Vitexco: Exemplar-based Video Colorization using Vision Transformer

Duong Thanh Tran
Computing Fundamental Department
FPT University
Ho Chi Minh City, Vietnam
duongttse160185@fpt.edu.vn

Nguyen Doan Hieu Nguyen
Computing Fundamental Department
FPT University
Ho Chi Minh City, Vietnam
nguyenndhse161261@fpt.edu.vn

Trung Thanh Pham
Computing Fundamental Department
FPT University
Ho Chi Minh City, Vietnam
trungptse160030@fpt.edu.vn

Phuong-Nam Tran
Computing Fundamental Department
FPT University
Ho Chi Minh City, Vietnam
namtpse150004@fpt.edu.vn

Thuy-Duong Thi Vu
Computing Fundamental Department
FPT University
Ho Chi Minh City, Vietnam
duongvtt9@fe.edu.vn

Duc Ngoc Minh Dang*
Computing Fundamental Department
FPT University
Ho Chi Minh City, Vietnam
ducdnm2@fe.edu.vn

***Abstract*—In the field of image and video colorization, the existing research employs a CNN to extract information from each video frame. However, due to the local nature of a kernel, it is challenging for CNN to capture the relationships between each pixel and others in an image, leading to inaccurate colorization. To solve this issue, we introduce an end-to-end network called Vitexco for colorizing videos. Vitexco utilizes the power of the Vision Transformer (ViT) to capture the relationships among all pixels in a frame with each other, providing a more effective method for colorizing video frames. We evaluate our approach on DAVIS datasets and demonstrate that it outperforms the state-of-the-art methods regarding color accuracy and visual quality. Our findings suggest that using a ViT can significantly enhance the performance of video colorization.**

***Index Terms*—image colorization, video colorization, exemplar-based, vision transformer**

I. Introduction

Image colorization has long been an exciting topic for researchers, with numerous techniques being applied to achieve more realistic and colorful results. With the development of image colorization, attention has turned to video colorization. However, colorizing black-and-white videos is more challenging than colorizing black-and-white images due to the need to maintain consistency frame-by-frame, preventing colors from changing dramatically between adjacent frames. Recurrent networks are well-suited to this task, and many studies have utilized this framework for video colorization in conjunction with CNN [1] to extract information from each video frame. However, the local property of CNN's kernel, which cannot maintain relationships between pixels that are too far apart in each frame, may result in incorrect colorization when two objects with a strong relationship are far apart in a frame. The introduction of the ViT [2] has solved this problem. In this paper, we introduce an end-to-end network called Vitexco for colorizing videos which can make the model maintain the relationships between pixels. Additionally, we explore the effectiveness of the ViT in improving the accuracy of video colorization and compare its results with other approaches. We also examine how different hyper-parameters affect the performance of the ViT in video colorization. The main contributions of this work are summarized below:

* *Corresponding author: Duc Ngoc Minh Dang (ducdnm2@fe.edu.vn)*

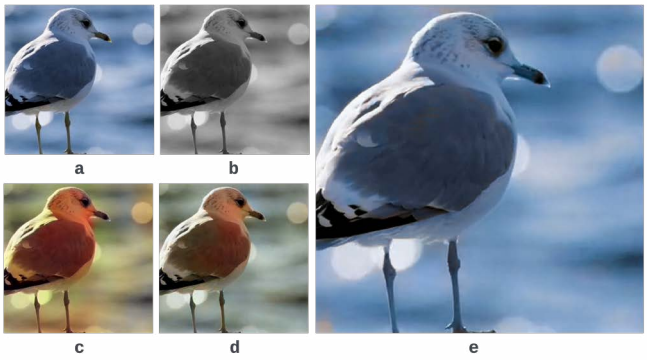

Fig. 1. The result of colorizing images using different methods. a) Ground truth b) Grayscale c) Zhang *et al.* [3], d) Victoria *et al.* [4] e) Our Vitexco

- We introduce a network architecture that incorporates a ViT [2] backbone and complementary subnetworks to extract information from each video frame, improving colorization outcomes.
- We create a high-quality dataset for training the model with a large diversity of things, people, colors, and others. This dataset is a combination of Hollywood2, ImageNet-1k, and extra videos that we collected from Pixabay.

