A Comparative Analysis of Two Generative Models for Cinematic Color Grading

Angela Ye

Abstract

Color correcting and color grading are essential for visual storytelling, conveying specific moods, and enhancing the overall impact of a film. Currently, color grading is a manual process: colorists use their intuition and mastery of complex editing software. In this work, a novel approach is introduced: the use of generative AI models to color correct and color grade images, given a specific label. Two primary frameworks were custom-built and tested: a Conditional Diffusion Model using U-Net architectures with forward and reverse diffusion processes, and a Conditional Wasserstein Generative Adversarial Network (C-WGAN) utilizing a critic and a generator with a U-Net architecture. Both models were trained on a subset of the Flickr 8 dataset, augmented to reflect a range of stylistic transformations such as brightness adjustment, vivid colors, grainy film effects, and more. The C-WGAN produced better outputs compared to the Conditional Diffusion Model, with edits appearing to be more noticeable. The Conditional WGAN demonstrated higher effectiveness in replicating human color grading tasks efficiently, with a Structural Similarity Index Measure (SSIM) of 0.7729 and a Peak Signal-to-Noise Ratio (PSNR) of 25.922. Regarding computational efficiency, the average inference time (editing time) was 0.44 seconds per image. These findings suggest that generative AI—especially C-WGANs—holds promise for automating aesthetic tasks such as color grading, offering a balance between originality, visual appeal, and efficiency.

1 Introduction

The entertainment industry is a staple in global businesses and pop culture, being a global commodity that generates $2.8 trillion globally per year [9]. Recently, driven by rapid advancements in technology, the industry has gone through a remarkable series of changes. Artificial intelligence is transforming production processes and reducing expenses, aiding filmmakers on editing and special effects [15]. This study explores the role of artificial intelligence in a specific post-production process: color correction and color grading.

When viewing visual media, the footage appearing on the screen has been edited to look more vivid and colorful compared to the raw footage that comes from the camera. Color correction and color grading are critical post-processing steps in photography and film production that enhance the visual tone, mood, and aesthetic appeal of an image. Color correction is about fixing problems in exposure, contrast, white balance, and overall color to produce a natural baseline image that matches the eye's view. On the other hand, the creative process of color grading allows the editor to make their own artistic choices, using specific color schemes to tell a specific story [16]. Traditionally, this process has been performed manually using professional software such as Adobe Lightroom Classic, DaVinci Resolve, Adobe Premiere Pro, etc. In these workspaces, color wheels, video scopes, and color curves, along with terms such as luminance, hue, white balance, and gamma, can be intimidating to inexperienced colorists and editors (in other words, regular people). Although

mobile phones such as Apple and Android offer automated color correction features, the underlying methods remain hidden from the public.

To address this problem, this study evaluates a conditional diffusion model and a conditional Wasserstein GAN (C-WGAN) on the task of color grading. Although prior work has explored colorization and aesthetic image translation, few studies have conducted a direct comparison between GANs and diffusion models in such a specific task. The main contributions are as follows: (1) Identifying an application of generative models in video editing (2) Introducing two models, a C-WGAN and Conditional Diffusion Model, tasked for color correction and color grading (3) Comparing and analyzing the output of each model, formulating an answer to: "Which generative model is better suited for the task of color grading?"

2 Related Work

The growth of generative AI models raises the question of whether machines can assist or even replicate the work of human colorists. In previous studies, a conditional generative adversarial network (cGAN)-based Pix2Pix framework was developed for image-to-image translation tasks like semantic mapping, image colorization, and edge-to-photo. This framework mapped input images to output images while also learning a loss function for training the mapping. The Pix2Pix model, which included a U-Net generator and a PatchGAN discriminator, showed that cGANs can produce more realistic textures and vibrant colors than models using only L1 or L2 loss [4].

In addition, Treneska et al. (2022) proposed a new method that uses a cGAN for self-supervised feature learning through image colorization. Their research demonstrated that cGANs can aid in transfer learning for two other tasks: multi-label image classification and semantic segmentation, improving performance compared to models trained from scratch. They evaluated their model using PSNR and SSIM metrics and achieved a mean SSIM of 0.85, underscoring its structural accuracy compared to the ground truth [14].

