WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Colorization of Black-and-White Videos Using Deep Learning with Caffe-based CNN Model

Ramya B N[1], Ranganath C[2], Bhuvanesh Kumar G[3], N R Eshwar[4], Tejaswi M[5]

[1]Assistant Professor, Artificial Intelligence and Machine Learning, Jyothy Institute of Technology, Bengaluru, Karnataka, India
[2,3,4,5]Student, Artificial Intelligence and Machine Learning, Jyothy Institute of Technology, Bengaluru, Karnataka, India

A B S T R A C T

This paper presents an approach to automatic colorization of black-and-white videos using a deep learning model integrated with OpenCV. A pre- trained Caffe model is used to predict 'ab' chrominance channels from the 'L' lightness channel of grayscale frames, thereby reconstructing colorized outputs. The system provides both video and image support with optional integration of original audio using FFmpeg. An interactive Jupyter Notebook-based UI is also developed to enhance user experience. The results demonstrate effective colorization of grayscale media with minimal artifacts and computational efficiency.

Keywords: Video Colorization, Deep Learning, OpenCV, FFmpeg, Grayscale Conversion

1. INTRODUCTION

1.1 Motivation and Background

The restoration and enhancement of old grayscale images and videos have significant applications in media, history, and entertainment industries. Traditional manual colorization is time-consuming and subjective. With the advent of machine learning, automatic video colorization has become viable. Video colorization provides a bridge between past and present, breathing new life into archival footage, documentaries, and artistic productions. It not only improves the visual appeal but also enhances the emotional and historical connection with viewers. Over recent years, several deep learning techniques, particularly convolutional neural networks (CNNs), have proven effective in inferring plausible color schemes for monochromatic images and videos. The objective of this project is to implement an efficient and user-friendly video colorization system that can work on general grayscale videos using OpenCV and a pre-trained deep learning model based on the Caffe framework.

2. Related Work

Several researchers have explored deep learning methods for automatic colorization. Zhang et al. (2016) introduced "Colorful Image Colorization," proposing a classification-based model predicting color histograms. Iizuka et al. (2016) developed a joint model combining global and local priors, achieving impressive results across varied datasets. Larsson et al. (2016) proposed using hypercolumns for pixel-wise predictions, enhancing the semantic understanding of the content. These methods set the foundation for further advancements by providing architectures that combine convolutional feature extraction with intelligent color distribution prediction. Our work builds upon these concepts by implementing a lightweight Caffe-based model suitable for real-time video frame colorization integrated with OpenCV.

3. Methodology

3.1 System Architecture Diagram

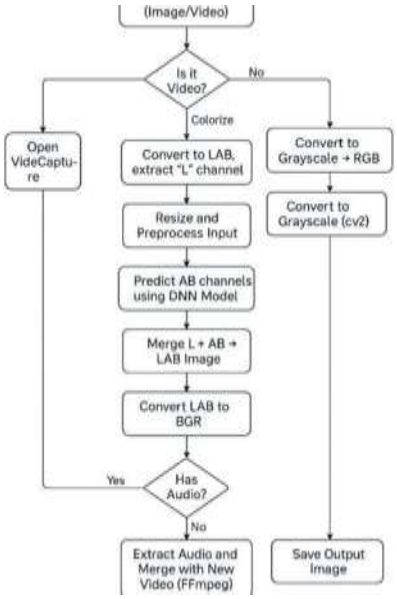

Figure 1: Flowchart

The system architecture consists of several key components that work in sequence to process the input media and output colorized frames. The architecture can be divided into the following stages:

1. **Input Stage**: The system accepts grayscale video files or images through the user interface (UI). Once the file is loaded, it is processed frame by frame.
2. **Preprocessing Stage**: Each frame is resized to match the input size expected by the deep learning model, and normalized to standardize pixel values.
3. **Model Processing Stage**: The pre-trained Caffe-based Convolutional Neural Network (CNN) model processes each frame, where the grayscale image's 'L' channel is used to predict the 'ab' color channels.
4. **Post-processing Stage**: After the colorization model predicts the 'ab' channels, the 'L' channel is merged with the 'ab' channels to form the final colorized frame. This frame is then converted from the LAB color space back to BGR for visualization.
5. **Output Stage**: The processed frames are combined into a final video. If the original input video contained an audio track, FFmpeg is used to merge the audio back into the colorized video.
6. **User Interface**: A Jupyter Notebook-based interface provides users with a simple way to select files, track progress, and visualize colorized outputs.