The rest of this paper is structured as follows. The related work is presented in Section II. The proposed methodologies are presented in Section III. Sections IV and V present and analyze the preliminary results. Finally, the conclusion and potential future work are concluded and listed in Section VI.

II. Related work

A. Interactive Colorization

One of the earliest and most straightforward methods for colorization was using user-provided hints [5]–[7]. These hints could be color points, strikes, or scribbles. The underlying principle of this method was based on the assumption that pixels that were in proximity to one another and thus belonged to the same object would share similar colors. The hints could be local and global [8], which were fed into an overall network; some models leveraged ViT [9] and used user instruction to colorize images. Although this method was relatively simple, it had proven effective in producing colorized images and videos. However, user-guided methods were unsuited for video colorizing tasks due to the significant human effort and aesthetic skills required to produce colorful images. The complexity of these techniques necessitated a considerable investment of time and resources, which may have needed to be more practical for video colorization projects.

B. Exemplar-based colorization

Another method involved transferring colors from a reference image to the target grayscale image. The reference image was selected based on its semantic similarity to the grayscale image. This approach could also be applied to video colorization by colorizing each frame individually. However, this method could lead to flickering issues if not implemented carefully. To address this problem, researchers introduced various architectures to improve the quality of colorized video. Deep exemplar-based Video colorization [10] consisted of 2 subnetworks: the similarity subnetwork and the colorization subnetwork. The recurrent mechanism was applied inside the colorization subnetwork to ensure consistency between adjacent frames. BiSTNet [11] created a Bidirectional Temporal Fusion Block for better results in transferring colors from reference images to video frames.

C. Fully Automatic Colorization

One way to colorize grayscale images was through training a deep neural network to learn the mapping. Several approaches have been proposed, including the use of Convolutional layers to encode and decode the input grayscale image, as demonstrated in [12], [13]. Other methods [4], [14] employed GAN to generate the color images. Recently, [15] applied Swin Transformer [16] instead of CNN for a better result in colorizing images because it outperformed traditional CNN in computer vision tasks. However, this method had several challenges, including the need for a large dataset and a deep network, making training difficult. Additionally, the model might have generalized poorly to new images, mainly if the training dataset was limited. Another limitation of this approach was that the resulting colorization might not have been easily customized. Unlike exemplar-based methods, which allowed users to specify the desired color of specific objects or regions, the deep learning approach was less flexible. Therefore, it might not have been suitable for applications requiring customized colorization.

III. Methodology

This section outlines the overall Vitexco network for training video colorization. We begin by introducing the Correspondence Subnet $\mathcal{A}$ as an information extractor. We then describe the Colorization subnet $\mathcal{B}$, the primary component responsible for the colorization process. Lastly, we provide an overview of the losses utilized in this study.

Vitexco is built on top of Deep Exemplar-based Video Colorization [10] with the improvement of integrating the ViT [2] to capture all relationships between pixels in each frame. To improve the relationships between individual pixels and the rest of the image, Vitexco incorporates a Correspondence subnet that utilizes a ViT backbone. Instead of relying on the VGG19 [2] as in [10], we introduce a Feature Transform block. To simplify the colorization process for each video frame, we utilize the LAB color space and adjust only the A and B channels, which provide distinct advantages.