Most recently, Vavilala et al. (2025) introduced a diffusion-based dequantization model that performs both color palette transfers and color-conditioned inpainting. This approach allows detailed color control while maintaining texture, marking a significant improvement over earlier GAN-based methods. They showed that by conditioning a frozen diffusion model on quantized color inputs, luminance or gradient texture maps, and an auxiliary encoder, the model can create visually appealing color transformations while allowing for high-level palette editing [18].

3 Methodology

3.1 Data

In this work, I used the Flickr8k Dataset available through the Kaggle Repository, comprising 8,000 images containing a variety of scenes and colors; the images were chosen from six different image groups to ensure diversity and variability [6]. To prepare the data for training and analysis, each image first went through preprocessing, being resized to (224, 224, 3) and changed to a np.uint8 image type. The images then went through a series of systematic edits using the color augmentation functions in the imgaug Python library [3]. This library converts an image from an RGB color space into an HVS or Lab color space to separate hue, saturation, and brightness, directly replicating the actual color grading process.

A total of 5 labels were used: Brightened, Darkened, Grainy Film, Muted Colors, and Vivid Colors. These 5 labels create 9 different edits for the images, listed below:

"Brightened", "Darkened": color correction edits to ensure consistency in exposure.

"Grainy Film," "Muted Colors," and "Vivid Colors": color grading edits that add a specific mood to the image.

"Vivid Colors + Grainy Film," "Muted Colors + Grainy Film," "Vivid Colors + Brightened," "Muted Colors + Darkened": combinations of color correction and color grading edits.

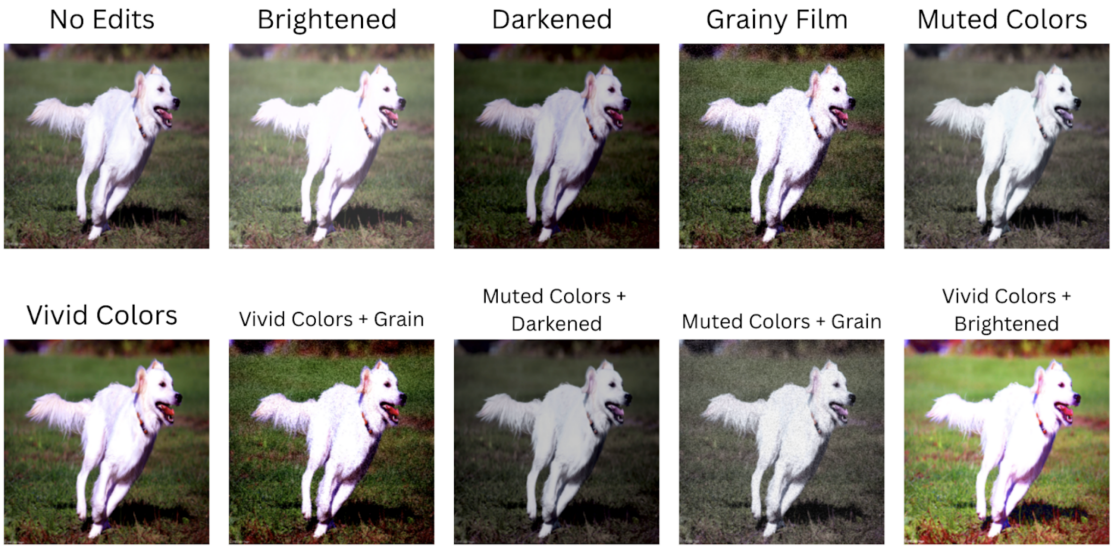

Figure 1: Figure 1: All color corrected and color graded edits

The resulting dataset is different from the usual image datasets, each item in the dataset is structured as a tuple containing the original image, the edited image, and the corresponding label for the edit that was made. All labels were multi-hot encoded, meaning that a label [0, 1, 0, 0, 0, 0] means "Brightened", while [0, 0, 0, 1, 0, 1] corresponds to the combination "Grainy Film + Vivid Colors."

3.2 Conditional Wasserstein GAN (C-WGAN)

Figure 3 shows the architecture of the C-WGAN. The C-WGAN consists of two components: a generator and a critic. In order to constrain the model's edits solely to color modifications, the conventional generator architecture was enhanced.

