Peng ZZ, Yang YX, Tang JH *et al.* Video colorization: A survey. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 39(3): 487-508 May 2024. DOI: 10.1007/s11390-024-4143-z

Video Colorization: A Survey

Zhong-Zheng Peng[†] (彭中正), Yi-Xin Yang[†] (杨艺新)
Jin-Hui Tang (唐金辉), *Distinguished Member, CCF, Senior Member, IEEE, Member, ACM*
and Jin-Shan Pan[*] (潘金山), *Senior Member, CCF, IEEE*

School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China

E-mail: pengzz@njust.edu.cn; yangyixin@njust.edu.cn; jinhuitang@njust.edu.cn; jspan@njust.edu.cn

Received January 23, 2024; accepted April 24, 2024.

Abstract Video colorization aims to add color to grayscale or monochrome videos. Although existing methods have achieved substantial and noteworthy results in the field of image colorization, video colorization presents more formidable obstacles due to the additional necessity for temporal consistency. Moreover, there is rarely a systematic review of video colorization methods. In this paper, we aim to review existing state-of-the-art video colorization methods. In addition, maintaining spatial-temporal consistency is pivotal to the process of video colorization. To gain deeper insight into the evolution of existing methods in terms of spatial-temporal consistency, we further review video colorization methods from a novel perspective. Video colorization methods can be categorized into four main categories: optical-flow based methods, scribble-based methods, exemplar-based methods, and fully automatic methods. However, optical-flow based methods rely heavily on accurate optical-flow estimation, scribble-based methods require extensive user interaction and modifications, exemplar-based methods face challenges in obtaining suitable reference images, and fully automatic methods often struggle to meet specific colorization requirements. We also discuss the existing challenges and highlight several future research opportunities worth exploring.

Keywords video colorization, deep convolutional neural network, spatial-temporal consistency

1 Introduction

Video colorization aims at adding color to black and white (monochrome) videos, making them more vivid and visually appealing. Due to technological constraints, a large amount of existing videos, which possess high historical value and carry profound human emotions, remain in black and white. Moreover, the replication of these videos is infeasible due to the considerable lapse of time. Therefore, the necessity for colorizing these videos is increasingly significant. However, video colorization is highly ill-posed and usually struggles with spatial-temporal inconsistencies, i.e., variations in the quality of individual frames and noticeable fluctuations between consecutive frames. Although great progress has been made, restoring high-quality colorized videos remains a challenging problem.

Video colorization methods can generally be categorized into four main categories: optical-flow based methods, scribble-based methods, exemplar-based methods, and fully automatic methods. Compared with image colorization methods, the simplest way for video colorization is initially applying an image colorization technique, followed by post-processing enhancement to promote temporal consistency in videos[1–3]. These methods utilize optical flow to propagate information between frames, thereby yielding smooth results. However, optical-flow based methods rely on the performance of employed image colorization methods and the accuracy of estimated optical flow, consequently constraining the performance of

Survey

This work was supported by the National Natural Science Foundation of China under Grant Nos. U22B2049 and 62332010.

[†]Co-First Author (Zhong-Zheng Peng wrote the section about the classification of video colorization methods, and Yi-Xin Yang wrote the introduction. Zhong-Zheng Peng and Yi-Xin Yang jointly participated in the remaining sections.)

[*]Corresponding Author

these methods. To achieve better each-frame performance, several methods incorporate the propagation of color information from a color reference frame or sparse user scribbles throughout the entire video[4–21]. However, it is not easy to obtain a qualified reference image which requires extensive user interaction and revisions.

To avoid non-trivial human effort involved in exemplar-based or scribble-based methods, fully automatic video colorization methods[22–34] have increasingly gained popularity. These methods predominantly rely on the potent expressive power and learning capacity of deep convolutional neural networks. They enrich color representation and offer an end-to-end optimization approach. These methods can be trained on large-scale datasets to learn complex color mapping relationships and can automate and refine the colorization process without user interference. Furthermore, the flexibility and applicability furnished by deep learning based frameworks contribute to their capability to handle a vast array of video content and circumstances, thereby substantially enhancing the quality of colorization.

Deep learning based methods greatly improve the performance of video colorization, owing significantly to their powerful representational capacity. In these methods, how to enhance spatial-temporal consistency is of great significance. In terms of spatial information, existing methods[1, 12, 16, 18, 19, 24, 26, 30, 33, 35] usually extract features by pre-trained models, such as VGG[36] or ResNets[37]. Video colorization methods that consider temporal consistency generally fall into four categories: optical-flow based methods, recurrent neural networks (RNNs) based methods, 3D convolution based methods, and bi-directional based methods. Optical-flow based methods aim to estimate the similarities between consecutive or far-away frames by utilizing calculated optical flow thus enforcing color consistency for frames within a video. These methods typically pay more attention to how to improve the accuracy of estimated optical flow and alleviate the influence of inaccurate flow, e.g., utilize a confidence mask to lower the weights of uncertain flow[1–3]. Different from optical-flow based methods that concentrate on detecting motion through pixel variations, RNN-based methods are designed for general sequence learning, which can include various types of temporal pattern recognition beyond motion. They can handle variable-length sequence input by maintaining a hidden state that effectively captures information from previously seen elements in the sequence[12, 18, 19, 23, 24]. Optical flow can be part of the input features for an RNN, enabling the network to leverage detailed motion information for its task. 3D convolution based methods apply 3D convolutions to a stack of consecutive video frames to capture temporal consistency[15, 38]. Bi-directional based methods add reference images at the beginning and end of the video, followed by propagating the color information from these reference images to the intermediate frames. These methods strive to amplify the spatial-temporal consistency in generated videos. However, they struggle to achieve an optimal balance between model complexity and performance. Future work still considers strengthening the performance of video colorization methods from a spatial-temporal perspective.

In this paper, we focus on recently published video colorization methods. The aims of this paper are:

- to review the preliminaries for video colorization, including problem definitions, choice of color spaces, benchmark datasets for performance evaluation, and video quality assessment;
- to discuss developments of video colorization methods and provide a taxonomy for categorizing the existing methods;
- to review video colorization methods from a new perspective in terms of spatial-temporal consistency;
- to analyze the challenges of video colorization and discuss research opportunities.

The rest of this paper is organized as follows. Section 2 introduces problem setting and terminology for video colorization. Section 3 summarizes various metrics for evaluating the quality of video colorization. Section 4 lists commonly used datasets for video colorization. In Section 5, we introduce state-of-the-art video colorization methods, which are further grouped into four subcategories. In Section 6, we summarize several strategies for maintaining spatial consistency in videos. In Section 7, we evaluate the performances of various video colorization methods. Section 8 discusses the loss functions used in video colorization methods. Section 9 and Section 10 outline the main challenges of video colorization and suggest possible next steps in the field. In Section 11, we provide a summary of this video colorization review.

2 Problem Setting and Terminology

2.1 Problem Definitions

Video colorization is a computer vision task that aims to generate fully colorized videos from their gray-

scale (monochrome) versions. Fig.1 shows a timeline of representative methods for video colorization. Given the input grayscale video X, we first split it to N grayscale frames as $X = \{\boldsymbol{x}_t\}_{t=1}^{N}$, where $\boldsymbol{x}_t \in \mathbb{R}^{H\times W\times 1}$, $H \times W$ denotes the spatial resolution, N is the number of frames of the input video. Our target is to restore the colorized video $Z = \{\boldsymbol{z}_t\}_{t=1}^{N}$, where $\boldsymbol{z}_t \in \mathbb{R}^{H\times W\times 3}$. To better restore the color videos based on the input grayscale frames, the selection of color space is critical in colorization tasks. It impacts color representation, deep learning model performance, alignment with human color perception, and handling of visual artifacts. The RGB, YUV, and CIELab color spaces are the most commonly used standards in colorization. Fig.2(a) shows that there is a predominant utilization of both the RGB and CIELab color spaces in comparison with the adoption of the YUV color space.

2.1.1 *RGB Color Space*

The RGB color space is an additive color model leveraging the primary colors: red (R), green (G), and blue (B), to which human vision is particularly sensitive. It is a crucial model in computer vision, graphics, and digital media. A way to model the luminance Y_1 (absolute amount of light emitted by an object per unit area), which is close to human perception is:

$$Y_1 = 0.298\,9R + 0.587\,0G + 0.114\,0B.$$

Several video colorization methods[2, 3, 6, 24, 26, 30–32] use RGB color space. They predict three channels R, G, and B in the RGB color space. While the RGB color space highlights the intensity of primary colors to which human eyes are sensitive, exploring the distinct information contained within each individual channel remains challenging. Additionally, the interpretability of information in each channel of the RGB color space is limited, thereby impeding the design of more effective neural network models.

2.1.2 *YUV Color Space*

Besides the RGB color space, the YUV color space is also widely used for video colorization. The YUV color space is used primarily in video systems

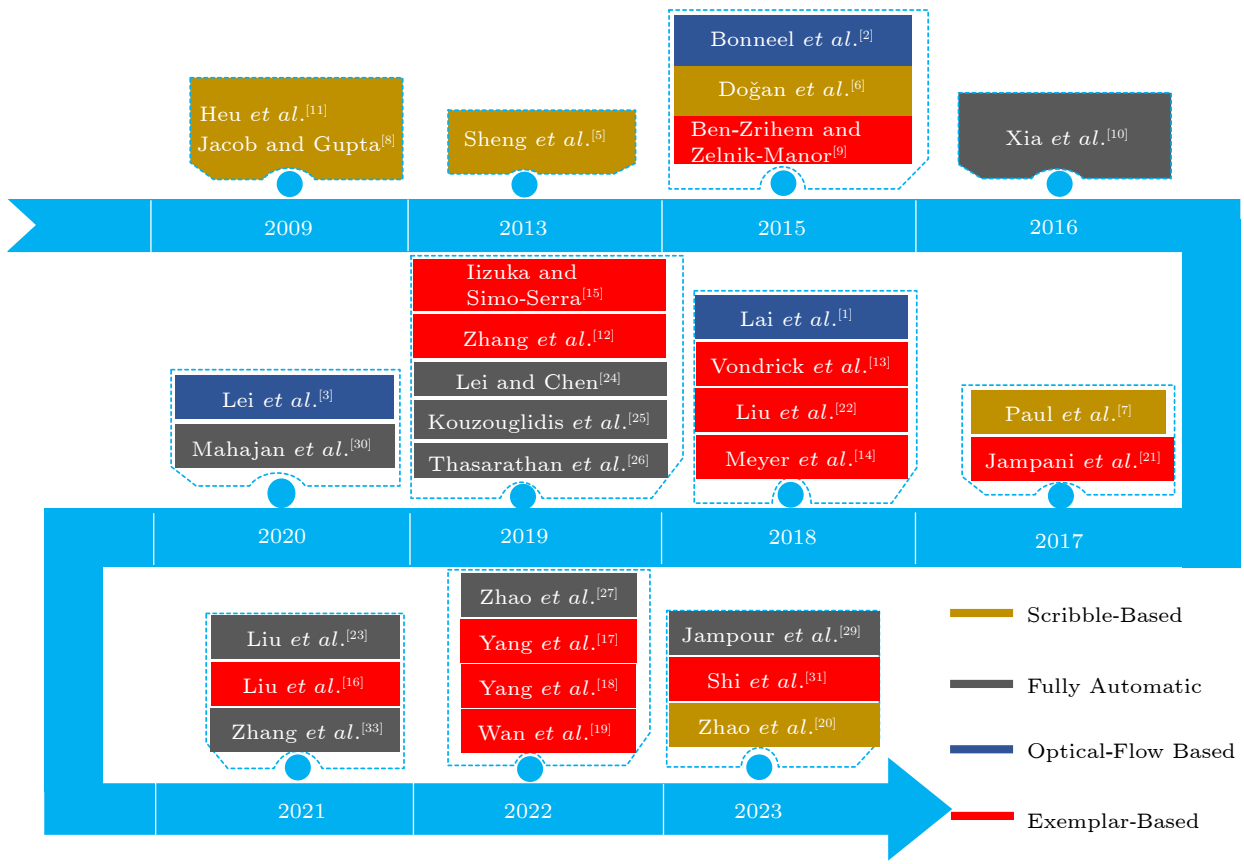

Fig.1. Timeline of video colorization methods. Different colors represent methods of different categories, as shown in the lower right. The initial emergence is the scribble-based methods, which are subsequently followed by the popularity of optical-flow based methods and exemplar-based methods. After that, fully automatic methods began to appear.