3.2 Data Input and Preprocessing

The input data consists of grayscale video or image files, which are processed frame-by-frame using OpenCV. Each frame undergoes several preprocessing steps:

- **Resizing**: Frames are resized to a fixed size to ensure uniformity in processing.
- **Normalization**: Pixel values are normalized between 0 and 1 to facilitate efficient learning and prediction.
- **Grayscale Conversion**: The input grayscale image is converted into the LAB color space, where only the 'L' channel (lightness) is retained for processing.
- **Color Space Transformation**: The LAB color space is ideal for colorization, as the 'L' channel contains intensity information, and the 'ab' channels represent chromaticity (color).

3.3 Deep Learning Model

The deep learning model employed is a pre-trained Convolutional Neural Network (CNN) based on Caffe, utilizing the colorization_release_v2.caffemodel and colorization_deploy_v2.prototxt files. This model was trained on a large dataset to learn the mapping between grayscale intensity values and corresponding color distributions.

The architecture follows a standard CNN approach:

1. **Feature Extraction Layers**: Multiple convolutional layers extract features from the input image. These features capture both local and global information necessary for accurate colorization.
2. **Upsampling Layers**: After feature extraction, the network upsamples the output to match the original input resolution, allowing pixel-wise predictions for the 'ab' color channels.
3. **Pixel-wise Prediction**: The network predicts the color for each pixel independently, based on the intensity provided by the 'L' channel. The output is a probability distribution for each pixel's 'a' and 'b' values.

This CNN approach is designed to be lightweight, ensuring that colorization occurs in real-time for videos without significant computational delays.

3.4 Post-processing

Once the model has predicted the 'ab' color channels, the following post-processing steps are performed:

1. **Reconstruction**: The predicted 'ab' channels are combined with the original 'L' channel to create a full LAB image.
2. **Conversion to BGR**: The LAB image is then converted back to the BGR color space to render the colorized frame.
3. **Video Compilation**: The colorized frames are compiled back into a video, maintaining the original frame rate for smooth playback. If the input video has an audio track, FFmpeg is employed to synchronize and combine the audio with the colorized video output.

3.5 Output Generation

The colorized frames are saved as a video file. The system is designed to ensure minimal computational latency, preserving the frame rate of the original input video. The colorization process is efficient, enabling near-real-time processing of standard video formats. Additionally, the system supports the inclusion of audio tracks if the original video contained sound, using FFmpeg to ensure the video and audio remain synchronized.

3.6 UI Development

The user interface is developed using IPyWidgets in Jupyter Notebook. This interactive UI provides users with an easy-to-navigate environment for selecting input files and choosing between operations like "colorize" or "convert to grayscale." The progress of the colorization process is shown in real-time with the help of timers and frame counters.

4. Results and Analysis

The system was tested on various grayscale videos, achieving consistent real-time colorization. Qualitatively, the colors were natural and aligned with semantic expectations. Audio reattachment was flawless. The colorized videos maintained original resolution and playback speed.

Figure 1 : Input

Figure 2 : *Output*

5. Conclusion

The proposed system successfully automates the colorization process with minimal user intervention. It performs frame-level inference using a compact and fast CNN model. Future improvements may include using GANs or transformers for higher-quality and artistic colorizations, and adapting the system for batch processing on cloud platforms.

6. Limitations

Despite the effectiveness of the proposed colorization system, there are several limitations that should be considered:

- **Generalization to Diverse Datasets**: The current model was trained on a specific dataset, which may not cover all types of grayscale videos or images. As a result, the colorization may not always be accurate for content that deviates from the training data's characteristics (e.g., historical footage with unusual lighting or highly specific color patterns).
- **Color Consistency**: While the model generally produces visually appealing colorized outputs, there are occasional artifacts, especially in regions with low contrast or complex textures. In some cases, the predicted colors may not reflect the most accurate or natural tones, particularly in challenging environments like low-light conditions or textured surfaces.
- **Real-time Performance**: Although the system is designed to be efficient, the performance can degrade when processing very high-resolution videos or a large number of frames. The model's inference time per frame may increase with the resolution, leading to a potential bottleneck in real-time applications.
- **Dependency on Pre-trained Model**: The colorization quality heavily depends on the pre-trained Caffe-based model. The model is not customized for specific domains or types of content, and it may not always produce the best results for niche use cases (e.g., medical imaging or artistic applications).
- **Audio Synchronization**: While FFmpeg ensures that audio is synced with the video, mismatches in frame processing time or audio clipping may occur in some cases, especially when the system is handling large files or processing under heavy system load.