In a standard GAN, the generator tries to convert random noise into observations that look as if they have been sampled from the original dataset, and the discriminator tries to predict whether an observation comes from the original dataset or is one of the generator's forgeries [2]. Specifically for the generator, it takes a random vector from a latent space and upsamples it into an image by applying convolutional transpose operations. However, in this scenario, the generator consists of a UNET, a U-shaped encoder-decoder network architecture that includes encoding blocks, decoding blocks, and a bridge called a "bottleneck" in between the two [13]. Instead of generating a completely new image, the UNET contains skip connections that maintain the structural integrity of the image.

The goal of this research is to produce images with different tones of color, not completely new images that have not been seen before. To accomplish this, a label map is created. The encoder is responsible for downsampling the image, capturing and learning features such as color distribution and structure. It is implemented as a Convolutional Neural Network (CNN), a type

of deep learning algorithm primarily used for analyzing visual data, such as images [2]. On the other hand, the decoder is responsible for upsampling (similar to the generator of a standard GAN), modifying the colors and tone of the image based on learned transformations. It is built with transposed convolutional layers and concatenations for skip connections to pass through. A diagram of the UNET's structure is shown below.

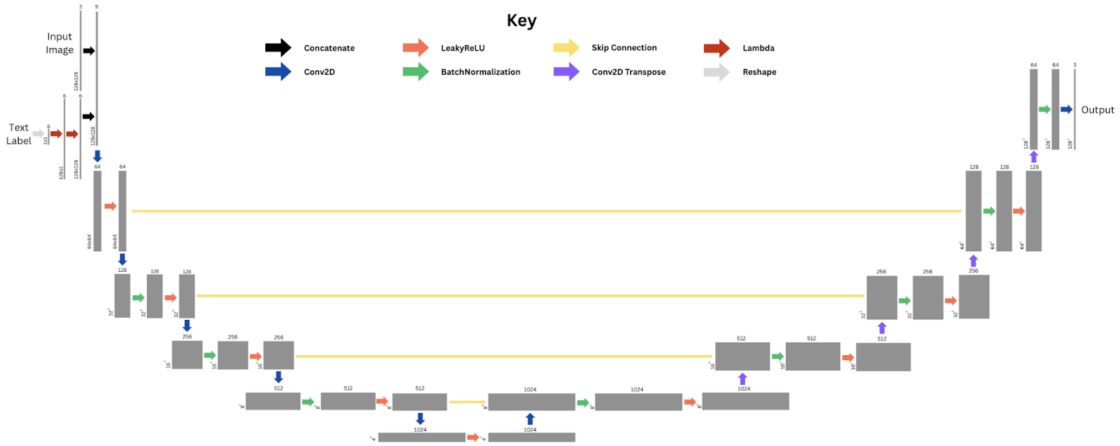

Figure 2: Figure 2: U-Net Architecture

Usually, the discriminator of a GAN is to predict if an image is real or fake, in the form of a supervised image classification problem with stacked convolutional layers and a single probability between 0 and 1. The critic in the C-WGAN is different from a discriminator; a Wasserstein distance (Earth Mover's distance) is calculated, outputting a score rather than a probability. Intuitively, the Wasserstein distance can be seen as the minimum work needed to transform one distribution to another [10]. The critic tries to estimate this distance, with a lower Wasserstein distance indicating a better match between the two distributions.