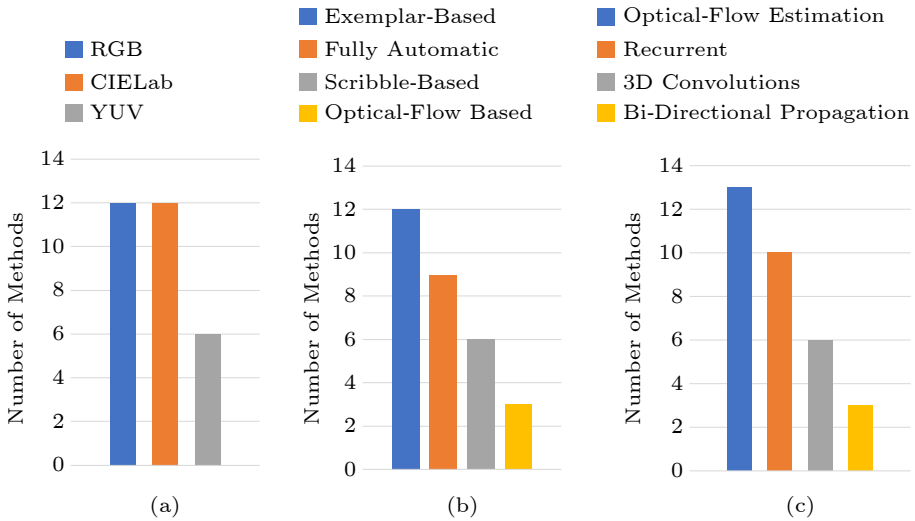

Fig.2. Statistical analysis of video methods, including (a) color space, (b) category, and (c) temporal enhancement.

and broadcasting. It is known for its separation of luminance Y_2 from chrominance (color information, represented by U and V). The term YUV signifies these three components. Note that Y_2 in YUV represents luminance exclusively based on human perception, whereas the luminance Y_1 in RGB is a composite function of the three color channels and is not specifically tailored to human perception. Unlike RGB, which directly denotes the intensities of Red, Green, and Blue, YUV discloses color information in a way that is more perceptually relevant and closer to how humans perceive color. Y_2 component aligns with the overall brightness seen by human eyes, while U and V summarize the color difference from gray at the same luminance level.

Compared with RGB, YUV is resilient to changes in lighting conditions and beneficial for data compression. Deep learning based video colorization methods[4, 11] use the YUV color space and achieve favorable performance. However, when employing this color space for video colorization tasks, artifacts remain unpredictable.

2.1.3 *CIELab Color Space*

The CIELab color space is widely used in [1, 5, 7–10, 12–19, 21–23, 25, 27–29, 31]. It is a color-opponent space with dimensions L for lightness and a^* and b^* for the green-red and blue-yellow color components respectively. The CIELab color space was developed by the International Commission on Illumination[①] (CIE) to create a space that is more perceptually uniform than its counterparts, meaning that a given numerical change corresponds approximately to the same perceived change in color.

Unlike the RGB color space, which provides an additive color model based on how much red, green, and blue light is emitted, the CIELab color space describes how a color appears to the human eye. It does not rely on a specific device (like a monitor or printer) for interpretation and is, therefore, considered device-independent. This is a marked contrast from the RGB model which can have substantial variation across different devices due to their distinct color-enhancing methodologies.

Similarly, while the YUV color space separates luminance (brightness) and chrominance (color information) which is particularly useful for color television broadcasting, it is not designed to align with the human perception of color. Hence CIELab, with its intention to mimic the human perceptual experience of color, offers a better understanding of color, particularly beneficial for precise color manipulations and color difference calculations.

3 Video Quality Assessment Metrics

For the purpose of assessing the quality of video colorization in regard to subjective assessment, objective assessment, temporal consistency evaluation, color diversity evaluation, color diversity evaluation, and semantic interpretability evaluation, existing video colorization methods employ a variety of evaluation metrics. Here are several key measures.

[①] http://cie.co.at, May 2024.

3.1 Subjective Assessment

This typically involves human observers rating the quality of colorized videos. It is considered the most accurate way to measure colorization quality since human perception is the ultimate arbiter. However, it is time-consuming.

3.2 Objective Assessment

This uses mathematical models and neural network models to evaluate colorization quality against a reference colorized version of the grayscale video.

Image Quality Evaluations. The commonly used metrics include: peak signal-to-noise ratio (PSNR), mean squared error (MSE), root mean squared error (RMSE), structural similarity index measurement (SSIM)[39], Fréchet inception distance (FID)[40], learned perceptual image patch similarity (LPIPS)[41], raw accuracy (RA)[25], and color consistency (CC)[25].

Temporal Consistency Evaluations. Temporal consistency in the context of video colorization refers to the stability of colorization results across consecutive frames. In a colorized video, the colors of the same object or scene must remain consistent throughout the video sequence to avoid flickering or sudden color changes that can disrupt the experience of viewers. For temporal consistency, the warp error (E_{warp}) proposed in [1] is widely used in video colorization.

$$E_{\text{warp}}(V_t, V_{t+1}) = \frac{1}{\sum_{i=1}^{n} M_t^{(i)}} \sum_{i=1}^{n} M_t^{(i)} ||V_t^{(i)} - \hat{V}_{t+1}^{(i)}||_2,$$

$$E_{\text{warp}} = \frac{1}{T-1} \sum_{t=1}^{T-1} E_{\text{warp}}(V_t, V_{t+1}),$$

where $\hat{V}_{t+1}$ represents the warped frame of V_{t+1}; M_t denotes the non-occluded mask for the non-occluded regions, with values of 0 or 1; t represents the time step, and n represents the total number of pixels in a frame.

However, E_{warp} does not correlate with video color and can be significantly influenced by the performance of the flow estimation models employed in its measurement. Consequently, Liu *et al.*[23] proposed the color distribution consistency (CDC) to measure temporal consistency, which is specially devised for video colorization tasks. As E_{warp} is unrelated to the colors of videos and is susceptible to the performance of the flow estimation module, CDC which is specifically designed for video colorization tasks is employed to measure the temporal consistency of color distributions. Specifically, CDC is a metric that can estimate the Jensen-Shannon (JS) divergence of color distributions between consecutive frames:

$$CDC = \frac{1}{3 \times (N-1)} \sum_{c \in \{R,G,B\}} \sum_{i=1}^{N-t} JS(\mathcal{P}_c(I^i), \mathcal{P}_c(I^{i+t})),$$

where N represents the length of the video sequence, t represents the time step, and $\mathcal{P}_c(I^i)$ represents the normalized probability distribution of color image i on channel c, which can be computed using the image histogram.

Color Diversity Evaluation. To evaluate the vividness of generated videos, the colorfulness score (CF)[42] is employed in existing video colorization methods to measure the color diversity. The CF can be written as:

$$CF = \sigma_{\text{rgyb}}(\boldsymbol{z}_t) + 0.3 \times \mu_{\text{rgyb}}(\boldsymbol{z}_t),$$

where $\sigma_{\text{rgyb}}(\cdot)$ and $\mu_{\text{rgyb}}(\cdot)$ represent the standard deviation and the mean value of the pixel cloud in the color plane, respectively, as explained in [42], and $\boldsymbol{z}_t$ is the output frame at time t.

Semantic Interpretability Evaluation. To determine semantic interpretability, the measure of top-1 and top-5 accuracy is employed in existing video colorization methods based on a pre-trained VGG-16[36].

4 Datasets for Video Colorization

For video colorization, colorization models need to be trained and tested on high-quality datasets. These datasets should contain a large number of video sequences with high resolution, good brightness and contrast, consistent colors, and a wide range of scenes. Multiple datasets are available for video colorization, and each comes with its unique characteristics. The datasets listed below fully meet the requirements for video colorization and have been adopted by many state-of-the-art methods[12, 13, 15, 18–20, 22–24].

Kinetics 600②. It is a large-scale, high-quality dataset for human action recognition introduced by Google DeepMind③. Despite the primary use case, its

②https://github.com/cvdfoundation/kinetics-dataset, May 2024.
③https://deepmind.google, May 2024.

diverse and extensive collection of YouTube video URLs can be used for video colorization tasks, provided the videos are converted to grayscale before colorization. It comprises a total of 480 000 video clips, divided into 600 categories. Each category in the training dataset, the validation dataset, and the test dataset includes about 390 000, 50 000, and 60 000 video clips, respectively.

Vimeo-90*K*[④]. This dataset has a large variety of videos that makes it suitable for the colorization task[43]. It contains 90 000 video clips from 39 000 different videos from Vimeo[⑤], a website where users can upload, share, and view videos. The Vimeo-90K uses 64 612 clips for training and 7 824 clips for testing.

ImageNet VID[⑥]. ImageNet VID is a large-scale public dataset for video object detection and contains more than 1 million frames for training and more than 100 thousand frames for validation[44].

*UCF*101[⑦]. It is a dataset of realistic action videos collected from YouTube, and it is highly versatile because it comprises 101 action categories[45]. It consists of 13 320 video clips, totaling 27 hours. Following the most popular setting used in the UCF101 dataset, [46, 47] adopt three training/testing splits for evaluation.

DAVIS[⑧]. The DAVIS (Densely Annotated Video Segmentation) dataset consists of high-quality video sequences and provides pixel-level annotations, mainly for video object segmentation tasks[48–50]. Nonetheless, its varied contents and detailed segmentation can further facilitate video colorization research. During training and testing, it employs 60 and 30 video clips, respectively.

Videvo[⑨]. Videvo is an online platform that offers free stock videos and motion graphics. Lai *et al.*[1] have gathered 100 high-quality videos from Videvo.net[⑩] to constitute the Videvo dataset. Its vast and diverse content can be utilized for training and testing on video processing tasks, such as video colorization. The Videvo dataset comprises 80 videos for training and 20 videos for testing.

YouTube-8*M*[⑩]. This is a massive multi-genre dataset containing links to YouTube videos[51]. It primarily focuses on the task of video understanding. However, its large scale and variety make it suitable for adapting to video colorization tasks. The YouTube-8M dataset consists of 8 264 650 video clips, which are divided into the training, validation, and test set in a ratio of 70% : 20% : 10%.

ACT[⑪]. ACT (Actor-Action) is a large-scale video dataset meticulously annotated for different facets, including human actions, actors, and interactions. Made up of 383 hours of soap opera videos, it contains over 75 000 unique clips of 430 actors performing over 15 000 categories of actions. For video colorization, the diverse and rich content in ACT provides a challenging environment that tests the robustness and adaptability of colorization algorithms to different scenes and actor motions. It comprises a total of 3 782 video clips, with 3 036 videos used for training and 746 videos used for testing.

MS-COCO[⑫]. MS-COCO (Microsoft Common Objects in Context) is a widely used dataset for object detection, segmentation, and captioning studies. Although it consists of static images, it can still be beneficial for video colorization. MS-COCO contains a vast selection of images with complex scenes, providing a rich variety of color textures that learning algorithms can leverage while training to colorize videos. In the 2014 version, it contains 164 062 images, divided into the training set (82 783), the validation set (40 504), and the test set (40 775). In the 2015 version, there are 165 482 images used for training, 81 208 images used for validation and 81 434 images used for testing. In the 2017 version, the training, validation, and test sets consist of 118 287, 5 000, and 40 670 images, respectively.

