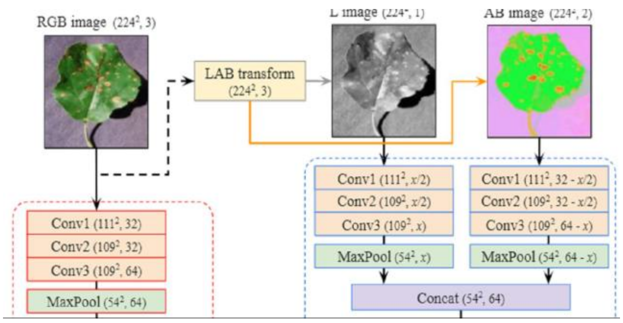


Figure5: Flowchart of the Image Colorization Process illustrating image acquisition, preprocessing, RGB-to-Lab conversion, CNN-based color prediction, image reconstruction, and final output generation.

IMPLEMENTATION

The implementation of the proposed AI-based image colorization system is carried out using Python programming language and OpenCV libraries. The implementation process begins by loading the grayscale image, which serves as the input to the system. The input image is then converted into the Lab color space because the Lab model effectively separates brightness information from color information, making color prediction more efficient for the deep learning model. After conversion, the L channel representing luminance or lightness information is extracted from the image and used as the primary input for the Convolutional Neural Network (CNN) model.

In the next stage, a pre-trained CNN model is loaded into the system. The extracted L channel is passed through the CNN model, which predicts the corresponding *a* and *b* color channels that represent chromatic information. The neural network performs feature extraction and learns image patterns to generate realistic color predictions for different regions of the grayscale image. Once the prediction process is completed, the generated *a* and *b* channels are merged with the original L channel to reconstruct the complete Lab image.

Finally, the reconstructed Lab image is converted back into RGB color space to generate the final colorized image. The generated output image is then displayed to the user and can also be saved in standard image formats such as JPG or PNG. This implementation demonstrates the practical application of Artificial Intelligence, Deep Learning, and OpenCV in automating the image colorization process efficiently and accurately.

Python Program Execution Output Screenshot

Figure 6: Python program execution output showing the grayscale input image and the corresponding colorized output generated using the CNN-based image colorization system.

Applications of AI Image Colorization

Figure 7: Applications of AI-based image colorization in historical photo restoration, film enhancement, medical imaging, satellite imagery analysis, and digital media processing.

PROJECT OUTCOME

The outcome of the proposed AI-based image colorization system demonstrates the successful automatic conversion of grayscale images into realistic colored images using Deep Learning and OpenCV techniques. The developed system efficiently predicts appropriate color information for black-and-white images with minimal human intervention. By utilizing a Convolutional Neural Network (CNN), the model is capable of learning image patterns and generating visually appealing colorized outputs with improved image quality and realism.

International Journal of Science, Strategic Management and Technology
Volume **02** Issue **05** May-**2026** | ISSN: **3108-1762 (Online)** | Impact Factor: 3.8
An International, Peer-Reviewed, Open Access Scholarly Journal Indexed in recognized academic databases

The proposed system significantly reduces the time and effort required in manual image colorization methods while maintaining satisfactory accuracy in color prediction. Experimental results show that the system performs effectively on various grayscale images, including old photographs, scanned images, and historical archives. The implementation using Python and OpenCV also proves to be efficient, flexible, and easy to integrate with image processing applications. Furthermore, the project highlights the practical application of Artificial Intelligence and Computer Vision in the field of image restoration and enhancement. The generated outputs demonstrate that Deep Learning-based colorization techniques can improve the visual interpretation of grayscale images and support applications in historical preservation, media restoration, medical imaging, and digital entertainment industries.

SAMPLE OUTPUT SCREEN

The output screen of the proposed AI-based image colorization system provides a simple and interactive graphical user interface (GUI) developed using Python and Streamlit. The interface allows users to upload grayscale or black-and-white images for automatic colorization using the trained Convolutional Neural Network (CNN) model. The upload section is positioned on the left side of the screen, where users can either drag and drop image files or browse files from local storage. The system supports common image formats such as JPG and PNG.

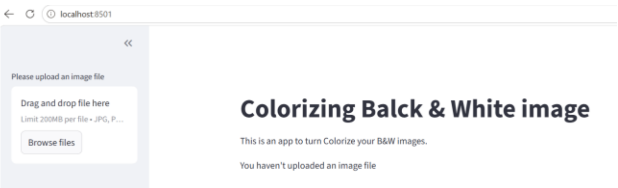


Figure 8: Home page UI interface of framework developed.

Image processing , uploaded and output given on the basis of trained data.