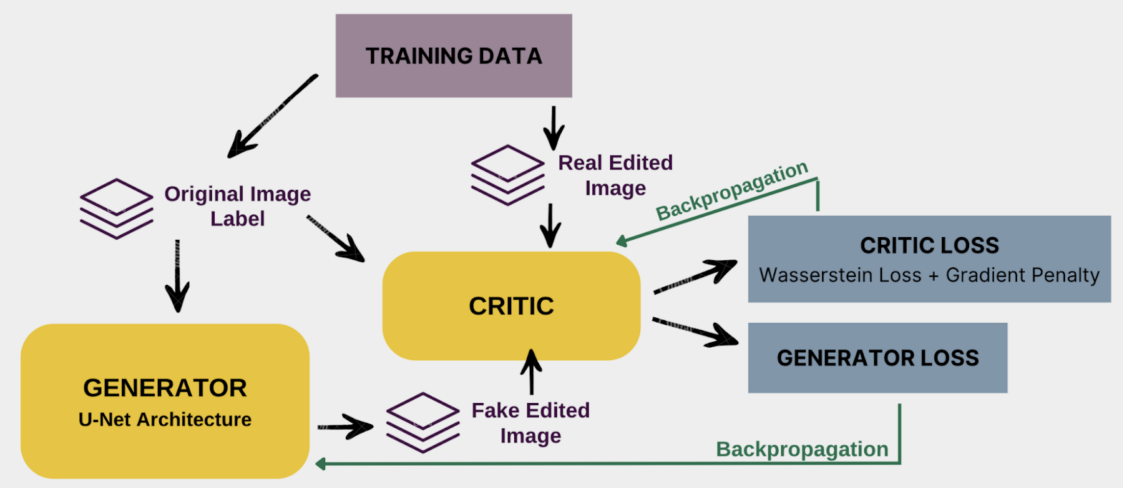

Figure 3: Figure 3:C-WGAN Workflow

3.3 Conditional Diffusion Model

The second generative framework used in this study is a Conditional Diffusion Model. Unlike GANs, diffusion models operate by adding and removing noise from images to learn specific features and data distributions.

The diffusion model comprises two key phases: forward diffusion and reverse diffusion. In the forward diffusion process, small amounts of Gaussian noise are added to the image over a large number of steps, so that eventually it is indistinguishable from standard Gaussian noise. This process can be mathematically defined by:

$$q(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}\big(\mathbf{x}_t;\, \sqrt{\bar{\alpha}_t}\,\mathbf{x}_0,\, (1-\bar{\alpha}_t)\mathbf{I}\big) \tag{1}$$

where $\mathbf{x}_0$ is the original image, $\mathbf{x}_t$ is the image at timestep t, and $\bar{\alpha}_t$ is a predefined noise schedule parameter that controls the variance of the Gaussian noise at timestep t.

In the reverse diffusion phase, the goal is to transform random noise into a meaningful output using a neural network [2]. The denoising step is implemented using a U-Net, the same architecture used for the C-WGAN's generator. The U-Net's role here is to predict the noise that was added to the edited image at a given diffusion step. A Mean Absolute Error (MAE) is used as the loss function to measure how accurately the predicted noise matches the actual noise added. The weights of the U-Net (the core of the diffusion model) are trained to minimize the difference between the predicted noise and the actual noise [1]. This is done iteratively during the training process. In each training step:

1. A batch of original images, corresponding edited images, and their style labels are fed into the model.
2. A random amount of noise is added to the edited images based on a randomly sampled diffusion timestep.
3. The U-Net takes the noisy edited image, the original image, the style label, and the diffusion timestep as input.
4. The U-Net predicts the noise that was added in step 2.
5. The Mean Absolute Error between the predicted noise and the actual noise is calculated.
6. The gradients of this loss with respect to the U-Net's trainable weights are computed.
7. An optimizer (like AdamW) uses these gradients to update the U-Net's weights, effectively teaching it to predict the correct noise.

An Exponential Moving Average (EMA) of the model weights is also maintained. This EMA model is used for generating images during inference and often produces more stable and higher-quality results.

To condition the diffusion process, an 'apply_condition' function was implemented in the U-Net's bottleneck pathway in order to incorporate the image's style label and time embeddings [2]. In our model, each image is paired with a condition label that specifies the desired color grade. This label is first projected into the same dimensionality as the model's internal image features. By adding this signal to the features at multiple stages in the network, it allows the network to align the denoising process with the desired color grading effect, ensuring that the output will match the specific color style. By training the U-Net to accurately predict noise conditioned on the original image, style label, and diffusion timestep, the reverse diffusion process can effectively transform random noise into images with the desired color correction applied.