*Hollywood*2[⑬]. The Hollywood2 dataset is a popular dataset used in video classification research but can also contribute to video colorization tasks. In the context of video colorization, its diversity provides a challenging training environment due to the high de-

④https://github.com/anchen1011/toflow, May 2024.
⑤https://github.com/vimeo, May 2024.
⑥https://image-net.org/challenges/LSVRC, May 2024.
⑦https://www.crcv.ucf.edu/data/UCF101.php, May 2024.
⑧https://davischallenge.org, May 2024.
⑨https://github.com/phoenix104104/fast_blind_video_consistency, May 2024.
⑩https://www.videvo.net, May 2024.
⑪https://research.google.com/youtube8m, May 2024.
⑫https://www.cs.cmu.edu/xiaolonw/actioncvpr.html, May 2024.
⑬https://www.di.ens.fr/laptev/actions/hollywood2, May 2024.

gree of scene variety, actor appearance variation, and dynamic lighting changes. The Hollywood2 dataset is collected from 69 films, 33 of which are used for training and 36 for testing.

5 Classification of Video Colorization Methods

We classify video colorization methods into four categories: optical-flow based methods, scribble-based methods, exemplar-based methods, and fully automatic methods. In the following subsections, we discuss representative methods of each category (see Table 1). Fig.2(b) shows that exemplar-based methods account for the majority.

5.1 Optical-Flow Based Methods

A critical challenge in video colorization is the spatial-temporal consistency of video frames. The spatial-temporal consistency effectively assesses the quality of a video colorization algorithm. Current methods based on optical flow enhance spatial-temporal consistency by accurately aligning features across consecutive frames after effectively calculating location correspondences between neighboring frames.

The simplest video colorization methods are to directly apply image colorization techniques on each frame. However, these techniques often lead to flickering issues. To address this issue, Lei *et al.*[3] introduced a novel algorithm named Deep Video Prior (DVP) to effectively propagate color information over the video frames without relying on the computation of similarity between adjacent pixels. The algorithm is based on the fact that similar inputs will yield similar output results from convolutional neural networks (CNNs) and the same object in different video frames has similar appearances. In DVP, fully convolutional networks are employed to simulate the original image processing algorithm and maintain temporal consistency in the video. The fully convolutional network can be adjusted accordingly to adopt U-Net or other suitable CNN architectures (e.g., FCN[52]), based on different tasks. Additionally, the method utilizes an iterative weighted training strategy to address the issue of multimodal inconsistency.

Similarly, Bonneel *et al.*[2] proposed a post-processing method aiming to enhance the flickering colorized frames obtained by applying the image colorization method to each frame independently and generating temporally consistent video sequences. The core of the method is exploring the temporal regularity from the original grayscale video, and using it as a temporal consistency guidance to stabilize the processed sequence. Notably, the method uses the frequency domain to propagate color. And the predicted frames are compared with the original unprocessed video frames by minimizing the least-squares energy. Experiments show that this method improves temporal smoothing and is able to produce high-quality results on a wide variety of applications independently of their inner workings.

Due to the robust performance of the optical-flow techniques, some methods whose primary goal is not colorization can still generate satisfactory colors. To reduce the significant human labor costs of animation video production, the work by Siyao *et al.*[53] primarily focuses on interpolating animation frames and predicting the colors of the animated frames. This method combines segmentation techniques, the recurrent flow refinement (RFR) network, and feature extraction using the pre-trained VGG-19 network[36]. The RFR network in this method draws inspiration from the architecture of the Transformer model to achieve recurrent refinement of optical flow. Additionally, Laplacian filters are employed to extract the edge contours of video frames, and the trapdball algorithm[54] is utilized to fill these contours and generate color patches. By using this combined technique, the researchers can obtain intermediate animation frames that have vivid color and clear details, providing an enhanced visual sense.

Applying optical flow can achieve satisfactory colorization performance in some simple scenes. However, when faced with scenarios involving large-scale motion of objects, the issue of color bleeding remains.

5.2 Scribble-Based Methods

Due to the performance limitations of optical-flow based video colorization methods, which are dependent on the performance of image colorization and the accuracy of flow estimation, researchers have adopted an approach combined with scribbles. Scribble-based video colorization methods first introduce color points into video frames and then propagate the colors of these points to the corresponding target objects.

Scribble-based video colorization methods are the earliest techniques employed in the field of coloriza-

Table 1. Summarization of Representative Video Colorization Methods

Year	Method	Category	Space	Backbone	Feature Propagation	Loss Function	Experimental Dataset	Evaluation Metric	Venue
2009	Heu *et al.*[11]	S	YUV	–	–	–	"Funny Face" movie	Visual comparison	ICIP
2009	Jacob and Gupta[8]	S	RGB	–	–	–	"City Lights" movie	Visual comparison	ICIP
2013	Sheng *et al.*[5]	S	YUV	–	–	–	Several videos	Visual comparison	TCSVT
2015	Bonneel *et al.*[2]	O	RGB	–	–	–	Several videos	Visual comparison	TOG
2015	Doğan *et al.*[6]	S	RGB	–	–	–	Several videos	Visual comparison	WICED
2015	Ben-Zrihem and Zelnik-Manor[9]	E	RGB	–	–	–	Several videos	Visual comparison	CVPR
2016	Xia *et al.*[10]	F	RGB	–	–	–	Several videos	PSNR	ICIP
2017	Paul *et al.*[7]	S	YUV	–	–	–	Several videos	PSNR	TCSVT
2017	Jampani *et al.*[21]	E	YUV	–	Bilateral network	L_2	DAVIS	PSNR	CVPR
2018	Lai *et al.*[1]	O	RGB	–	Recurrent	CP, ST, LT	DAVIS, Videvo	E_{warp}, $D_{\text{perceptual}}$	ECCV
2018	Vondrick *et al.*[13]	E	CIELab	ResNet-18[37]	3D convolutions	CE	Kinetics	Visual comparison	ECCV
2018	Liu *et al.*[22]	E	CIELab	CNN	Bi-directional propagation	L_2	ACT, MS-COCO	PSNR, RMSE	ECCV
2018	Meyer *et al.*[14]	E	YUV	CNN	Local and global propagation	L_1, E_{warp}	DAVIS	PSNR	BMVC
2019	Iizuka and Simo-Serra[15]	E	CIELab	CNN	3D convolutions	L_1	YouTube-8M	PSNR	ACM TOG
2019	Zhang *et al.*[12]	E	CIELab	VGG-19[36]	Recurrent	L_1, PL, CT, SL, Adv, TC	DAVIS, Videvo, Hollywood2	PSNR, Top-1, Top-5, FID, CF	CVPR
2019	Lei and Chen[24]	F	RGB	VGG-19[36]	Recurrent	SR, DL, TC	DAVIS, Videvo	PSNR, LPIPS	CVPR
2019	Kouzouglidis *et al.*[25]	F	CIELab	CNN	3D convolutions	L_1, Adv	Real films	PSNR, RA, CC	ISVC
2019	Thasarathan *et al.*[26]	F	RGB	VGG-19[36]	Recurrent	TC, CL, Adv	Anime, Dragonball	PSNR, SSIM, FID	CRV
2020	Lei *et al.*[3]	O	CIELab	CNN	Deep video prior	L_1	DAVIS	E_{warp}, F_{data}	NIPS
2020	Mahajan *et al.*[30]	F	CIELab	VGG-19[36]	Recurrent	L_1, PL	DAVIS	PSNR, MSE	MIDAS
2021	Liu *et al.*[23]	F	CIELab	CNN	Bi-directional propagation	TC	DAVIS, Videvo	PSNR, CDC, CF, E_{warp}	arXiv
2021	Liu *et al.*[16]	E	CIELab	VGG-16[36]	Recurrent	PL, Adv, TC	DAVIS, Videvo	PSNR, LPIPS, FID, E_{warp}	ICIP
2021	Zhang *et al.*[33]	F	RGB	ResNet[37]	Recurrent	L_1, PL	Cartoon, films	PSNR, SSIM	WACV
2022	Zhao *et al.*[27]	F	RGB	ResNet[37]	Recurrent	L_1, PL, Adv, ST, LT	DAVIS, Videvo	PSNR, SSIM, E_{warp}	ACM TMM
2022	Yang *et al.*[17]	E	YUV	CNN	3D convolutions	L_1	DAVIS, Videvo	PSNR, SSIM	ICIGP
2022	Yang *et al.*[18]	E	CIELab	VGG-19[36]	Bi-directional propagation	L_1, PL, CT, SL, Adv, TC, HL, EL	DAVIS, Videvo	PSNR, SSIM, LPIPS, CDC, FID, CF	arXiv
2022	Wan *et al.*[19]	E	CIELab	VGG-19[36]	Recurrent	L_1, PL, Adv	DAVIS, Videvo	PSNR, SSIM, LPIPS, E_{warp}	CVPR
2023	Jampour *et al.*[29]	F	RGB	GAN	3D convolutions	Adv	Several, Videos	PSNR, SSIM, RMSE, MAE, FID	JAIHC

(to be continued)

Table 1. Summarization of Representative Video Colorization Methods (Continued)

Year	Method	Category	Space	Backbone	Feature Propagation	Loss Function	Experimental Dataset	Evaluation Metric	Venue
2023	Shi *et al.*	E	RGB	VGG-19[36]	3D convolutions	L_1, PL, Sty, Adv	Animation	PSNR, SSIM, MSE, FID	TVCG
2023	Zhao *et al.*[20]	S	CIELab	VGG-16[36]	Recurrent	L_1	DAVIS, Videvo	PSNR, SSIM	TIP

Note: CP, content perceptual loss[1]. ST, short-term temporal loss[1]. LT, long-term temporal loss[1]. CE, cross entropy loss. $D_{\text{perceptual}}$, perceptual distance. CT, contextual loss[12]. SL, smoothness loss[12]. Adv, adversarial loss. TC, temporal consistency loss[12]. PL, perceptual loss[12]. SR, self-regularization loss[24]. DL, diversity loss[24]. HL, hard example mining loss[18]. EL, edge-enhancing loss[18]. Sty, style loss[31]. E_{warp}, temporal warping error[2]. F_{data}, data fidelity[3]. S, scribble-based method. O, optical-flow based method. E, exemplar-based method. F, fully automated method. Top-1 and Top-5 denote the best top-1 and top-5 class accuracy, respectively. "–" means this item is not available or not indicated in its paper.

tion. Early methods based on scribbles utilize traditional methods to achieve color propagation without deep learning techniques such as the work by Levin *et al.*[55] and the work by Yatziv and Sapiro[4]. The former relies on an assumption that neighboring pixels in space-time that have similar intensity may exhibit similar colors. And it employs a quadratic cost function to formulate an optimization problem which can be solved by standard approaches. The latter is based on the techniques of luminance-weighted chrominance blending and fast intrinsic distance computations. With fewer chrominance scribbles, the method can rapidly achieve high-quality colorization, significantly reducing both complexity and computational costs compared with previous techniques.

Given that the aforementioned methods rely on traditional techniques rather than deep learning, their spatial-temporal coherence in video colorization is not satisfactory. Therefore, Heu *et al.*[11] proposed a method to propagate the colors through the scribbles from the initial frame in the video or an example image in deep learning techniques. They employed a block-matching technique to estimate the differences between the last colorized frame and the current frame, effectively preserving the spatial-temporal coherence of the video. However, this algorithm exhibits several shortcomings. For instance, it requires that the video frames and the scribble image have similar brightness, potentially limiting its applicability. Moreover, when dealing with long videos, occluded objects may cause noticeable color bleeding, thus affecting the overall quality of video colorization. The algorithm follows the following execution flow. Initially, users are required to paint corresponding colors on the first frame of the grayscale video sequences. Subsequently, it utilizes motion compensation prediction to colorize the current frame using colors from the previous frame. Then, it interpolates colors according to adjacent pixels. This process iterates continuously until all video frames are colorized.