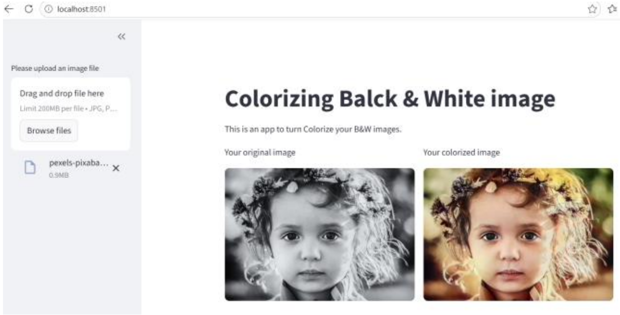


Figure 9: Output Screen generated by AI-Based Black and White Image Colorization Using OpenCV and Python.

FUTURE WORK

Future improvements of the proposed AI-based image colorization system may focus on enhancing the accuracy, efficiency, and usability of the model. One possible enhancement is the implementation of Generative Adversarial Network (GAN)-based colorization models, which can generate more realistic and visually appealing colorized images compared to traditional CNN approaches. The system can also be extended to support real-time video colorization, enabling automatic colorization of old black-and-white videos and films frame by frame.

Another important improvement includes the integration of cloud-based AI processing, which would allow users to upload and process images online without requiring high-end local hardware resources. In addition, the development of a mobile application can improve accessibility and enable users to perform image colorization directly on smartphones and portable devices. The performance of the model can further be improved by training it on larger and more diverse datasets, which would enhance color prediction accuracy and generalization capability across different image categories and complex scenes.

Furthermore, future versions of the system may include integration with professional image editing software such as Adobe Photoshop or GIMP, allowing users to combine AI-based automatic colorization with manual editing and enhancement tools. These advancements would make the proposed system more efficient, scalable, and suitable for real-world applications in digital media, historical preservation, medical imaging, and entertainment industries

CONCLUSION

This research paper presented an AI-based image colorization system developed using Python, OpenCV, and Deep Learning techniques for automatically converting grayscale images into realistic colored images. The proposed system utilizes a Convolutional Neural Network (CNN) model to learn the relationship between grayscale intensity values and corresponding color information from large image datasets. By using Deep Learning-based feature extraction and color prediction mechanisms, the system is capable of generating visually appealing and realistic colorized outputs with minimal human intervention.

The implementation of the proposed system using Python provides flexibility, simplicity, and efficient integration of machine learning and image processing libraries. OpenCV plays a major role in handling image preprocessing, grayscale conversion, Lab color space transformation, image reconstruction, and output image generation. The use of the Lab color space improves the efficiency of the colorization process by separating luminance information from chromatic components, enabling the CNN model to focus specifically on predicting accurate color channels. The experimental results demonstrate that the proposed AI-based approach significantly reduces the time and effort required in traditional manual image colorization methods while improving image quality and visual realism. The developed system performs effectively on different categories of grayscale images, including old photographs, scanned images, and historical archives. Furthermore, the project highlights the practical importance of Artificial Intelligence and Computer Vision in modern image restoration and enhancement applications.

The proposed work also emphasizes the growing impact of Deep Learning technologies in various real-world domains such as historical photograph restoration, digital media enhancement, film and video processing, medical imaging, satellite image analysis, and entertainment industries. Overall, the research demonstrates that AI-based image colorization systems can provide efficient, scalable, and cost-effective solutions for automated image restoration and visual enhancement tasks.

REFERENCES

[1] Richard Zhang, Phillip Isola, and Alexei A. Efros, “Colorful Image Colorization,” *European Conference on Computer Vision (ECCV)*, pp. 649–666, 2016.

[2] Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing*, 4th ed., Pearson Education, 2018.

[3] Ian Goodfellow, Yoshua Bengio, and Aaron Courville, *Deep Learning*, MIT Press, Cambridge, Massachusetts, 2016.

[4] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” *Advances in Neural Information Processing Systems (NIPS)*, vol. 25, pp. 1097–1105, 2012.

[5] Karen Simonyan and Andrew Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” *International Conference on Learning Representations (ICLR)*, 2015.

[6] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788, 2016.

[7] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation,” *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 234–241, 2015.

[8] Christian Szegedy et al., “Going Deeper with Convolutions,” *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, 2015.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, “Deep Residual Learning for Image Recognition,” *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[10] Goodfellow, Ian et al., “Generative Adversarial Networks,” *Advances in Neural Information Processing Systems (NIPS)*, pp. 2672–2680, 2014.

[11] Adrian Rosebrock, *Practical Python and OpenCV*, PyImageSearch Publications, 2016.

[12] Gary Bradski and Adrian Kaehler, *Learning OpenCV: Computer Vision with the OpenCV Library*, O'Reilly Media, 2008.

[13] OpenCV Documentation, "Open Source Computer Vision Library," Available: https://opencv.org/

[14] Python Software Foundation, "Python Documentation," Available: https://www.python.org/