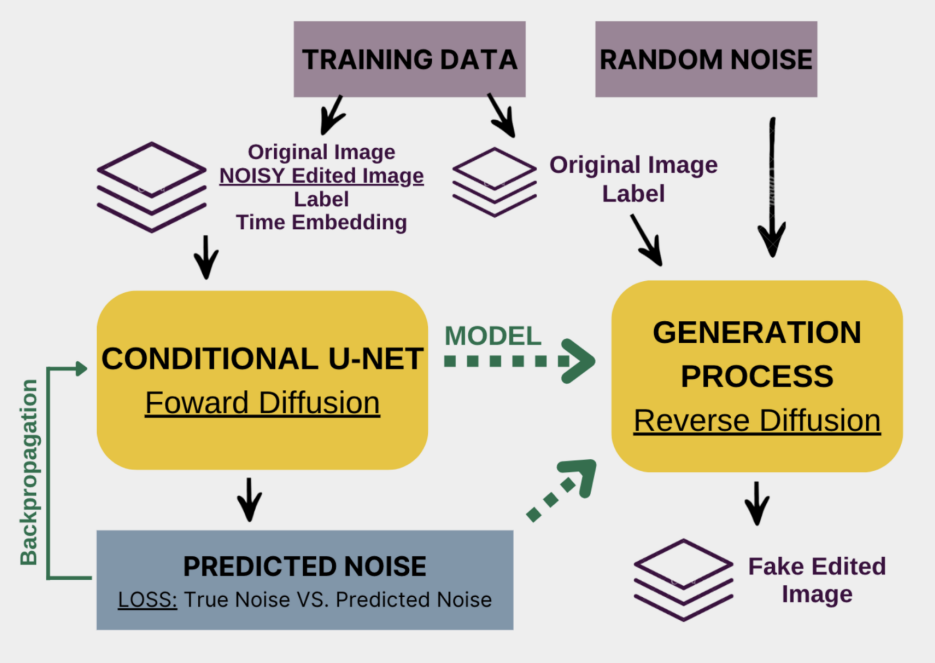

Figure 4: Figure 2: Conditional Diffusion Model Workflow

4 Experiments and Results

4.1 Qualitative Evaluation

Figure 5 and Figure 6 present the generated outputs from both generative models. The images generated from the C-WGAN are significantly better; there is a distinguishable difference in edits such as increased brightening, darkening, warm vivid colors, etc. In contrast, outputs generated by the Conditional Diffusion model exhibited only subtle changes, with muted effects and noticeable residual noise.

Figure 5: Color Edits Generated By C-WGAN

Figure 6: Color Edits Generated By Conditional Diffusion Model

4.2 Loss Curves

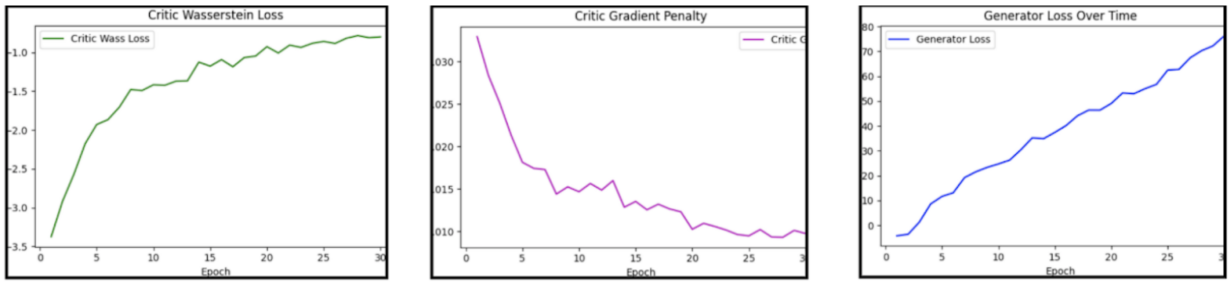

Figure 7: C-WGAN Loss Curves (Critic Wasserstein Loss, Critic Gradient Penalty, & Generator Loss)

There were three losses during the C-WGAN training. The Critic Wasserstein Loss (also known as the discriminator loss) utilizes the Wasserstein distance (mentioned in 3.1 methodology), starting from a negative value due to the critic loss being the [average critic score on real images] - [average critic score on fake images]. It steadily increased towards zero as training went on, signifying that the critic was able to maximize the difference in scores and differentiate less between real and generative images over time [7]. The Critic Gradient Penalty decreased significantly, indicating that training was stable. The penalty forces a Lipschitz constraint onto the network, ensuring that the critic's output will not change too much with small changes in input by guiding it to learn a smooth, well-behaved function for effective GAN training. However, the Generator Loss unexpectedly increased over training epochs. The generator is supposed to minimize the loss by generating samples that can fool the critic into giving a low Wasserstein distance estimation [7]. This trend, although typically indicative of poor generator performance, is justified in this context due to the distinct pixel color differences between the real and fake edited images.