To overcome the color bleeding issue in the aforementioned method during the color propagation process, Sheng *et al.*[5] proposed a method to maintain temporal coherence by using optimization in the rotation-aware Gabor feature space. The method clusters video frames and applies the Gabor filter to optical flow computation to achieve real-time color propagation within and between frames. Temporal coherence is further enhanced through scribbles provided by users in video frames. The main procedure of this method is as follows: 1) establish rotation-aware Gabor filters to identify texture features of the images, 2) divide the feature space generated by Gabor filters into K-D tree subgraphs adaptively, 3) represent the correspondence between different subgraphs through the constructed Gabor flow, and 4) propagate the corresponding colors to the pixels of these subgraphs in parallel.

Most previous scribble-based video colorization methods are highly labor-intensive and suffer from inaccurate scribble propagation. Doğan *et al.*[6] applied semi-automatic permeability-guided filtering techniques to expand the colors from the scribbles over entire input frames. They consider the local features of object boundaries to avoid color bleeding and the utilization of global entropy helps maintain overall image spatial consistency. In this approach, users are initially required to manually provide scribbles for some keyframes of the input video. Subsequently, an automatic propagation method is applied to process the scribbles and the input video, generating spatiotemporally propagated scribble colors.

In the process of video colorization, the issue of object occlusion often leads to color bleeding. Therefore, Paul *et al.*[7] proposed a technique based on spatiotemporal color propagation in the 3D volume to address this problem. The approach proposed by Paul *et al.*[7] differs from other methods by employing a steerable pyramid decomposition technique to propa-

gate color without the need for bi-directional propagation. By employing spatial-temporal color propagation in the 3D space instead of using motion vector computation, this approach helps to avoid the propagation of inaccurate color caused by object occlusion in intermediate frames, resulting in more accurate and continuous color generation. In this algorithm[7], the process of color propagation is as follows. Scribbles are first added to the selected keyframes. Then, the spatial-temporal features of the video are extracted by a pyramid composed of filters. Finally, by incorporating spatial-temporal information from surrounding pixels, this method can generate colorized videos while preserving both temporal and spatial consistency.

In recent times, scribble-based video colorization methods have become less attractive due to the requirement of a significant amount of manual intervention and human effort. Moreover, most scribbles provided cannot meet the need for accurate colorization of details and edges of objects.

5.3 Exemplar-Based Methods

Although scribble-based video colorization methods provide users with personalized colorization options, they are time-consuming due to the extensive manual operations. To address this issue, researchers have proposed exemplar-based video colorization methods. Exemplar-based colorization methods[9, 12–19, 21, 22, 31] have been highly praised in the field of video colorization recently, in light of their distinct advantages in colorization efficiency, color coherence, and color accuracy compared with scribble-based methods. Its basic principle is to extract corresponding colors from reference frames and transfer these colors to grayscale video frames.

Considering the high cost of collecting large-scale annotated datasets, Vondrick *et al.*[13] employed a self-supervised approach to train their network, reducing the need for human efforts. They proposed an exemplar-based colorization method, utilizing a visual tracker network to transfer colors from reference frames to the grayscale frames. Specifically, feature representations of the example frames and grayscale frames extracted by convolutional neural networks are mapped in a shared feature space. Subsequently, a similarity matrix is generated by calculating the similarities between these features. In the similarity matrix, each element represents the correspondence of each feature element between the reference frame and the grayscale frame. This algorithm[13] primarily emphasizes pixel-level similarities while paying less attention to temporal consistency. Consequently, flickering artifacts usually occur in the results.

In previous video colorization methods, the lack of sufficient utilization of semantic information from the scenes often leads to color artifacts during color propagation, thereby affecting colorization quality. To address this issue, a method proposed by Meyer *et al.*[14], incorporating the global and local propagation of features from reference frames to avoid spatial-temporal degradation during the frame-by-frame propagation process, has excellent colorization performance and maintains color coherence effectively. Furthermore, this method employs a softmax layer for feature interpolation, thereby expediting the convergence speed during the training process and enhancing the robustness of the model. Besides, this method also incorporates an object color preservation mechanism to better colorize the occluded objects by retaining global information in video frames.

Exemplar-based colorization methods often assume the first frame is colorized and then propagate its color to subsequent frames. However, this frame-by-frame propagation may lead to the accumulation of errors, affecting the colorization quality of subsequent frames. In the method developed by Zhang *et al.*[12], they did not use traditional convolutional neural networks but employed recurrent neural networks (RNNs) to transfer colors from reference frames to grayscale video frames. The main characteristic of this method is the simultaneous utilization of the colors from both the reference frame image and previous frames to jointly guide the colorization of the current frame. As a result, this method is capable of propagating more accurate colors to each frame relieving the the accumulation of color errors during propagation. Although this method achieves good colorization results in some simple video scenes, there are still occurrences of color bleeding when handling complicated scenes.

Due to poor storage conditions, many old movies have suffered severe damage. To restore the brilliance of these classic films, Iizuka and Simo-Serra[15] applied attention mechanisms to video colorization and their proposed algorithm notably demonstrates the capability to restore the visual quality of old films. To ensure good spatial-temporal consistency in videos, this algorithm introduces spatial-temporal convolutional layers and utilizes multiple reference

images to guide the video colorization process. This algorithm primarily consists of two sub-networks: the visual enhancement network and the colorization network. The visual enhancement network, based on the U-Net architecture, is mainly employed for denoising and deblurring video frames to enhance image visual quality and recover details. The colorization network utilizes an attention mechanism to explore semantic correspondences between multiple example frames and grayscale frames and then transfers the corresponding colors to the entire grayscale frame based on semantic information, generating accurate and spatial-temporarily consistent colors.

Due to non-local semantic correspondences in existing methods, adverse effects such as color bleedings between objects and color averaging have occurred. To improve the colorization performance, Akimoto *et al.*[56] proposed a method focused on reducing color bleeding. This method utilizes a self-attention based network to assign the reference frame and the previous frame to the same group. Additionally, this method uses generated semantic masks of objects in each frame to guide the colorization process for consistent color propagation. This method primarily comprises three components: the estimation process of spatial-temporal correspondence between the reference frame and the target frame, the color transfer process, and the color refinement process utilizing information from adjacent frames.

In previous exemplar-based colorization methods, using a single reference image often fails to cover all objects in the video clip, leading to color transfer errors from the reference image to the frames to be colorized. Therefore, Yang *et al.*[18] proposed a bi-directional semantic feature fusion scheme, introducing two example frames at the beginning and end of each frame sequence. In this method, a semantic sub-network is initially employed to obtain a pair of semantic correspondences between the input grayscale frame and the two exemplars. Then the bi-directional semantic correspondences are combined to warp the colors from exemplars based on the temporal clues. Finally, edge detection and semantic segmentation information as guidance information is inputted together with wrapped colors into the colorization sub-network to generate more accurate colors.

In these methods mentioned above, the selection of reference images largely impacts the quality of colorization. As a result, searching suitable reference images for real-world grayscale videos from the Internet is challenging to users, even with the assistance of automatic search systems.

5.4 Fully Automatic Methods

Due to the guidance of reference images, exemplar-based colorization methods can achieve relatively good colorization results. However, acquiring suitable reference images is challenging which limits its applicability. Therefore, to reduce the complicated work of obtaining suitable references, fully automatic video colorization methods are becoming increasingly popular. Fully automatic video colorization methods aim at automatically transforming grayscale videos into color videos without relying on any colorized example image or scribble hint during the inference process.

However, it is noted that fully automatic colorization typically involves a frame-by-frame prediction process, which may lead to temporal inconsistencies. This is because there is a lack of explicit color references for each frame in the video, resulting in color changes of the same object in different frames. In the following discussion, we explore the implementation of various methods and analyze the potential advantages and drawbacks of these methods.

To address the issue of automatic video colorization without annotated data and user guidance, Lei and Chen[24] proposed to employ k-nearest neighbors (kNN) to search for pixel-pair similarity in the feature space. It introduces a time loss function to constrain temporal consistency. Although video colorization is a multi-modality problem, the method proposes the diversity perceptual loss to generate multiple colorized videos to differentiate multiple colorization modes. In [24], a two-stage network structure is employed, where both networks F and G are based on the U-Net architecture. The network F is used to convert grayscale video frames into coarse colorization results, which are then passed to the network G to obtain finer colorization results. Additionally, it employs VGG-19[36] and PWC-Net[57] for feature extraction from input images and optical flow computation, respectively.

In previous fully automatic video colorization methods, there are severe flickering artifacts and suboptimal colorization effects. Therefore, Liu *et al.*[23] proposed a method that utilizes inter-frame information leveraged by optical flow and then propagates the colors from two anchor frames to intermediate frames by bi-directional propagation. The optical flow between adjacent frames is estimated by the FlowNet2 network[58] and then utilized to align inter-frame features. By leveraging optical flow techniques,

color information from the first frame $\boldsymbol{x}_1$ and the last frame $\boldsymbol{x}_N$ is propagated to the intermediate frames $\{\boldsymbol{x}_t\}_{t=2}^{N-1}$. Without employing any loss with the ground-truth color video, the method also introduces a self-regularized learning scheme to minimize differences of predictions at different time steps to learn temporal consistency, thereby free from the influence of training or testing data. Similar to image-based colorization models, this network also consists of a feature extraction module and a color mapping module. The feature extraction module is initially used to extract features from the anchor frames. Subsequently, information is propagated frame by frame in both forward and backward directions, and the relevant deep features are then fed into the color mapping module to predict the chromatic channels of grayscale frames. Additionally, to integrate the features propagated in both directions, this method employs a feature fusion module.

To address the two main challenges commonly found in previous video colorization methods: temporal consistency and the integration of the colorization network with the refinement network, some strategies have been proposed by researchers. Zhao *et al.*[27] proposed an end-to-end hybrid-recurrent network based on a generative adversarial network (GAN), introducing a dense long-term loss to minimize temporal differences between frames over an extended period. GAN-based methods can suffer from challenges in dealing with the ill-posed colorization problem due to the limited representation space of GANs. In this method, the generator mainly consists of three components: a global feature extractor, a placeholder feature extractor, and an encoder-decoder. The global feature extractor is responsible for encoding the global semantics of grayscale frames, and the placeholder feature extractor encodes the semantics of previous color frames. The encoder-decoder utilizes the U-Net architecture to colorize grayscale frames. Both the global feature extractor and the placeholder feature extractor employ a fully convolutional ResNet-50-IN network[37], while the discriminator adopts a PatchGAN architecture[59].

Automatic colorization methods are effective to some extent, but they also have some drawbacks. There are relatively definite colors for objects such as flags, clothes, and buildings in practical colorization tasks with specific era backgrounds or artistic styles. The automatic colorization methods often struggle to meet these requirements.

6 Categorization from Novel Perspective

How to maintain spatial-temporal consistency is vital for video colorization. To better understand the development of existing methods in terms of spatial-temporal consistency, we review video colorization methods from a novel perspective.

6.1 Spatial Consistency

During the process of video colorization, there are numerous factors that influence spatial consistency in videos. Below are listed several strategies aiming at preserving the spatial consistency of videos.

6.1.1 Feature Extraction Models

In the task of video colorization, feature extraction models are capable of learning spatial features (such as object shapes and textures) and temporal features (such as motion information and dynamic changes) present within the video. These features serve as valuable assistance to the colorization model, aiding it in better understanding the content of the video and its color distribution. Moreover, feature extraction models also help reduce the complexity and redundancy of original data, thereby enhancing the efficiency of the video colorization model.

VGG-Based. These methods[12, 16, 18, 19, 24, 26, 30] utilize the feature representation capabilities of the VGG networks[36] for visual data. Zhang *et al.*[12] employed a pre-trained VGG-19[36] to extract features from both the input video frames and the reference frames to compute the similarity between them. Lei and Chen[24] augmented the input to the network by adding hyper-column features extracted from the VGG-19 network[36].