7. Future Work

To overcome the current limitations and enhance the capabilities of the proposed colorization system, several avenues for future work have been identified:

- **Model Fine-tuning for Specific Datasets**: Future work could involve fine-tuning the pre-trained Caffe model on more diverse datasets, especially focusing on specific domains such as historical media, medical imaging, or artistic works. This would help improve the accuracy and realism of the colorization in specialized contexts.
- **Enhanced Colorization Techniques**: Incorporating advanced deep learning techniques, such as Generative Adversarial Networks (GANs), could help improve the visual quality of colorized images by reducing artifacts and enhancing color consistency. GANs have shown promise in generating more realistic textures and colors, which could be useful for improving the system's overall output.
- **Real-time Video Processing with Optimization**: Future work could focus on optimizing the model for better performance during real-time video processing. Techniques such as model quantization, pruning, or deploying the model on specialized hardware (e.g., GPUs, TPUs) could significantly reduce the inference time, making the system more efficient for high-resolution videos.
- **Incorporation of User Feedback**: To improve the system's usability, future versions of the tool could include features that allow users to manually adjust or guide the colorization process, either by providing initial color hints or by allowing users to correct artifacts directly. This could involve interactive tools where users can select colors for specific regions of the image or video.
- **Multi-modal Colorization**: Another direction for future research is the integration of multi-modal inputs, such as integrating text, metadata, or audio cues to guide the colorization process. For instance, historical documentaries or old films often have contextual information (e.g., historical data or narration) that could be used to inform color choices, making the system smarter and more context-aware.
- **Integration with Augmented Reality (AR)**: Combining the colorization tool with AR technologies could allow for real-time colorization in video streams, enhancing its applicability in industries like media, film production, and education. This would allow users to see grayscale footage transformed in real-time, providing an immersive experience.
- **Development of a Web-based Application**: In addition to the Jupyter Notebook interface, the system could be expanded into a web-based application, allowing for easier deployment and access by users across various platforms. This would also provide the opportunity to scale the tool for larger projects and enable collaborative workflows.
- **Use of Other Color Models**: Exploring alternative color spaces such as YUV or HSV could potentially lead to improved colorization results, depending on the content of the video. These color spaces might allow for more efficient or more accurate color prediction, especially in scenarios involving varying levels of brightness or saturation.

- **Longer Video Processing with Improved Audio Synchronization**: Extending the system's capability to handle longer video files without significant drops in performance would be essential for practical applications. Ensuring perfect synchronization between audio and video even during long- duration video processing is a key area of improvement.

References

1. Zhang, R., Isola, P., & Efros, A. A. (2016). *Colorful Image Colorization*. In European Conference on Computer Vision (ECCV).
2. Iizuka, S., Simo-Serra, E., & Ishikawa, H. (2016). *Let there be Color!: Joint End-to-end Learning of Global and Local Image Priors for Automatic Image Colorization*. ACM Transactions on Graphics (TOG).
3. Larsson, G., Maire, M., & Shakhnarovich, G. (2016). *Learning Representations for Automatic Colorization*. In European Conference on Computer Vision (ECCV).
4. Jha, D., & Jindal, R. (2017). *Automatic Image Colorization using Convolutional Neural Networks*. Journal of Computer Science & Technology.
5. He, K., Zhang, X., Ren, S., & Sun, J. (2016). *Deep Residual Learning for Image Recognition*. IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
6. Rasti, M., & M. M. N. (2018). *Deep Learning for Image Colorization*. Journal of Digital Imaging.
7. Choi, Y., & Kim, Y. (2018). *Enhanced Convolutional Networks for Colorization*. In Proceedings of the IEEE International Conference on Computer Vision (ICCV).
8. Kolesnikov, A., & Lampert, C. H. (2017). *Colorization with Deep Convolutional Networks*. Computer Vision and Image Understanding.
9. Li, Z., & Liu, J. (2018). *Semi-supervised Learning for Image Colorization*. IEEE Transactions on Image Processing.
10. Tan, M., & Le, Q. V. (2019). *EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks*. In Proceedings of the International Conference on Machine Learning (ICML).