Animation Frame Colorization with GANs

João Vitor Santiago Nogueira
Universidade Federal Fluminense - ICT
Universidade de São Paulo - IME
Email: jv_nogueira@id.uff.br

Leonardo de Oliveira Carvalho
Universidade Federal Fluminense - ICT
Email: leonardooc@id.uff.br

***Abstract*—This research paper presents an innovative approach to alleviate the labor-intensive nature of traditional 2D handmade animation utilizing artificial intelligence techniques. Specifically, we focus on refining the process of image colorization for 2D animations by employing Generative Adversarial Networks (GANs). The proposed method involves leveraging the power of GANs to paint a sequence of black and white frames in a manner that emulates the colors present in a single colored example.**

I. Introduction

The process of traditional 2D handmade animation is known to be a labor-intensive endeavor. To mitigate some of the challenges associated with this art form, artificial intelligence techniques have emerged as promising tools. This paper presents a refinement of image colorization methods for 2D animations by utilizing Generative Adversarial Networks (GANs).

A. GANs in Image Colorization

GANs have gained significant attention in the field of artificial intelligence due to their ability to generate realistic and high-quality images [1]. A GAN is a framework that consists of two neural networks: a generator and a discriminator. The generator network learns to produce images that resemble a given dataset, while the discriminator network is trained to distinguish between real and generated images. Through an iterative adversarial process, the generator improves its ability to generate increasingly realistic images that can fool the discriminator. This adversarial training leads to the creation of images that exhibit high visual fidelity and capture important statistical properties of the training data.

In the context of image colorization, GANs offer a powerful approach for inferring plausible color information from grayscale images. By training the generator network on a large dataset of colored images and corresponding grayscale versions, the GAN can learn the mapping between grayscale input and colorized output. This enables the generator to effectively colorize black and white images, producing visually appealing results that preserve the semantic meaning and coherence of the original content.

B. Related Works

Previous image colorization research primarily targeted real-world images, but 2D classical animation frames offer distinct challenges and opportunities. Unlike photos or videos, animation frames rely heavily on line art and silhouettes, with minimal emphasis on texture. This lack of texture necessitates specialized techniques that make effective use of available cues for accuracy, typically requiring a color reference to convey the intended artistic style.

Similar works done using GANs for image colorization include [2], [3], where the painting is guided by some user color hints. This is labor intensive and not accurate enough for painting hundreds of animation frames consistently.

Therefore, other methods were developed, such as by [4], a pioneer in the field, and from which this paper borrows most of its method, and [5], which introduce several constraints techniques that help each frame remain consistent.

There are other methods without using GANs, such as [6], [7], that use cell and graph based detection to color each frame, but that methodology has very different requirements and constraints.

C. Our Contribution

This paper expands on the groundwork laid by the aforementioned [4], whose methods are used as the foundation for this research.

We segmented the dataset into shots, using a single human-colored frame as reference per shot. That decision allowed us to further streamline some aspects of sequence reading, ensuring the reference frame isn't blank, AI generated, or from a different scene. By employing this technique we aimed to improve accuracy and decrease the effects of compounding errors and over-fitting. This adjustment also aligns the training process more closely with desired working conditions, where only a single colored frame is available per shot.

We improved the decoder in the network's auto-encoder by addressing the checkerboard pattern effect [8]. This involved replacing transposed convolutions with up-sampling and regular convolutions.

The outcomes of this research contribute to the advancement of AI-assisted techniques in the field of 2D animation, facilitating the creation of visually captivating and expressive hand-drawn animations while reducing the labor-intensive nature of the process. Our source code can be found over at https://github.com/JoaoVitorSantiagoNogueira/tcc-testes.

II. Methodology

A. Data Collection and Preprocessing

Our methodology begins with dataset acquisition for training and evaluating our image colorization model. We gather real 2D animated movies and series, providing a diverse source

of animation frames. These frames are sequentially extracted to maintain the original content's temporal coherence.

The training dataset consisted of 801 images borrowed from three animation sources. This is relatively small compared to similar papers that used 84,000 [4] and 11,000 [5] frames, each from a single source. Such limitations were due to time and processing power constraints. While the results exhibit a slight compromise in quality, they serve as a demonstration of potential with future development and additional resources.