ResNet-Based. These methods employ residual networks (ResNets[37]) which use deeper layers and skip connections, giving the network the ability to capture and represent complex features[1, 27, 33]. Zhang *et al.*[33] opted for the more efficient ResNet-50[37] to gain better features in their study. Zhao *et al.*[27] have used pre-trained ResNet-50-IN[37] as a feature extractor to provide semantics for the network to identify colors for objects with similar edges.

Other Networks. In addition to the above methods, the utilization of features derived from large-scale pre-trained visual models[60] is gaining increasing popularity. These models can effectively model non-local and semantic information and are robust in handling

complex scenarios. These attributes underscore their potential significantly to revolutionize the landscape of video colorization techniques.

6.1.2 Spatial Color Consistency

In addition to choosing different feature extraction networks, spatial relationships between neighboring pixels are often considered to ensure spatial color consistency. For example, Lei and Chen[24] employed a self-regularization loss, which performs the kNN on both the predicted frame and the ground-truth frame during training to make sure that the pixels in the predicted frame at the same spatial locations as the kNN pixels in the ground truth frame have similar colors.

6.1.3 Prior Knowledge

Methods based on prior knowledge can help maintain the spatial consistency of videos. For instance, in [12, 15, 18] semantic information and color distribution features from reference frames are utilized to provide prior knowledge.

6.1.4 Edge Information

The edge information between objects is crucial for maintaining spatial consistency during the coloring process. Neglecting edge information may lead to color bleeding. For example, [18] effectively alleviates color bleeding by utilizing edge loss.

6.2 Temporal Consistency

Temporal consistency is of vital importance in video colorization tasks for several reasons: provides visual coherence in the perceived flow of frames, ensures contextual relevance of added color information across frames, enhances realism by mimicking the consistent color behavior of real-world videos, and contributes to narrative continuity that may be influenced by color. Fig.2(c) shows that recurrent-based methods dominate most methods.

3D Convolutional Networks. These types of networks[13, 15, 17, 25, 29, 31] directly handle time series data and capture temporal continuity in videos. Iizuka and Simo-Serra[15] employed 3D convolutions to manage multiple input frames and reference images concurrently, thereby enhancing spatial-temporal consistency. Shi *et al.*[31] proposed a temporal refinement network to learn spatial-temporal features through 3D convolutions to ensure the temporal color consistency of the results. 3D convolutions can process temporal sequences, inherently capturing spatial-temporal dependencies. This allows for coherent color transitions across video frames. In addition, by analyzing the time dimension, these networks can decode context better, leading to more accurate colorization choices, especially in dynamic scenes. Besides, 3D convolutional networks can automatically learn discriminative features for colorization without manual intervention, simplifying the pre-processing stage. However, there are several drawbacks of 3D convolutional networks. The complexity of 3D convolution operations leads to increased computational requirements and processing time. Moreover, 3D convolutional networks typically require more memory due to their consideration of an additional dimension.

Optical Flow Estimation. Optical-flow based algorithms[1, 3] calculate motion between pixels or features, used to maintain color consistency between continuous video frames. Optical flow estimation is good at understanding and interpreting object motion between sequential video frames. This can significantly aid in predicting the colorization attributes of moving objects, leading to more accurate and visually consistent results. In addition, optical flow estimation can promote spatial coherence in the video resulting in a smoother transition of colors between the frames. Despite the numerous advantages, optical flow estimation algorithms expose a sensitivity to disturbance and abrupt changes in illumination, which may precipitate imprecise flow estimations consequently causing colorization inaccuracies. And large displacements between frames can be challenging for optical flow estimation techniques to handle, which can influence the color consistency across frames.

Recurrent Neural Networks. Recurrent neural networks (RNNs)[1, 12, 16, 19, 20, 24, 26, 27, 30, 33] and its variants (like LSTM[61] or GRU[62]) can handle sequence data, establishing temporal dependencies between video frames for consistency. Zhang *et al.*[12] took the result of the previous frame as input to preserve temporal consistency when colorization the current frame. Zhao *et al.*[27] employed a placeholder feature extractor that serves as a feedback connection to encode the semantics of the previous colorized frame in order to maintain spatial-temporal consistency. Lei and Chen[24] took the i-th colorized candidate images from

frame t and frame $t+1$ as well as two confidence maps as input, and then employed the refinement network to output a colorized video frame for frame t. As RNNs are designed to process sequential data, they can sustain color consistency in consecutive frames, producing a smoother, more visually appealing result. Moreover, RNNs can handle sequences of varying lengths, making them versatile for videos of different durations. However, despite their theoretical capacity to model long-term dependencies, in practice, vanilla RNNs often falter in learning from long sequences due to what is termed as the "vanishing gradient" problem. Short-term dependencies tend to be captured more effectively than longer ones.

Bi-Directional Propagation Networks. Unlike unidirectional approaches, bi-directional propagation[18, 22, 23] incorporates both forward and backward temporal data, resulting in more robust color transformation planning. Yang *et al.*[18] employed a bi-directional propagation model to aggregate information efficiently from both exemplars. By propagating and cross-verifying color information in both temporal directions, bi-directional methods can generate more accurate and natural colorization. Moreover, bi-directional propagation can enhance temporal consistency by optimizing the coherence between forward and backward propagation, achieving more fluid color transitions in videos. However, the complexity of bi-directional propagation algorithms is generally higher than that of their unidirectional counterparts, resulting in increased computational demand and processing time. In addition, due to the requirement of learning color propagation rules in both directions, bi-directional methods may necessitate a larger training dataset.

7 Performance Evaluation

In this section, we evaluate the performance of representative video colorization methods based on deep learning techniques.

Table 2 summarizes the performance comparison of various representative deep learning methods on three commonly used video colorization datasets (the DAVIS dataset[48], Videvo dataset[1], and NVCC 2023 dataset[63]). It is worth noting that all algorithms adopt the same training strategy. The methods developed by Zhang *et al.*[64], Kang *et al.*[65], and Ji *et al.*[66] are based on image colorization techniques. The results indicate that directly applying image colorization methods to video colorization is not satisfactory. This is because image-based colorization methods only consider the colorization performance of the individual video frame, ignoring the temporal consistency across frames in the video sequence. Liu *et al.*[23] employed fully automatic approaches to design video colorization networks, achieving better temporal consistency compared with image colorization networks. This is because these methods utilize optical flow or 3D convolutions to align video frames. However, the fully automated methods have relatively high FID value, that is, relatively poor colorization performance on all three datasets, as Table 2 shows. Compared with the strategies mentioned above, exemplar-based methods are widely used to generate more vibrant colors of video frames. Iizuka and Simo-Serra[15], Zhang *et al.*[12], and Yang *et al.*[18] transferred color from reference to video frames, and these methods achieve good performance in terms of the FID metric on the DAVIS dataset[48], Videvo dataset[1], and NVCC 2023 dataset[63]. Current exemplar-based methods also introduce optical flow or 3D convolutions into the networks for better temporal consistency and these methods can achieve better colorization performance due to the guidance of reference frames compared with previous fully automated approaches.

Table 2. Performance Evaluation of Representative Video Colorization Methods Conducted on Three Commonly Used Video Colorization Datasets

Method	Category	DAVIS[48]				Videvo[1]				NVCC 2023[63]			
		PSNR	SSIM	FID	CDC	PSNR	SSIM	FID	CDC	PSNR	SSIM	FID	CDC
Zhang *et al.*[64]	Image-based	30.90	0.959	114.56	0.004 903	31.18	0.959	79.49	0.002 526	30.64	0.939	71.64	0.003 436
Kang *et al.*[65]	Image-based	30.54	0.934	85.41	0.004 168	30.75	0.934	54.91	0.002 189	30.38	0.957	64.08	0.002 622
Ji *et al.*[66]	Image-based	31.12	0.948	92.40	0.004 212	31.36	0.953	67.71	0.001 872	30.78	0.933	62.49	0.002 544
Lei and Chen[24]	Fully-automatic	30.53	0.951	110.88	0.006 880	30.01	0.949	82.07	0.010 254	28.65	0.927	81.42	0.008 882
Liu *et al.*[23]	Fully-automatic	31.10	0.955	116.64	0.003 743	31.30	0.957	80.71	0.001 694	30.46	0.936	72.73	0.002 443
Iizuka and Simo-Serra[15]	Exemplar-based	30.56	0.949	90.73	0.004 619	30.61	0.957	58.81	0.003 258	29.69	0.927	50.55	0.005 238
Zhang *et al.*[12]	Exemplar-based	33.24	0.951	70.21	0.003 853	33.11	0.957	55.01	0.001 925	32.03	0.930	35.43	0.002 694
Yang *et al.*[18]	Exemplar-based	33.72	0.936	45.18	0.004 168	34.07	0.967	32.32	0.001 861	33.43	0.949	26.63	0.002 864

Figs.3–5 show the comparison among one image-based colorization method, one fully automatic video colorization method, and two exemplar based video colorization methods. Image-based colorization methods[65] may not effectively colorize video frames, leading to noticeable color bleeding in the generated results. TCVC[23], as a fully-automated colorization method can result in desaturated colors in the scenes such as grass, airplanes, and pools. Across all three datasets, BiSTNet[18] achieves the optimal colorization results, indicating that exemplar-based video colorization methods can generate higher-quality colorization results. Meanwhile, even with similar architectures, different methods may produce different results, such as BiSTNet[18] and DeepExemplar[12], both of which are exemplar-based video colorization methods. The main reason for the performance variation is that BiSTNet[18] utilizes two reference frames at the beginning and end of the video sequence, along with a bidirectional temporal feature fusion module, while DeepExemplar[12] only uses the first frame of the video sequence as the reference frame.

Fig.3. Evaluation results of various state-of-the-art video colorization methods on the DAVIS datasets. The red, purple, and green boxes display the magnified effects of different areas of the selected image, respectively. (a) Gray. (b) CIC[64]. (c) DDColor[65]. (d) ColorFormer[66]. (e) FAVC[24]. (f) TCVC[23]. (g) DeepRemaster[15]. (h) DeeepExemplar[12]. (i) BiSTNet[18]. (j) GT.

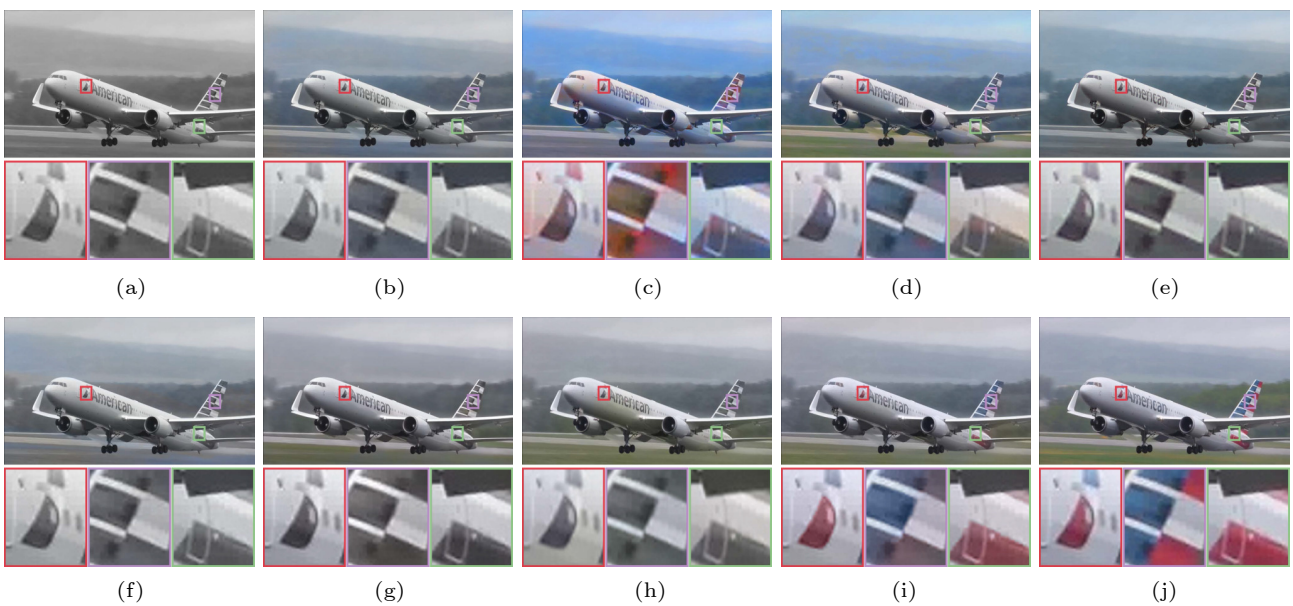

Fig.4. Evaluation results of various state-of-the-art video colorization methods on the Videvo datasets. The red, purple, and green boxes display the magnified effects of different areas of the selected image, respectively. (a) Gray. (b) CIC[64]. (c) DDColor[65]. (d) ColorFormer[66]. (e) FAVC[24]. (f) TCVC[23]. (g) DeepRemaster[15]. (h) DeeepExemplar[12]. (i) BiSTNet[18]. (j) GT.