A. Vitexco

Figure 2 illustrates the overall architecture of our Vitexco. Let $X^l = \{x_0^l, x_1^l, x_2^l, ..., x_{T-1}^l\}$ denote a grayscale video, where $x_t^l \in \mathbb{R}^{H\times W\times 1}$ represents the frame at time t, $H \times W$ indicates the size of each frame, and T is the total number of frames in the video. Our goal is to generate a color video $\hat{X}^{lab} = \{\hat{x}_0^{lab}, \hat{x}_1^{lab}, \hat{x}_2^{lab}, ..., \hat{x}_{T-1}^{lab}\}$ where $\hat{x}_t^{lab} \in \mathbb{R}^{H\times W\times 3}$ represents the frame at time t in the LAB color space. Each x_t^l has a reference image $y^{lab} \in \mathbb{R}^{H\times W\times 3}$ used to guide the coloring process. To obtain the reference image, we compare the video's grayscale image with the image in the ImageNet dataset [17] to measure their similarity. Vector features are extracted from both images using ViT, and their similarity is computed using cosine similarity. The reference image with the highest similarity score is chosen as the best match. After the best reference image is selected, both inputs are converted to the LAB color space for colorization. In this color space, the l channel represents lightness, the a channel represents the red/green value, and the b channel represents the blue/yellow value.

To begin colorization, we pass the grayscale video X^l through a Correspondence subnet frame-by-frame. The Correspondence subnet aligns the reference image y^{lab} to each video frame based on semantic similarity. The output of the Correspondence subnet consists of the warped color $\mathcal{W}$ and a confidence map $\mathcal{M}$, which measures the reliability of the correspondence between the reference image and the current frame. Although this method creates a base color for the image, the accuracy of the color warping may vary across different regions of the image, and it does not utilize any information from the previous frame.

To overcome the issue above, we employ the Colorization Subnet to select the well-matched colors and propagate them properly. The network takes in four inputs: the grayscale input x_t^l, the warped color map $\mathcal{W}$ and the confidence map $\mathcal{M}$, and the colorized previous frame x_{t-1}^{lab}. Through this process, we can achieve more accurate and visually appealing colorizations for the video frames.

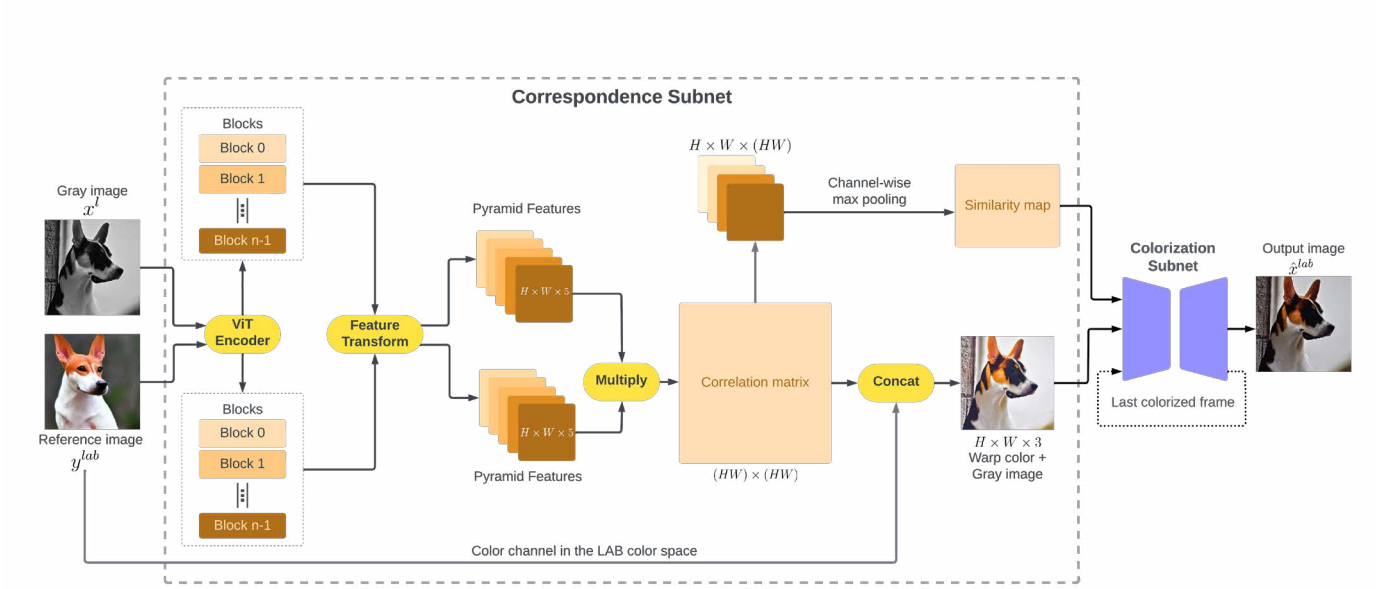

Fig. 2. Overall architecture of the network. (HW is the multiplication of W - width and H - height of the video frame)