Following the data extraction process, individual animation frames undergo a transformation to yield two distinct versions: the original frame retaining its native colors and a corresponding black and white rendition. This monochromatic version can be represented either in grayscale or as line art. While employing grayscale yields superior results, opting for line art better aligns with the real-world use case. The extraction of line art can be achieved through methodologies such as the one proposed in [9]. This preprocessing step ensures the availability of both color information, utilized as a reference for color input and output, and grayscale representation, serving as a foundation for training the Generative Adversarial Network (GAN) model.

The dataset comprised frames from diverse sources, in contrast to other studies that focused on a singular source. This approach, while potentially reducing the quality of training examples, offers the advantage of accommodating a wider range of animation and coloring styles, leading to improved results when presented with new and varied art styles.

B. Shot-based Segmentation

One of the new techniques employed by this paper is to keep each frame sequence divided into 'shots', this process helps to retain consistency between elements within subsequent frames and ensure coherent colorization results. A shot refers to a continuous sequence of frames that depict a particular scene or action without any major cuts or transitions. By segmenting the frame sequence into shots, we can capture the context and visual consistency within each shot independently. This process was done manually, but can be done automatically with shot transition detection techniques [10], [11].

This segmentation allows the colorization model to focus on preserving the visual coherence within individual shots, which ensures that the colorization process considers the specific context within each shot, rather than having it guess how to color new frames once there is a transition. Once the frame sequence is divided into shots, we treat each shot as a unit for the subsequent colorization processing. This approach facilitates the accurate transfer of colors from the reference frames to the corresponding black and white frames.

C. Generative Adversarial Network Architecture

To accomplish the task of image colorization for 2D animation frames, a Generative Adversarial Network (GAN) architecture is employed. Inspired by the paper "Automatic Temporally Coherent Video Colorization," the adopted network architecture is kept the same for both the generator and discriminator components of our GAN.

By employing the same network architecture as the base paper, we establish a solid foundation to the colorization methodology. This architecture has proven effective in and ensures compatibility and consistency with existing research in the field. The single change to the network structure was the aforementioned replacement of the transposed convolutional layer by an upsample and a convolutional layer in the decoder to avoid checkered patterns [8].

1) Generator Network: The generator network takes a single RGB-A image as input. Each image is reduced to a 256 by 256 bit-map image, to standardize the size and shape of the images to fit the input network architecture's input.

The color channels are taken from the very first frame of each shot, and serve as the color reference for all other frames from the same shot.

The inspiring paper uses a black frame as color reference 50% of the time to avoid overfitting and mask scene transitions. However, our methodology omits this process as it appears to exacerbate overfitting and we have a well defined scene transition. This occurs because the network comes to rely on specific stylistic features from the training set to guess colors rather than learning to extracting them from a reference.

The alpha channel is a black and white image (either grey scale of line art) of the frame to be colored, and changes to account for each frame. Those are the images used as the position reference for our method.

The architecture of the generator consists of convolutional layers, auto-encoders (downsampling and upsampling) layers, and residual blocks. The convolutional layers extract hierarchical features from the input grayscale image, while the downsampling and upsampling layers decrease the resolution to make it easier to process and then increase it to return it's original size. The residual blocks help to preserve the details and ensure a smooth color transition between adjacent pixels. The architecture can be seen on the upper half of Figure 1.

2) Discriminator Network: The discriminator network is responsible for distinguishing between real color animation frames and colorized frames generated by the generator. It aids in training the generator to produce more realistic and images by providing feedback on the quality of the generated images.

The discriminator is trained on real colored frames and the generated ones, to teach it how to recognize features from the real images, and therefore, the inconsistencies in the generated images.

The discriminator architecture also comprises convolutional layers, which extract features from the input frames, followed by fully connected layers for classification. The architecture can be seen on the bottom half of Figure 1.

The generator and discriminator networks are trained in an adversarial manner, where the generator aims to generate colorized frames that can fool the discriminator, while the discriminator strives to accurately differentiate between real and generated frames. This adversarial training process promotes the improvement of the generator's ability to generate visually compelling and realistic colorizations.