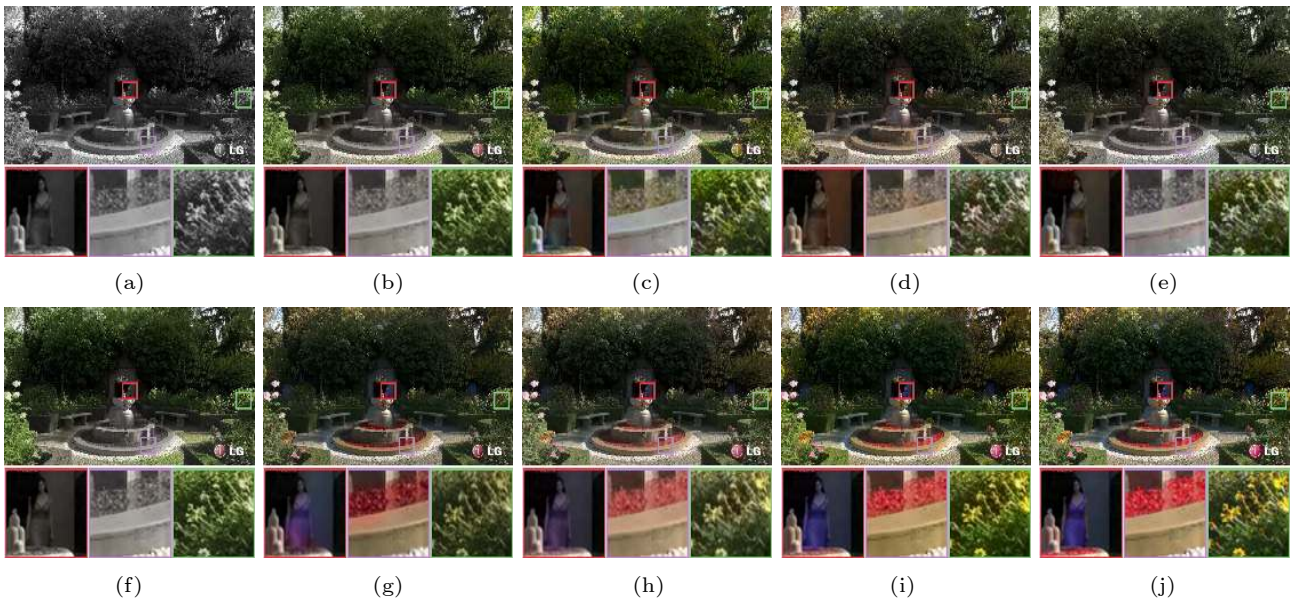

Fig.5. Evaluation results of various state-of-the-art video colorization methods on the NVCC2023 datasets. The red, purple, and green boxes display the magnified effects of different areas of the selected image, respectively. (a) Gray. (b) CIC[64]. (c) DDColor[65]. (d) ColorFormer[66]. (e) FAVC[24]. (f) TCVC[23]. (g) DeepRemaster[15]. (h) DeeepExemplar[12]. (i) BiSTNet[18]. (j) GT.

8 Loss Functions

The categorization of commonly used loss functions in video colorization methods is described in this section. It is worth noting that all loss functions introduced below are simplified formulations and they have variations when applied in specific applications.

8.1 Reconstruction-Based Loss Functions

L_1 Loss. This loss function is used to represent the absolute differences between the target and the prediction. It provides pixel-level supervision and encourages the output from the generator to closely resemble the ground truth as much as possible. The standard L_1 loss can be written as:

$$L_1 = \sum |\boldsymbol{z}_t - \hat{\boldsymbol{z}}_t|,$$

where $\hat{\boldsymbol{z}}_t$ is the ground truth at time t.

L_2 Loss. This function calculates the square of the differences between the target and the predicted values, making it more sensitive to outliers than L_1. The L_2 loss can be written as:

$$L_2 = \sum (\boldsymbol{z}_t - \hat{\boldsymbol{z}}_t)^2.$$

8.2 Perceptually-Based Loss Functions

Perceptual Loss. This is a more advanced loss function typically used in vision tasks. It is designed to measure perceptual and semantic differences, often computed in the feature space using high-level features extracted from pre-trained networks (e.g., VGG[36]). It enhances the quality and realism of the colorized images, making them closer to the real color images perceived by the human visual system. Here is a simplified form:

$$L_{\text{perc}} = \sum ||\phi(\boldsymbol{z}_t) - \phi(\hat{\boldsymbol{z}}_t)||_2^2,$$

where ϕ denotes a function extracting features.

Contextual Loss. Contextual loss[12] is proposed to inspire the resemblance of colors in the output to those in the reference. It measures the local feature similarity in the context of the whole image. This concept proves to be apt for transferring colors from semantically related regions.

Style Loss. Style loss[31] assesses the difference in style characteristics between the output and the target. Its role is to capture the style information of the original images, ensuring the preservation of their stylistic features during the generation process. The style loss can be written as:

$$L_{\text{style}} = \sum_{i=1}^{5} ||\mathcal{G}(\phi(\boldsymbol{z}_t(i))) - \mathcal{G}(\phi(\hat{\boldsymbol{z}}_t))||_1,$$

where $\mathcal{G}(\cdot)$ represents the Gram matrix computed for input features.

8.3 Spatially-Based Loss Functions

Smoothness Loss. This function encourages smoother and more continuous outputs by minimizing the difference between neighboring pixels. The smoothness loss can be written as:

$$L_{\text{smooth}} = \sum |\boldsymbol{z}_t^{(i,\,j+1)} - \boldsymbol{z}_t^{(i,\,j)}| + |\boldsymbol{z}_t^{(i+1,\,j)} - \boldsymbol{z}_t^{(i,\,j)}|,$$

where i, j denote the spatial location, L_{smooth} calculates the difference between adjacent pixels along the horizontal and vertical directions.

Self-Regularization Loss. Self-regularization loss[24] explicitly poses a penalty to the temporal consistency between adjacent frames. The color consistency between neighboring pixels can be enhanced in the bilateral space by the self-regularization loss. The formula for the self-regularization loss can be expressed as:

$$L_{\text{sr}} = \sum \sum ||M_{t+d\to t} \odot (\boldsymbol{z}_t - \mathcal{W}(\boldsymbol{z}_{t+d}, F_{t,\,t+d}))||_2,$$

where $F_{t,\,t+d}$ is the estimated optical flow between the current frame $\boldsymbol{z}_t$ and the previous frame $\boldsymbol{z}_{t+d}$, $\mathcal{W}(\cdot)$ denotes the operation of transforming images through the utilization of optical flow, $M_{t+d\to t} = \exp(-\alpha\,||\boldsymbol{z}_t - \mathcal{W}(\boldsymbol{z}_{t+d}, F_{t,\,t+d})||_2)$ is the visibility mask, $\odot$ denotes the element-wise multiplication operation.

Edge-enhancing Loss. Edge-enhancing loss[18] enforces the model to generate better-defined edges in the output. By utilizing edge-enhancement loss to constrain network training, color-bleeding issues occurring in neighboring objects in the image can be alleviated. The edge-enhancing loss is defined as:

$$L_{\text{edge}} = ||\mathcal{S}(\boldsymbol{x}_t) - \mathcal{S}(\boldsymbol{z}_t)||_2,$$

where $\boldsymbol{x}_t$ is a grayscale image, and $\mathcal{S}(\cdot)$ represents the Sobel filter used in [67].

8.4 Temporally-Based Loss Functions

Short-Term Temporal Loss. Short-term temporal loss[1] is set to ensure consistency between consecutive frames in the generated video sequence. This loss is designed to reduce the inconsistency between the consecutive generated outputs, thereby achieving smoother transitions from one frame to the next in the final result. The formula for the short-term temporal loss can be expressed as:

$$L_{\text{st}} = \frac{1}{N}\sum_{t=1}^{N} ||\boldsymbol{z}_{t+1} - \mathcal{W}(\boldsymbol{z}_t)||_1,$$

where $\boldsymbol{z}_t$ is the generated output at time t, $\boldsymbol{z}_{t+1}$ is the next output frame in the sequence, $\mathcal{W}(\cdot)$ denotes the operation of transforming images through the utilization of the optical flow, N is the total number of frames and $||.||_1$ denotes the L_1 norm (sum of the absolute differences).

Long-Term Temporal Loss. Long-term temporal loss[1] assists in maintaining long-time temporal consistency, by comparing the prediction of the present frame with the future frame:

$$L_{\text{lt}} = \frac{1}{N}\sum_{t=1}^{N} ||\boldsymbol{z}_{t+T} - \mathcal{W}(\boldsymbol{z}_t)||_1,$$

where T represents a constant time interval.

Temporal Consistency Loss. Temporal consistency loss[12] enforces the output to have a stable transition over time. It explicitly penalizes the color change along the flow trajectory. The formula for the temporal consistency loss can be expressed as:

$$L_{\text{tc}} = ||\mathcal{W}(\boldsymbol{z}_{t-1}, F_{t,\,t-1}) - \boldsymbol{z}_t||_2^2,$$

where $F_{t,\,t-1}$ is the estimated optical flow between the current frame $\boldsymbol{z}_t$ and the previous frame $\boldsymbol{z}_{t-1}$, $\mathcal{W}(\cdot)$ denotes the operation of transforming images through the utilization of optical flow. The optical flow exhibits the motion of the pixels in the video frames.

8.5 Diversity-Based Loss Functions

Adversarial Loss. In tasks like GANs, adversarial loss helps create more realistic outputs. It typically involves a game between two elements, a generator and a discriminator. Formally, the minimax game can be represented as:

$$L_{\text{adv}}(G, D) = E_r[\log D(r)] + E_n[\log(1 - D(G(n)))],$$

where G and D are the generator and discriminator, r is the real data, and n is the noise sample.

Diversity Loss. Diversity loss[24] is designed to generate multiple colorized videos which distinguish various solution modes. This loss not only enhances the availability of unique solutions but also significantly contributes to temporal coherence. It achieves this by mitigating the ambiguity inherent in colorization tasks through the generation of multiple operational modes. The diversity loss can be written as:

$$L_{\text{diversity}} = \sum_{t=1}^{N} \min_{i} \{||\phi(\boldsymbol{z}_t(i)) - \phi(\hat{\boldsymbol{z}}_t(i))||_1\} + \sum_{t=1}^{N} \sum_{i=1}^{d} \beta ||\phi(\boldsymbol{z}_t(i)) - \phi(\hat{\boldsymbol{z}}_t(i))||_1,$$

where $\boldsymbol{z}_t(i)$ represents the i-th colorized image output by the network, β is a decreasing sequence and d is set to 4 in [24].