B. Correspondence subnet

ViT backbone: After choosing the reference image with the highest semantic similarity to a video frame using the pre-trained ViT, the gray-scale video frame x_t^l followed by a reference image y^{lab} are passed through the Correspondence subnet to build the semantic correspondence between them. In Deep Exemplar-based Video Colorization [10], Zhang *et al.* used a VGG19 pre-trained on image classification. However, for better information extraction from video frames, we propose using ViT [2] pre-trained on image classification.

To leverage the features of the ViT, we make modifications to the Correspondence Subnet [10]. Specifically, we replace the input feature maps from 5 layers of the VGG19 pre-trained model with the 5 token embeddings of the ViT. This allows us to take advantage of the ViT features and improve the network's performance for video colorization. However, the Colorization Subnet requires various input features that may differ in size and shape. This presents a challenge when integrating the ViT output from the Correspondence Subnet.

Feature Transform Subnet: To address the issue mentioned above, we develop a Feature Transform Subnet, which aims to transform the ViT output features to match the subsequent subnet's input features. Figure 3 illustrates the feature transform, which consists of five decoder models. Each decoder model contains upsampling layers. Using these layers facilitates the transformation of features to learn the intrinsic characteristics of the video frame.

Overall, our Correspondence Subnet employs the ViT to produce a correlation matrix $\mathcal{M} \in \mathbb{R}^{HW \times HW}$, which is more efficient than the Correspondence subnet in [10] that utilizes VGG19 features.

C. Colorization subnet

This subnet gets the output from the Correspondence Subnet. More specifically, the network receives four inputs: the gray-scale video frame x_t^l, the warped color map $\mathcal{W}$, the confidence map $\mathcal{M}$ and the predicted previous frame $\hat{x}_{t-1}^{lab}$. We call this subnet $\mathcal{B}$ and have the formula:

$$\hat{x}_t^{lab} = \mathcal{B}(x_t^l, \mathcal{A}(x_t^l, y^{lab})|\hat{x}_{t-1}^{lab}, y^{lab}) \tag{1}$$

D. Losses

1) L1 Loss: This loss function calculates the differences in color between the colorized frame $\hat{x}_t^{lab}$ and the corresponding ground truth frame x_t^{lab}. Minimizing this loss helps the video prediction model generate more accurate predictions that closely resemble the original video frames.

$$L_{L1} = ||\hat{x}_t^{lab} - x_t^{lab}||_1 \tag{2}$$

2) Adversarial Loss: Incorporated as a constraint in the video colorization process to enhance the realism of the colorized videos. In line with the approach adopted in Deep Exemplar-based Video Colorization [10], a video discriminator is employed instead of an image discriminator. This decision is motivated by the observation that flickering in colorized videos can be easily detected compared to real videos. Using a video discriminator, the model can learn to colorize videos while minimizing the flickering issue. This approach effectively improves the quality and visual fidelity of the colorized videos.

$$\begin{aligned} L_{adv}^{\mathbb{G}} = \mathbb{E}_{(\hat{x}_{t-1},\hat{x}_t)}(D(\hat{x}_{t-1},\hat{x}_t) - \mathbb{E}_{(x_{t-1},x_t)}D(x_{t-1},x_t) - 1)^2 \\ +\mathbb{E}_{(x_{t-1},x_t)}(D(x_{t-1},x_t) - \mathbb{E}_{\hat{x}_{t-1},\hat{x}_t}D(\hat{x}_{t-1},\hat{x}_t) + 1)^2 \end{aligned} \tag{3}$$