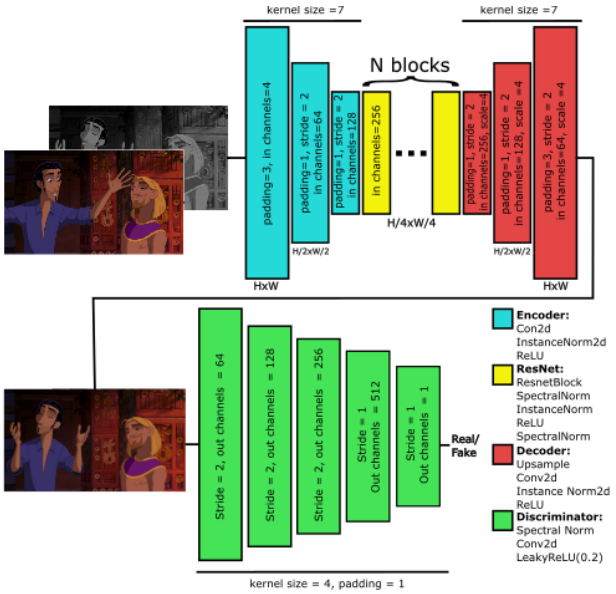

Fig. 1. Network Structure and method pipeline.

3) Loss Function and metrics: In order to train the network, a Loss Function needs to be defined. To remain consistent we opted to use the same as defined in [4], as making too many changes would make harder to isolate each of their contributions. The function itself is made of other loss functions, combined through a weighted sum, these were, adversarial loss (generated due to GANs errors), content loss and style loss generated by feeding the frames into a pre-trained network [12] that compares content (shape, silhouettes, color) and style (shading, strokes, details), and L1 distance.

To evaluate the performance of our image colorization model, we utilize three prominent error metrics: Peak Signal-to-Noise Ratio (PSNR) [13], Fréchet Inception Distance (FID) [14], and Structural Similarity Index (SSIM) [15]. These metrics serve as golden standards in many similar projects and provide quantitative measures for comparing our final results against ground truth and state-of-the-art colorization techniques.

The combination of these error metrics allows us to perform a comprehensive evaluation of our image colorization model. PSNR provides insight into the pixel-wise accuracy, FID assesses the overall similarity to real images, and SSIM gauges the structural and perceptual similarity. By comparing our results against these established standards, we can gauge the effectiveness of our colorization approach and identify areas for potential improvements.

III. Results

Some animation frames' final results are displayed in figures 2 and 3. Each figure presents the method-colored frame on the left and the ground-truth on the right.

The first example, in Figure 2 the generated and real frame closely resemble each other, with some minor differences, such as the color of the bracelets and earrings being tan colored, matching the character's skin tone, rather than the expected teal.

The second example, shown in Figure 3, shows two frames that, while similar in structure and shading, are very distinct in hue. The generated frame is very desaturated in comparison to it's ground-truth.

Fig. 2. Method Applied to The Road to El Dorado

Fig. 3. Method Applied to Spirited Away

Figure 4 shows the evolution of the Generator's error. It quickly decreases through the first 20 generations, then goes on a slower descent, fluctuating and finding a few smaller minimums.

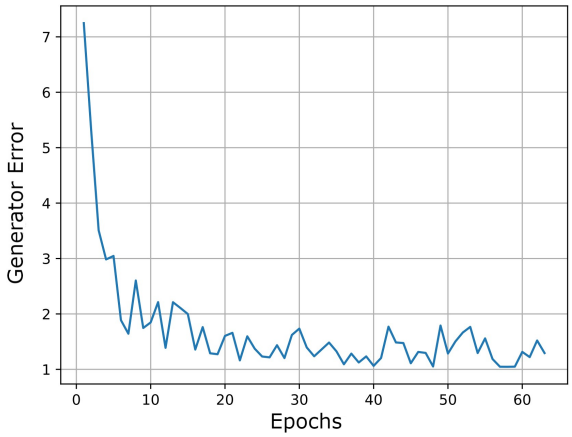

Fig. 4. Generator error though the Epochs

Table I shows the results obtained with the proposed method comparing to similar methods by others. The bold values represent the best score achieved among all methods. Our methodology achieved the best results in the PSNR department, while wielding the worse in the FID and middling scores in the SSIM category. These results are further discussed in the discussion segment.

TABLE I
METRICS COMPARISON

	SSIM	PSNR	FID
Our method	0.76	**29.87**	104.10
Baseline in [4]	0.72	14.15	20.69
[4]	0.78	17.38	**9.29**
[5]	**0.86**	22.41	27.93

A video showcase of the method applied to the whole validation set can be found over in the supplementary material.

IV. CONCLUSION

The research on Generative Adversarial Networks (GANs) for animating frames shows promising but imperfect results. Areas for improvement include segmentation techniques, color adjustment, loss functions, and generator structure.

A. Result Analysis

Our paper builds upon prior work, and the results reflect this expansion. The consistent SSIM score indicates that visual similarity was maintained despite the introduced alterations.