Hard Example Mining Loss. Hard example mining loss[68] focuses more on hard examples during training for the robustness of the model. This loss function can automatically pay more attention to difficult regions, thereby encouraging the model to produce clearer boundaries. The hard example mining loss is defined as:

$$L_{\text{hard}} = \frac{||M \odot (f - \hat{f})||_1}{||M||_1} + \lambda \times \frac{||M^{\text{h}} \odot (f - \hat{f})||_1}{||M^{\text{h}}||_1},$$

where M^{h} represents the binary mask of hard regions, f denotes the computed optical flow, and $\hat{f}$ represents the ground-truth flow.

8.6 Information Theoretic Loss Functions

Cross Entropy Loss. Cross entropy loss is the preferred loss function for binary classification tasks. It measures the dissimilarity between the ground truth and the estimated probabilities. For video colorization tasks, we can discretize pixel values into multiple individual numerical entities to function as classification labels, thereby facilitating the calculation of cross-entropy loss.

9 Future Research Prospects

Advancements in video colorization have been noteworthy in the past decade, with profound implications for numerous applications in media, entertainment, and digital artistry. However, despite progress, various research avenues remain open for exploration.

Temporal Consistency. One of the key challenges in video colorization is maintaining temporal consistency across frames. While individual frames might be colorized accurately, subtle differences between adjacent frames can result in jittering or flickering effects. Future research might focus on developing more sophisticated algorithms to ensure smooth transitions and temporal coherence in colorized videos.

Improved Training Data. Deep learning based methods for video colorization require large, diverse datasets for training. Currently, the utility of these models is limited by the availability and diversity of such datasets. More accurate and diverse training data, potentially gathered from a wider array of sources, could improve the reliability and generalizability of these models.

Interactive and User-Guided Approaches. The capacity to incorporate user inputs into video colorization models can improve accuracy and user satisfaction. Future research could work towards making these models more interactive, allowing users to have more control over the colorization process.

Integration with Other Video Enhancement Techniques. There is great potential in the intersection of video colorization with other video enhancement processes like super-resolution, noise reduction, and frame interpolation. Further development of integrated models could lead to comprehensive video restoration and enhancement solutions.

Real-Time Processing Capabilities. Efforts could be directed toward improving the computational efficiency of video colorization methods to accommodate real-time colorization of live video streams.

10 Emerging Trends

Self-Superviesed Learning for Video Colorization. In tasks of video colorization, self-supervised learning can be employed to train colorization models when datasets are relatively small. The core of self-supervised learning lies in spatiotemporal consistency in videos. Since consecutive frames in video sequences exhibit significant color similarity, this can serve as a regularization constraint to guide learning and ensure temporal consistency. Additionally, in terms of spatial consistency, powerful pre-trained networks can be leveraged to extract features from video frames, enabling the network to understand the content and structure of the images and infer the color information of video frames.

Unpaired Learning for Video Colorization. To address the issue of having only grayscale video frame datasets in video colorization tasks, unpaired learning methods can be employed. In unpaired learning, typically two datasets are required: one comprising grayscale video frames and the other containing colored video frames. Subsequently, using the generator and discriminator components of a GAN, the conversion of grayscale frames to colored frames and the evaluation of the colored frames are carried out sepa-

rately. Additionally, content consistency and temporal consistency of the generated colored frames can be ensured by employing cycle consistency loss and temporal consistency loss respectively.

Large Models for Video Colorization. In the era of large models, these models are capable of generating many high-quality images and videos, indicating that they encapsulate rich knowledge. Therefore, we can extract this knowledge from these large models as prior knowledge for the colorization network, and then utilize this prior knowledge to guide the video colorization process.

11 Conclusions

In this survey, an extensive overview of various techniques involved in video colorization was provided. On the basis of user interaction, video colorization techniques can be broadly divided into four categories: fully automatic colorization, scribble-based colorization, optical-flow based colorization, and exemplar-based colorization. We discussed the strengths and weaknesses of different methods and provided an overview of the loss functions adopted by various colorization methods. Additionally, we compared and analyzed the performance of various video colorization methods on benchmark datasets. Through comparisons, exemplar-based video colorization methods guided by examples demonstrate better colorization performance compared with the other methods. While deep learning based video colorization techniques have made significant advancements, current methods still face numerous challenges. Therefore, we outlined the trends of self-supervised learning, unpaired learning, and the large models in the field of video colorization, aiming to provide insights for researchers.

Conflict of Interest Jin-Hui Tang is an associate editor for Journal of Computer Science and Technology and was not involved in the editorial review of this article. All authors declare that there are no other competing interests.

References

[1] Lai W S, Huang J B, Wang O, Shechtman E, Yumer E, Yang M H. Learning blind video temporal consistency. In *Proc. the 15th European Conference on Computer Vision*, Oct. 2018, pp.170–185. DOI: 10.1007/978-3-030-01267-0_11.

[2] Bonneel N, Tompkin J, Sunkavalli K, Sun D Q, Paris S, Pfister H. Blind video temporal consistency. *ACM Trans. Graphics*, 2015, 34(6): 196. DOI: 10.1145/2816795.2818107.

[3] Lei C Y, Xing Y Z, Ouyang H, Chen Q F. Deep video prior for video consistency and propagation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2023, 45(1): 356–371. DOI: 10.1109/TPAMI.2022.3142071.

[4] Yatziv L, Sapiro G. Fast image and video colorization using chrominance blending. *IEEE Trans. Image Processing*, 2006, 15(5): 1120–1129. DOI: 10.1109/TIP.2005.864231.

[5] Sheng B, Sun H Q, Magnor M, Li P. Video colorization using parallel optimization in feature space. *IEEE Trans. Circuits and Systems for Video Technology*, 2014, 24(3): 407–417. DOI: 10.1109/TCSVT.2013.2276702.

[6] Doğan P, Aydın T O, Stefanoski N, Smolic A. Key-frame based spatiotemporal scribble propagation. In *Proc. the 2015 Eurographics Workshop on Intelligent Cinematography and Editing*, May 2015, pp.13–20. DOI: 10.2312/wiced.20151073.

[7] Paul S, Bhattacharya S, Gupta S. Spatiotemporal colorization of video using 3D steerable pyramids. *IEEE Trans. Circuits and Systems for Video Technology*, 2017, 27(8): 1605–1619. DOI: 10.1109/TCSVT.2016.2539539.

[8] Jacob V G, Gupta S. Colorization of grayscale images and videos using a semiautomatic approach. In *Proc. the 16th IEEE International Conference on Image Processing*, Nov. 2009, pp.1653–1656. DOI: 10.1109/ICIP.2009.5413392.

[9] Ben-Zrihem N, Zelnik-Manor L. Approximate nearest neighbor fields in video. In *Proc. the 28th IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2015, pp.5233–5242. DOI: 10.1109/CVPR.2015.7299160.

[10] Xia S F, Liu J Y, Fang Y M, Yang W H, Guo Z M. Robust and automatic video colorization via multiframe reordering refinement. In *Proc. the 23rd IEEE International Conference on Image Processing*, Sept. 2016, pp.4017–4021. DOI: 10.1109/ICIP.2016.7533114.

[11] Heu J H, Hyun D Y, Kim C S, Lee S U. Image and video colorization based on prioritized source propagation. In *Proc. the 16th IEEE International Conference on Image Processing*, Nov. 2009, pp.465–468. DOI: 10.1109/ICIP.2009.5414371.

[12] Zhang B, He M M, Liao J, Sander P V, Yuan L, Bermak A, Chen D. Deep exemplar-based video colorization. In *Proc. the 32nd IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2019, pp.8044–8053. DOI: 10.1109/CVPR.2019.00824.

[13] Vondrick C, Shrivastava A, Fathi A, Guadarrama S, Murphy K. Tracking emerges by colorizing videos. In *Proc. the 15th European Conference on Computer Vision*, Sept. 2018, pp.391–408. DOI: 10.1007/978-3-030-01261-8_24.

[14] Meyer S, Cornillère V, Djelouah A, Schroers C, Gross M H. Deep video color propagation. In *Proc. the 29th British Machine Vision Conference*, Sept. 2018, Article No. 128. DOI: 10.3929/ethz-b-000319608.

[15] Iizuka S, Simo-Serra E. DeepRemaster: Temporal source-reference attention networks for comprehensive video enhancement. *ACM Trans. Graphics*, 2019, 38(6): Article No.176. DOI: 10.1145/3355089.3356570.

[16] Liu Y X, Zhang X Y, Xu X G. Reference-based video colorization with multi-scale semantic fusion and temporal augmentation. In *Proc. the 28th IEEE International Conference on Image Processing*, Sept. 2021, pp.1924–1928. DOI: 10.1109/ICIP42928.2021.9506422.

[17] Yang Y, Liu Y, Yuan H, Chu Y H. Deep colorization: A channel attention-based CNN for video colorization. In *Proc. the 5th International Conference on Image and Graphics Processing*, Jan. 2022, pp.275–280. DOI: 10.1145/3512388.3512428.

[18] Yang Y X, Pan J S, Peng Z Z, Du X Y, Tao Z L, Tang J H. BiSTNet: Semantic image prior guided bidirectional temporal feature fusion for deep exemplar-based video colorization. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2024. DOI: 10.1109/TPAMI.2024.3370920. (early access)

[19] Wan Z Y, Zhang B, Chen D D, Liao J. Bringing old films back to life. In *Proc. the 35th IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2022, pp.17673–17682. DOI: 10.1109/CVPR52688.2022.01717.

[20] Zhao Y Z, Po L M, Liu K C, Wang X H, Yu W Y, Xian P F, Zhang Y J, Liu M Y. SVCNet: Scribble-based video colorization network with temporal aggregation. *IEEE Trans. Image Processing*, 2023, 32: 4443–4458. DOI: 10.1109/TIP.2023.3298537.

[21] Jampani V, Gadde R, Gehler P V. Video propagation networks. In *Proc. the 30th IEEE Conference on Computer Vision and Pattern Recognition*, Jul. 2017, pp.3154–3164. DOI: 10.1109/CVPR.2017.336.

[22] Liu S F, Zhong G Y, De Mello S, Gu J W, Jampani V, Yang M H, Kautz J. Switchable temporal propagation network. In *Proc. the 15th European Conference on Computer Vision*, Sept. 2018, pp.89–104. DOI: 10.1007/978-3-030-01234-2_6.

[23] Liu Y H, Zhao H Y, Chan K C K, Wang X T, Loy C C, Qiao Y, Dong C. Temporally consistent video colorization with deep feature propagation and self-regularization learning. *Computational Visual Media*, 2024, 10(2): 375–395. DOI: 10.1007/s41095-023-0342-8.

[24] Lei C Y, Chen Q F. Fully automatic video colorization with self-regularization and diversity. In *Proc. the 32nd IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2019, pp.3748–3756. DOI: 10.1109/CVPR.2019.00387.

[25] Kouzouglidis P, Sfikas G, Nikou C. Automatic video colorization using 3D conditional generative adversarial networks. In *Proc. the 14th International Symposium on Visual Computing*, Oct. 2019, pp.209–218. DOI: 10.1007/978-3-030-33720-9_16.

[26] Thasarathan H, Nazeri K, Ebrahimi M. Automatic temporally coherent video colorization. In *Proc. the 16th Conference on Computer and Robot Vision*, May 2019, pp.189–194. DOI: 10.1109/CRV.2019.00033.

[27] Zhao Y Z, Po L M, Yu W Y, Rehman Y A U, Liu M Y, Zhang Y J, Ou W F. VCGAN: Video colorization with hybrid generative adversarial network. *IEEE Trans. Multimedia*, 2023, 25: 3017–3032. DOI: 10.1109/TMM.2022.3154600.