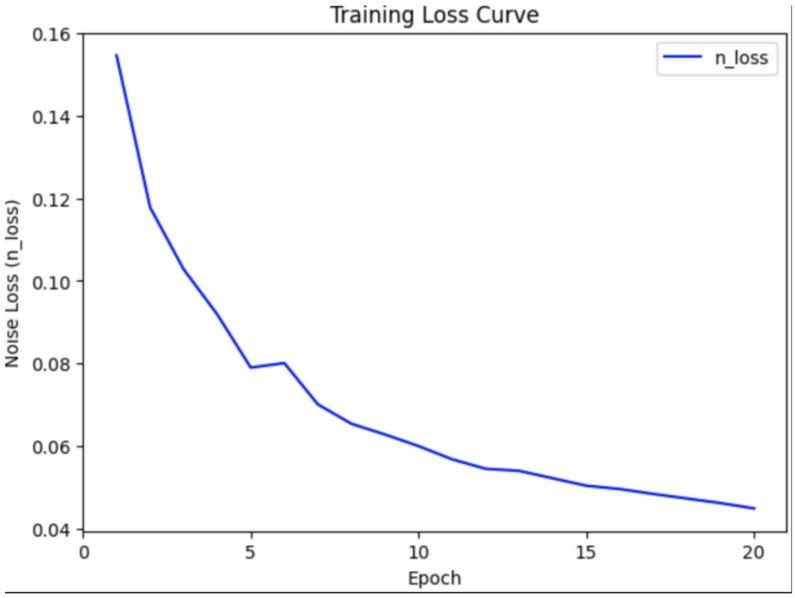

Figure 8: Conditional Diffusion Model Loss Curve (Noise Loss)

In the Diffusion Noise Loss Curve, the Mean Absolute Error (MAE) gradually decreased over 20 epochs. MAE is a metric that calculates the average magnitude of errors between paired observations. In this case, it shows how far off the model's predicted noise is from the actual noise. The lower the MAE, the more accurate the model [11]. Figure 8 shows that the diffusion model's noise prediction was not the main issue, the graph converges to a value less than 0.05.

4.3 Computational Efficiency and Evaluation Metrics

The Structural Similarity Index Measure (SSIM), Peak Signal-to-Noise Ratio (PSNR), inference time, and memory usage were all used as evaluation metrics to assess the C-WGAN's performance. The SSIM averaged at 0.779291, indicating that the images relatively look the same, although with ample space to improve [8]. The PSNR provides a numerical way to judge how well the model is replicating the desired color correction [5]. A good score usually lies between 30-50 dB, but in this study, it averaged at 25.9 dB. The average inference time per image was 44.71 milliseconds, showing the computational efficiency of the model. Finally, the memory usage of the generator was 170.34 MB with 44.6 million parameters, and 10.60 MB with 2.7 million parameters for the critic. While memory usage could be further optimized for mobile deployment, the current implementation is well-suited for server environments.

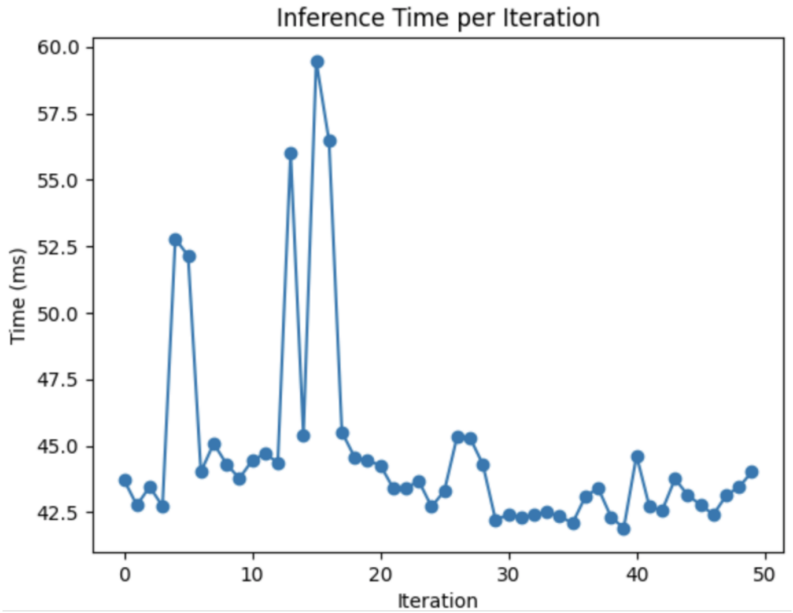

Figure 9: Inference Time, Measuring Computational Efficiency

5 Discussion

In this study's experiment, the GAN-based method produced better color-corrected and color-graded images compared to the Diffusion model based method. The GAN, though typically known for its unstable training process, performed visibly better with the given task. Although the Generator's loss kept increasing due to the critic effectively telling the difference between real and fake color grades, the output images justify that the images were color graded with the correct labels. The low PSNR score also supports the conclusion: with using a Mean Squared Error loss to calculate the difference between color intensities, the low score shows how the AI model was able to generate its own distinct color grade without copying the exact colors of the original edited image [5].