The decrease in the FID Score can be attributed to our small training set, a limitation imposed by time constraints. This constraint hindered the model's ability to generalize effectively across a broader range of animations. However, within frames from "The Road to El Dorado," which constituted the majority of the training set, notably superior results were achieved, suggesting room for improvement.

The increase in the PSNR score demonstrates the efficacy of scene segmentation and using the first frame of the sequence. This approach enabled the model to learn to paint each frame with reduced noise, mitigating uncertainties related to scene transitions and the compounding errors associated with using one generated frame as a reference for another.

B. Future Works

Some techniques that could improve the quality of the method can still be tested. Some pre-processing methods, like background removal, show potential for practical application. Color adjustment, transforming input and output images based on references rather than memorized palettes, offers promise in reducing overfitting.

Additional improvements can be achieved by altering loss functions or transitioning to a WGAN [16] model for enhanced convergence. The generator's structure can be further modified, with emerging Denoise-diffusion [17] and Diffusion-Gan [18] models presenting compelling alternatives.

C. Closing Thoughts

Although not yet on par with human-made works, ongoing advancements and proposed enhancements suggest the viability of this approach in the future. Careful consideration of ethical implications in human-machine interactions within the artistic industry moving forward is essential, as seen with the recent backlash due to the rise in quality of AI generated art.

REFERENCES

[1] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," 2014. [Online]. Available: https://arxiv.org/abs/1406.2661

[2] C. Furusawa, K. Hiroshiba, K. Ogaki, and Y. Odagiri, "Comicolorization: Semi-automatic manga colorization," 2017. [Online]. Available: https://arxiv.org/abs/1706.06759

[3] Y. Qu, T.-T. Wong, and P.-A. Heng, "Manga colorization," *ACM Transactions on Graphics (ToG)*, vol. 25, no. 3, pp. 1214–1220, 2006.

[4] H. Thasarathan, K. Nazeri, and M. Ebrahimi, "Automatic temporally coherent video colorization," in *2019 16th conference on computer and robot vision (CRV)*. IEEE, 2019, pp. 189–194.

[5] M. Shi, J.-Q. Zhang, S.-Y. Chen, L. Gao, Y.-K. Lai, and F.-L. Zhang, "Deep line art video colorization with a few references," *arXiv preprint arXiv:2003.10685*, 2020.

[6] R. Nascimento, F. Queiroz, A. Rocha, T. Ing Ren, V. Mello, and A. Peixoto, "Computer-assisted coloring and illuminating based on a region-tree structure," *SpringerPlus*, vol. 1, p. 1, 03 2012.

[7] K. Sato, Y. Matsui, T. Yamasaki, and K. Aizawa, "Reference-based manga colorization by graph correspondence using quadratic programming," in *SIGGRAPH Asia 2014 Technical Briefs*, 2014, pp. 1–4.

[8] A. Odena, V. Dumoulin, and C. Olah, "Deconvolution and checkerboard artifacts," *Distill*, 2016. [Online]. Available: http://distill.pub/2016/deconv-checkerboard

[9] J. Canny, "A computational approach to edge detection," *IEEE Transactions on pattern analysis and machine intelligence*, no. 6, pp. 679–698, 1986.

[10] J. Bescós, G. Cisneros, J. M. Martínez, J. M. Menéndez, and J. Cabrera, "A unified model for techniques on video-shot transition detection," *IEEE transactions on multimedia*, vol. 7, no. 2, pp. 293–307, 2005.

[11] T. Souček and J. Lokoč, "Transnet v2: An effective deep network architecture for fast shot transition detection," *arXiv preprint arXiv:2008.04838*, 2020.

[12] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[13] F. A. Fardo, V. H. Conforto, F. C. de Oliveira, and P. S. Rodrigues, "A formal evaluation of psnr as quality measurement parameter for image segmentation algorithms," *arXiv preprint arXiv:1605.07116*, 2016.

[14] Y. Benny, T. Galanti, S. Benaim, and L. Wolf, "Evaluation metrics for conditional image generation," *International Journal of Computer Vision*, vol. 129, pp. 1712–1731, 2021.

[15] J. Nilsson and T. Akenine-Möller, "Understanding ssim," *arXiv preprint arXiv:2006.13846*, 2020.

[16] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," 2017.

[17] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 6840–6851. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf

[18] Z. Wang, H. Zheng, P. He, W. Chen, and M. Zhou, "Diffusion-gan: Training gans with diffusion," *arXiv preprint arXiv:2206.02262*, 2022.