[28] Salmona A, Bouza L, Delon J. Deoldify: A review and implementation of an automatic colorization method. *Image Processing on Line*, 2022, 12: 347–368. DOI: 10.5201/ipol.2022.403.

[29] Jampour M, Zare M, Javidi M. Advanced multi-GANs towards near to real image and video colorization. *Journal of Ambient Intelligence and Humanized Computing*, 2023, 14(9): 12857–12874. DOI: 10.1007/s12652-022-04206-z.

[30] Mahajan A, Patel N, Kotak A, Palkar B. An end-to-end approach for automatic and consistent colorization of gray-scale videos using deep-learning techniques. In *Proc. the 2020 International Conference on Machine Intelligence and Data Science Applications*, May 2021, pp.539–551. DOI: 10.1007/978-981-33-4087-9_45.

[31] Shi M, Zhang J Q, Chen S Y, Gao L, Lai Y K, Zhang F L. Reference-based deep line art video colorization. *IEEE Trans. Visualization and Computer Graphics*, 2023, 29(6): 2965–2979. DOI: 10.1109/TVCG.2022.3146000.

[32] Veluri B, Pernu C, Saffari A, Smith J R, Taylor M B, Gollakota S. NeuriCam: Key-frame video super-resolution and colorization for IoT cameras. arXiv: 2207.12496, 2022. https://arxiv.org/abs/2207.12496, May 2024.

[33] Zhang Q, Wang B, Wen W, Li H, Liu J. Line art correlation matching feature transfer network for automatic animation colorization. In *Proc. the 2021 IEEE Winter Conference on Applications of Computer Vision*, Jan. 2021, pp.3871–3880. DOI: 10.1109/WACV48630.2021.00392.

[34] Casey E, Pérez V, Li Z R. The animation transformer: Visual correspondence via segment matching. In *Proc. the 2021 IEEE/CVF International Conference on Computer Vision*, Oct. 2021, pp.11303–11312. DOI: 10.1109/ICCV48922.2021.01113.

[35] Zhao H Y, Wu W H, Liu Y H, He D L. Color2Embed: Fast exemplar-based image colorization using color embeddings. arXiv: 2106.08017, 2021. https://arxiv.org/abs/2106.08017, May 2024.

[36] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In *Proc. the 3rd International Conference on Learning Representations*, May 2015.

[37] He K M, Zhang X Y, Ren S Q, Sun J. Deep residual learning for image recognition. In *Proc. the 29th IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2016, pp.770–778. DOI: 10.1109/CVPR.2016.90.

[38] Chen S Q, Li X M, Zhang X L, Wang M D, Zhang Y, Han J T, Zhang Y. Exemplar-based video colorization with long-term spatiotemporal dependency. *Knowledge-Based Systems*, 2024, 284: 111240. DOI: 10.1016/j.knosys.2023.111240.

[39] Wang Z, Bovik A C, Sheikh H R, Simoncelli E P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Processing*, 2004, 13(4): 600–612. DOI: 10.1109/TIP.2003.819861.

[40] Heusel M, Ramsauer H, Unterthiner T, Nessler B, Hochreiter S. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Proc. the*

31st International Conference on Neural Information Processing Systems, Dec. 2017, pp.6629–6640. DOI: 10.5555/3295222.3295408.

[41] Zhang R, Isola P, Efros A A, Shechtman E, Wang O. The unreasonable effectiveness of deep features as a perceptual metric. In *Proc. the 31st IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp.586–595. DOI: 10.1109/CVPR.2018.00068.

[42] Hasler D, Suesstrunk S E. Measuring colorfulness in natural images. In *Proc. the SPIE 5007, Human Vision and Electronic Imaging VIII*, Jun. 2003, pp.87–95. DOI: 10.1117/12.477378.

[43] Xue T F, Chen B A, Wu J J, Wei D L, Freeman W T. Video enhancement with task-oriented flow. *International Journal of Computer Vision*, 2019, 127(8): 1106–1125. DOI: 10.1007/s11263-018-01144-2.

[44] Deng J, Dong W, Socher R, Li L J, Li K, Fei-Fei L. ImageNet: A large-scale hierarchical image database. In *Proc. the 22nd IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2009, pp.248–255. DOI: 10.1109/CVPR.2009.5206848.

[45] Soomro K, Zamir A R, Shah M. UCF101: A dataset of 101 human actions classes from videos in the wild. arXiv: 1212.0402, 2012. https://arxiv.org/abs/1212.0402, May 2024.

[46] Wu Z X, Wang X, Jiang Y G, Ye H, Xue X Y. Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In *Proc. the 23rd ACM International Conference on Multimedia*, Oct. 2015, pp.461–470. DOI: 10.1145/2733373.2806222.

[47] Feichtenhofer C, Pinz A, Zisserman A. Convolutional two-stream network fusion for video action recognition. In *Proc. the 29th IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2016, pp.1933–1941. DOI: 10.1109/CVPR.2016.213.

[48] Perazzi F, Pont-Tuset J, McWilliams B, Van Gool L, Gross M, Sorkine-Hornung A. A benchmark dataset and evaluation methodology for video object segmentation. In *Proc. the 29th IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2016, pp.724–732. DOI: 10.1109/CVPR.2016.85.

[49] Pont-Tuset J, Perazzi F, Caelles S, Arbeláez P, Sorkine-Hornung A, Van Gool L. The 2017 DAVIS challenge on video object segmentation. arXiv: 1704.00675, 2017. https://arxiv.org/abs/1704.00675, May 2024.

[50] Caelles S, Pont-Tuset J, Perazzi F, Montes A, Maninis K K, Van Gool L. The 2019 DAVIS challenge on VOS: Unsupervised multi-object segmentation. arXiv: 1905.00737, 2019. https://arxiv.org/abs/1905.00737, May 2024.

[51] Abu-El-Haija S, Kothari N, Lee J, Natsev P, Toderici G, Varadarajan B, Vijayanarasimhan S. YouTube-8M: A large-scale video classification benchmark. arXiv: 1609.08675, 2016. https://arxiv.org/abs/1609.08675, May 2024.

[52] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation. In *Proc. the 2015 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2015, pp.3431–3440. DOI: 10.1109/CVPR.2015.7298965.

[53] Li S Y, Zhao S Y, Yu W J, Sun W X, Metaxas D, Loy C C, Liu Z W. Deep animation video interpolation in the wild. In *Proc. the 34th IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2021, pp.6583–6591. DOI: 10.1109/CVPR46437.2021.00652.

[54] Zhang S H, Chen T, Zhang Y F, Hu S M, Martin R R. Vectorizing cartoon animations. *IEEE Trans. Visualization and Computer Graphics*, 2009, 15(4): 618–629. DOI: 10.1109/TVCG.2009.9.

[55] Levin A, Lischinski D, Weiss Y. Colorization using optimization. *ACM Trans. Graphics*, 2004, 23(3): 689–694. DOI: 10.1145/1015706.1015780.

[56] Akimoto N, Hayakawa A, Shin A, Narihira T. Reference-based video colorization with spatiotemporal correspondence. arXiv: 2011.12528, 2020. https://arxiv.org/abs/2011.12528, May 2024.

[57] Sun D Q, Yang X D, Liu M Y, Kautz J. PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In *Proc. the 31st IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp.8934–8943. DOI: 10.1109/CVPR.2018.00931.

[58] Ilg E, Mayer N, Saikia T, Keuper M, Dosovitskiy A, Brox T. FlowNet 2.0: Evolution of optical flow estimation with deep networks. In *Proc. the 30th IEEE Conference on Computer Vision and Pattern Recognition*, Jul. 2017, pp.1647–1655. DOI: 10.1109/CVPR.2017.179.

[59] Chang Y L, Liu Z Y, Lee K Y, Hsu W. Free-form video inpainting with 3D gated convolution and temporal patchGAN. In *Proc. the 16th IEEE/CVF International Conference on Computer Vision*, Oct. 2019, pp.9065–9074. DOI: 10.1109/ICCV.2019.00916.

[60] Oquab M, Darcet T, Moutakanni T, Vo H, Szafraniec M, Khalidov V, Fernandez P, Haziza D, Massa F, El-Nouby A, Assran M, Ballas N, Galuba W, Howes R, Huang P Y, Li S W, Misra I, Rabbat M, Sharma V, Synnaeve G, Xu H, Jegou H, Mairal J, Labatut P, Joulin A, Bojanowski P. DINOv2: Learning robust visual features without supervision. arXiv: 2304.07193, 2023. https://arxiv.org/abs/2304.07193, May 2024.

[61] Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Computation*, 1997, 9(8): 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

[62] Cho K, Van Merriënboer B, Bahdanau D, Bengio Y. On the properties of neural machine translation: Encoder-decoder approaches. arXiv: 1409.1259, 2014. https://arxiv.org/abs/1409.1259, May 2024.

[63] Kang X Y, Lin X H, Zhang K, Hui Z, Xiang W M, He J Y, Li X M, Ren P R, Xie X S, Timofte R, Yang Y X, Pan J S, Zheng Z, Qiyan P, Jiangxin Z, Jinhui D, Jinjing T, Chichen L, Li L Q, Liang Q R, Gang R, Liu X F, Feng S, Liu S, Wang H, Feng C Y, Bai F R, Zhang Y Q, Shao G Q, Wang X T, Lei L, Chen S Q, Zhang Y, Xu H N, Liu Z Y, Zhang Z, Luo Y, Zuo Z C. NTIRE 2023 video colorization challenge. In *Proc. the 36th IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*,

Jun. 2023, pp.1570–1581. DOI: 10.1109/CVPRW59228.2023.00159.

[64] Zhang R, Isola P, Efros A A. Colorful image colorization. In *Proc. the 14th European Conference on Computer Vision*, Oct. 2016, pp.649–666. DOI: 10.1007/978-3-319-46487-9_40.

[65] Kang X Y, Yang T, Ouyang W Q, Ren P R, Li L Z, Xie X S. DDColor: Towards photo-realistic image colorization via dual decoders. In *Proc. the 2023 IEEE/CVF International Conference on Computer Vision*, Oct. 2023, pp.328–338. DOI: 10.1109/ICCV51070.2023.00037.

[66] Ji X Z, Jiang B Y, Luo D H, Tao G P, Chu W Q, Xie Z F, Wang C J, Tai Y. ColorFormer: Image colorization via color memory assisted hybrid-attention transformer. In *Proc. the 17th European Conference on Computer Vision*, Oct. 2022, pp.20–36. DOI: 10.1007/978-3-031-19787-1_2.

[67] Kim E, Lee S, Park J, Choi S, Seo C, Choo J. Deep edge-aware interactive colorization against color-bleeding effects. In *Proc. the 2021 IEEE/CVF International Conference on Computer Vision*, Oct. 2021, pp.14647–14656. DOI: 10.1109/ICCV48922.2021.01440.

[68] Pan J H, Bai H R, Tang J H. Cascaded deep video deblurring using temporal sharpness prior. In *Proc. the 33rd IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2020, pp.3040–3048. DOI: 10.1109/CVPR42600.2020.00311.

Zhong-Zheng Peng is currently pursuing his Ph.D. degree in School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing. His research interests include image/video colorization, super-resolution, and other restoration tasks.

Yi-Xin Yang is currently pursuing his Ph.D. degree in School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing. He received his M.S. degree in electrical engineering from University of California, Riverside, and B.S. degree in electronic information engineering from University of Electronic Science and Technology of China, Chengdu. His research interests include image/video colorization, super-resolution, and other restoration tasks.

Jin-Hui Tang received his B.E. and Ph.D. degrees from the University of Science and Technology of China, Hefei, in 2003 and 2008, respectively. He is currently a professor with the Nanjing University of Science and Technology, Nanjing. His research interests include multimedia analysis and computer vision.

Jin-Shan Pan received his Ph.D. degree in computational mathematics from the Dalian University of Technology, Dalian, in 2017. He is a professor of School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing. His research interest includes image deblurring, image/video analysis and enhancement, and related vision problems.