The diffusion model, while known to be mathematically more stable, could not produce the desired visual outputs. As seen in the results, the structure of the AI-generated images was preserved, but the style labels did not translate as strongly to tell what edit was produced.

In future work, different diffusion schedules (linear, quadratic, or even more advanced schedules like sigmoid or exponential) could be tested. Additionally, a stronger embedding technique for the style labels such as a small multi-layer-perceptron could be implemented to better capture the nuances of each edit. Modifications to the U-Net architecture itself could also be explored, especially incorporating attention mechanisms within the skip connection and decoder paths. Mean Squared Error or perceptual losses could be experimented with to see if they potentially achieve better image quality and more accurate style transfer compared to the current MAE loss.

6 Conclusion

The main contribution of this study focuses on comparing two generative models, C-WGAN and Conditional Diffusion Model, on the task of color grading, determining which model is better suited for the task.

For future work, color grading methods such as bleach bypasses, vintage filters, the teal and orange effect, and using triadic colors [17] can be added to the dataset to increase its complexity. Due to the difficulty of replicating these color grades, further improvements can also be made for both generative models. Another application is applying generative models to Lookup Tables (LUTs), a predetermined array of numbers that provide a shortcut to a specific color computation by transforming color input values to desired output values [12]. With only looking up and interpolating color values instead of individually dealing with saturation, hue, luminance, and other color edits, the model may be able to achieve similar color grading effects with much lower memory usage and more flexibility.

In summary, this study demonstrates that Conditional Wasserstein GANs are more effective than Conditional Diffusion Models for the task of color grading, producing more vivid and noticeable results while maintaining strong structural similarity and efficiency. These findings highlight the potential of generative AI not only as a technical tool, but also as a creative collaborator in shaping the future of digital cinematography and visual storytelling.

References

[1] Minshuo Chen, Song Mei, Jianqing Fan, and Mengdi Wang. An overview of diffusion models: Applications, guided generation, statistical rates and optimization. *arXiv preprint arXiv:2404.07771*, 2024. Section 5.

[2] David Foster. *Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play*. O'Reilly Media, 2019.

[3] imgaug contributors. imgaug: Image augmentation for machine learning experiments, 2020.

[4] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. *arXiv preprint arXiv:1611.07004*, 2017.

[5] JuliaImages. Structural similarity index, peak signal-to-noise ratio, 2025.

[6] Kaggle. Flickr 8k dataset. Accessed January 2025.

[7] Fabrizio Musacchio. Negative losses in wgan, July 2023. Blog post.

[8] Jimmy Nilsson and Tomas Akenine-Möller. Understanding ssim. *arXiv preprint arXiv:2006.13846*, 2020.

[9] Pepperdine Graziadio Business School. How the entertainment industry is making waves in the modern era. Online; accessed January 2025, 2022. Updated 2025.

[10] Akshay Sankar. Demystified: Wasserstein gans (wgan), September 2021. Towards Data Science.

[11] Peter Schneider. Mean absolute error – an overview, 2022. ScienceDirect Topics.

[12] StudioBinder. What are luts? the ultimate guide to color grading, October 2019.

[13] N. Tomar. What is unet?, January 2021. Analytics Vidhya.

[14] S. Treneska, E. Zdravevski, I. M. Pires, P. Lameski, and S. Gievska. Gan-based image colorization for self-supervised visual feature learning. *Sensors*, 22(4):1599, 2022.

[15] Variety Australia Staff. Digital innovation, the entertainment industry & how technology is reshaping it. *Variety Australia*, February 2025.

[16] VEGAS Creative Software. In depth: What are color correction vs. color grading in film. Accessed January 2025.

[17] Wondershare. 7 types of color grading for you to try. Accessed January 2025.

[18] Z. Zhan, D. Chen, J.-P. Mei, Z. Zhao, J. Chen, C. Chen, S. Lyu, and C. Wang. Conditional image synthesis with diffusion models: A survey. *arXiv preprint arXiv:2409.19365*, 2024.