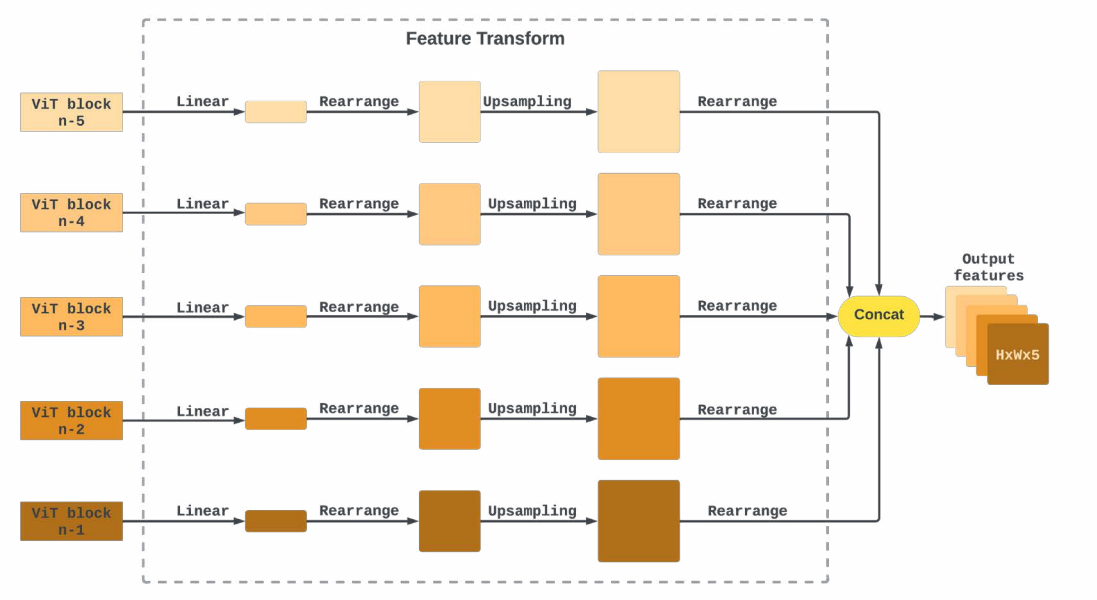

Fig. 3. The architecture of Feature Transform subnet.

3) Perceptual Loss: First introduced in [18] for neural style transfer. It quantifies the dissimilarity between the high-level characteristics of 2 images, such as edges, textures, and shapes. This helps the neural network learn how to transform an image into another with the desired features. We use this loss to make the output perceptually plausible. This loss penalizes the semantic difference of the predicted frame $\hat{x}_t$ and the correspondent ground truth frame x_t

$$L_{perc} = ||\phi^L_{\hat{x}} - \phi^L_x||^2_2 \tag{4}$$

where ϕ represents the output feature of the Feature Transform block in the Correspondence subnet.

4) Contextual Loss: Contextual Loss measures the difference between the high-level characteristics of 2 images, considering the context of the entire image and edges, textures, and shapes. In [19], the contextual loss was used to train a neural network to transfer the style of one image to another while preserving the overall structure and meaning. Given the ability to compare contextual meaning between a video frame and a reference image, the Contextual Loss is suitable for encouraging the colors in the predicted frame $\hat{x}_t$ to be similar to those in the corresponding reference image. We first compute the cosine similarity $d^L(i,j)$ between each pair of feature tensors $\phi^L_x(i)$ and $\phi^L_x(j)$ and $\hat{d}^L(i,j) = d^L(i,j)/(min_k d^L(i,k) + \epsilon)$ where $\epsilon = 1e-5$. We have the formula for contextual loss:

$$A^L(i,j) = \underset{j}{\text{softmax}}\left(1 - \hat{d}^L(i,j)/h\right) \tag{5}$$

where i, j represents the indices of feature tensor in the output features of the Feature Transform Subnet and bandwidth parameter $h = 0.1$.

5) Temporal Consistency Loss: A temporal consistency loss [20] is incorporated into the colorization process to account for temporal coherence in video colorization. This loss function explicitly penalizes color changes that occur along the flow trajectory. By incorporating this constraint, the model can ensure that the color changes in the video frames are consistent over time, thereby improving the overall visual quality and coherency of the colorized video. This approach is particularly effective in addressing flickering in video colorization tasks, as it encourages the model to produce smoother and more natural-looking color transitions. To reinforce temporal consistency in our model, we utilize the DeepFlow algorithm [21] to incorporate the optical flow information of each pair and method [22] to generate an occlusion mask. By leveraging these flow and mask components, we calculate the loss and improve the overall visual quality of our output.

$$L_{temp} = ||m_{t-1} \odot W_{t-1,t}(\hat{x}^{lab}_{t-1}) - m_{t-1} \odot \hat{x}^{lab}_t|| \tag{6}$$

where $W_{t-1,t}$ denotes the flow from x_{t-1} to x_t, m_{t-1} is the mask and $\odot$ denotes Hadamard product operation.

6) Smoothness Loss: This loss was also introduced in [10] to encourage spatial smoothness. In video colorization tasks, it is often assumed that neighboring pixels of the predicted frame $\hat{x}_t$ should have similar color values if they have similar colors in the corresponding ground truth frame. This loss function is incorporated into the colorization process to enforce this constraint. It is the difference between a pixel's color and the color of its 8-connected neighborhood. By minimizing this loss, the color transitions in the video frames are smooth, thereby improving the overall visual quality of the colorized video. This approach is particularly effective in producing natural-looking color transitions and reducing the occurrence of color artifacts in the colorized video.

$$L_{smooth} = \frac{1}{N}\Sigma_{c\in\{a,b\}}\Sigma_i(\hat{x}^c_t(i) - \Sigma_{j\in\mathbb{N}(i)} w_{i,j}\hat{x}^c_t(j)) \tag{7}$$

where N is the number of samples in a training step.

7) Overall Loss: We combine all the losses to create the final loss that we want to optimize

$$L = \lambda_{L1}L_{L1} + \lambda_{adv}L_{adv} + \lambda_{perc}L_{perc} + \lambda_{ctx}L_{ctx} + \lambda_{temp}L_{temp} + \lambda_{smooth}L_{smooth} \quad (8)$$

where λ is the weight of each loss function.

IV. Implementation Detail

Dataset: The study employs the Hollywood2 dataset for training [23], which is filtered to exclude low-quality and black-and-white videos, resulting in a corpus of 337 videos. We added 111 high-quality videos from Pixabay to diversify the data categories. Every video is sampled every 2 frames, averaging 55 frames per video. From these frames, we obtain an average of 54 pairs per video, each consisting of two consecutive frames annotated as the previous frame and the current frame. With each pair, we query the five most similar images from the ImageNet dataset [17]. To obtain that, we first extract embeddings of all the images in the ImageNet dataset using the ViT model [2] and store them in the database. Then, we compare the extracted embedding of the current frame to all the embeddings of the ImageNet dataset [17] using cosine similarity scores and retrieve the five highest scores. As a result, we obtained 24,609 samples for training. We use the DAVIS dataset [24] for evaluation and prepare additional information for the dataset as we do for the training dataset.

Hyper-parameters: We use tiny version of ViT [2]. The weights of the losses in our model were set as follows: $\lambda_{L1} = 2.000$, $\lambda_{adv} = 0.200$, $\lambda_{perc} = 0.005$, $\lambda_{ctx} = 0.500$, $\lambda_{temp} = 0.020$, and $\lambda_{smooth} = 5.000$. We employ an AdamW optimizer with a learning rate of 1×10^{-5} and $\beta_1 = 0.500$, $\beta_2 = 0.999$ for both the generator and discriminator. Using the gradient accumulation technique, the model was trained for 100,000 iterations (steps) with a batch size of 4. The model is trained on 2 GPU NVIDIA RTX 4090 for increasing training speed.

V. Experiments

Evaluation metrics: The evaluation of the performance of our model is challenging since the aesthetic appeal of a colorized picture or video is subjective and cannot be objectively quantified. Nonetheless, we conducted several assessments to gauge the proficiency of our model. We employ various standard metrics for evaluation, including:

- Peak Signal-to-Noise Ratio (PSNR) for estimating the ratio between the maximum possible intensity of images to the error in color estimation.
- Structural Similarity Index Measure (SSIM) [25] for indicating the similarity between two images related to structural information.
- Learned Perceptual Image Patch Similarity (LPIPS) [26] accounting for the distance between two images regarding human perception.

We apply these metrics on each video frame-by-frame and then calculate the average to get the final result.

TABLE I
Comparison with other models

Models	PSNR ↑	SSIM ↑	LPIPS ↓
Iizuka *et al.* [27]	23.88	0.947	0.176
Zhang *et al.* [3]	22.57	0.947	0.106
Zhang *et al.* [8]	24.88	0.949	0.116
Su *et al.* [28]	25.65	0.951	0.082
Lei *et al.* [12]	25.98	0.967	0.172
Liu *et al.* [29]	26.34	0.962	0.175
Iizuka *et al.* [30]	27.03	0.964	0.057
Zhang *et al.* [10]	28.64	0.972	**0.041**
Our Vitexco	**28.88**	**0.979**	0.042

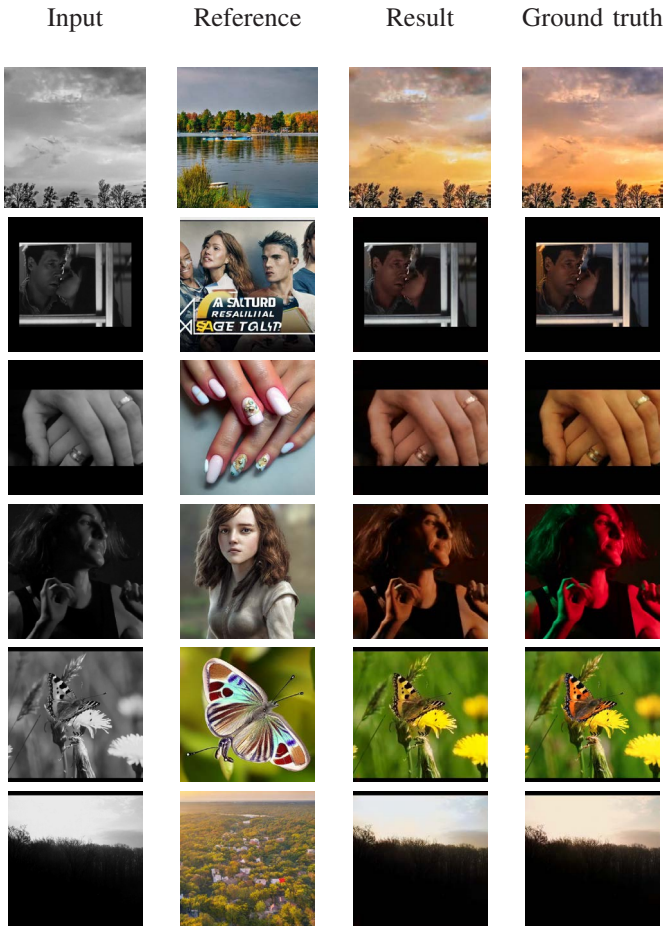

Fig. 4. Video frame predictions

Results: In our research, we aim to evaluate the performance of the proposed model on the DAVIS dataset. The evaluation results in Table I demonstrate our model's promising performance. Although it does not outperform all state-of-the-art models in every metric, our model still belongs to the group of highest-performing models. The visible results of our model are displayed in Figures 1 and 4. Our model produces special colorization, resulting in realistic images and good saturation.

VI. Conclusion

This study presents the first transformer-based video colorization algorithm that surpasses previous models based on CNN. Our approach achieves temporal consistency and

generates realistic effects in video colorization. However, the metrics used in this research are not natively designed for video generating evaluation and human aesthetic assessments. In future work, we intend to enhance the architecture of our model by employing attention methods, particularly in the correspondence and colorization subnets. We may also use Stable Diffusion on the Colorization Subnet for more realistic images and videos. Besides that, we may apply the Swin Transformer [16] instead of the Vanilla ViT [2] architecture as the encoder for taking advantage of the shifted windows mechanism.

REFERENCES

[1] Y. LeCun, Y. Bengio *et al.*, "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, p. 1995, 1995.

[2] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," 2021.

[3] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 649–666.

[4] P. Vitoria, L. Raad, and C. Ballester, "Chromagan: Adversarial picture colorization with semantic class distribution," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[5] X. Chen, D. Zou, Q. Zhao, and P. Tan, "Manifold preserving edit propagation," *ACM Trans. Graph.*, vol. 31, no. 6, nov 2012. [Online]. Available: https://doi.org/10.1145/2366145.2366151

[6] A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization," in *ACM SIGGRAPH 2004 Papers*, ser. SIGGRAPH '04. New York, NY, USA: Association for Computing Machinery, 2004, p. 689–694. [Online]. Available: https://doi.org/10.1145/1186562.1015780

[7] Q. Luan, F. Wen, D. Cohen-Or, L. Liang, Y.-Q. Xu, and H.-Y. Shum, "Natural image colorization," in *Proceedings of the 18th Eurographics conference on Rendering Techniques*, 2007, pp. 309–320.

[8] R. Zhang, J.-Y. Zhu, P. Isola, X. Geng, A. S. Lin, T. Yu, and A. A. Efros, "Real-time user-guided image colorization with learned deep priors," 2017.

[9] J. Yun, S. Lee, M. Park, and J. Choo, "icolorit: Towards propagating local hints to the right region in interactive colorization by leveraging vision transformer," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, January 2023, pp. 1787–1796.

[10] B. Zhang, M. He, J. Liao, P. V. Sander, L. Yuan, A. Bermak, and D. Chen, "Deep exemplar-based video colorization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[11] Y. Yang, Z. Peng, X. Du, Z. Tao, J. Tang, and J. Pan, "Bistnet: Semantic image prior guided bidirectional temporal feature fusion for deep exemplar-based video colorization," 2022.

[12] C. Lei and Q. Chen, "Fully automatic video colorization with self-regularization and diversity," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

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

[14] Y. Wu, X. Wang, Y. Li, H. Zhang, X. Zhao, and Y. Shan, "Towards vivid and diverse image colorization with generative color prior," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 14 377–14 386.

[15] Z. Wan, B. Zhang, D. Chen, and J. Liao, "Bringing old films back to life," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 17 694–17 703.

[16] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 10 012–10 022.

[17] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[18] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," 2015.

[19] R. Mechrez, I. Talmi, and L. Zelnik-Manor, *The Contextual Loss for Image Transformation with Non-Aligned Data*, September 2018.

[20] D. Chen, J. Liao, L. Yuan, N. Yu, and G. Hua, "Coherent online video style transfer," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[21] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid, "Deepflow: Large displacement optical flow with deep matching," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2013.

[22] M. Ruder, A. Dosovitskiy, and T. Brox, "Artistic style transfer for videos," pp. 26–36, 2016.

[23] M. Marszalek, I. Laptev, and C. Schmid, "Actions in context," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 2929–2936.

[24] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung, "A benchmark dataset and evaluation methodology for video object segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[25] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[26] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[27] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Let there be color! joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification," *ACM Trans. Graph.*, vol. 35, no. 4, jul 2016. [Online]. Available: https://doi.org/10.1145/2897824.2925974

[28] J.-W. Su, H.-K. Chu, and J.-B. Huang, "Instance-aware image colorization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[29] Y. Liu, H. Zhao, K. C. K. Chan, X. Wang, C. C. Loy, Y. Qiao, and C. Dong, "Temporally consistent video colorization with deep feature propagation and self-regularization learning," 2021.

[30] S. Iizuka and E. Simo-Serra, "Deepremaster: Temporal source-reference attention networks for comprehensive video enhancement," *ACM Trans. Graph.*, vol. 38, no. 6, nov 2019. [Online]. Available: https://doi.org/10.1145/3355089.3356570