Automatic Video Colorization Using 3D Conditional Generative Adversarial Networks

Panagiotis Kouzouglidis[1], Giorgos Sfikas[1,2](✉), and Christophoros Nikou[1]

[1] Department of Computer Science and Engineering,
University of Ioannina, 45110 Ioannina, Greece
sfikas@cs.uoi.gr

[2] Information Technologies Institute, CERTH, 57001 Thessaloniki, Greece

Abstract. In this work, we present a method for automatic colorization of grayscale videos. The core of the method is a Generative Adversarial Network that is trained and tested on sequences of frames in a sliding window manner. Network convolutional and deconvolutional layers are three-dimensional, with frame height, width and time as the dimensions taken into account. Multiple chrominance estimates per frame are aggregated and combined with available luminance information to recreate a colored sequence. Colorization trials are run successfully on a dataset of old black-and-white films. The usefulness of our method is also validated with numerical results, computed with a newly proposed metric that measures colorization consistency over a frame sequence.

Keywords: Video colorization · Generative Adversarial Networks · Three-dimensional convolution · Black-and-white films

1 Introduction

In this paper, we address the problem of automatic colorization of monochrome digitized videos [8,9,13,15,16]. Perhaps the most straightforward practical application is to colorizing black-and-white footage from old films or documentaries. Video compression is another possible application of note [16].

Video colorization methods can be categorized according to the level of user interaction required. A group of methods assume that a partially colored frame exists in the video, where color has been manually annotated in the form of color seeds [7,12,16]. The method must then propagate color from these seeds to the rest of the frame, then to other frames in the video. Other methods assume instead that a reference colored image exists that is similar in content and structure to the target monochrome video frames [1,13,15]. These methods may or may not require user intervention; for example, in [14] the user can specify matching areas between the reference and the target frames. In reference image-based methods, the problem of video colorization is hence converted to

G. Bebis et al. (Eds.): ISVC 2019, LNCS 11844, pp. 209–218, 2019.
https://doi.org/10.1007/978-3-030-33720-9_16

the problem of how to propagate color from the reference frame to other frames and/or from frame to frame. Optical flow estimation has been used to guide frame-to-frame color propagation [9,13]. In [12], Gabor feature flow is used as alternative to standard optical flow as a more robust guide to color propagation. Naturally, methods of this vein work best for coloring short videos or frames coming from the same scene [13].

In the present work, we propose a learning-based method for video colorization. As such, we assume that a collection of colored frames exist, that will be used to train the model. In particular, the proposed method is based on an appropriately designed Generative Adversarial Network (GAN) [4]. GANs have gained a fair amount of traction in the last few years. Despite their being harder to train even more than standard neural networks, requiring the employment of various heuristics and careful choosing of hyperparameters to attain convergence to a Nash equilibrium [3,11], they have proven to be excellent generative models. The proposed model employs a conditional GAN (cGAN) architecture, popularized by the pix2pix model [5]. In the current work, convolutional and deconvolutional layers are 3D (height, width, time dimensions) to accomodate for the sequential nature of video data.

The main novel points of the current paper are as follows: (a) we present a model for learning-based automatic video colorization that can take advantage of the sequential nature of video, while avoiding the use of frame-by-frame color propagation techniques that come with their own inherent limitations (typically they require existing colored key frames and/or are practically applicable within a single shot). Other recent works use learning methods to color video via propagation [8], or via frame-by-frame image colorization, with each frame processed separately [6]; (b) we elaborate on the issue of video colorization evaluation and propose a quantitative colorization metric specifically for video; (c) we show that the proposed method creates colorization models that are transferable, in the sense that learning over a particular frame sequence produces a plausible output usable on a sequence of different content.

The rest of the paper is organized as follows. In Sect. 2, we briefly discuss preliminaries on adversarial nets and present the architecture and processing pipeline of the proposed video colorization method. In Sect. 3, we elaborate on existing numerical evaluation methods and propose a new metric to evaluate video colorization. In Sect. 4, we show numerical and qualitative results of our method, tested on a collection of old films. We close the paper with Sect. 5, where we discuss conclusions and future work.

2 Proposed Method

The proposed method assumes the existence of a training set consisting of a sequence of N_{train} colored frames, and a test set consisting of a sequence of N_{test} monochrome frames that are to be colorized. During the training phase, a cGAN model is used to learn how to color batches of C ordered frames. Hyperparameter C is fixed beforehand with $C << min\{N_{train}, N_{test}\}$. During the

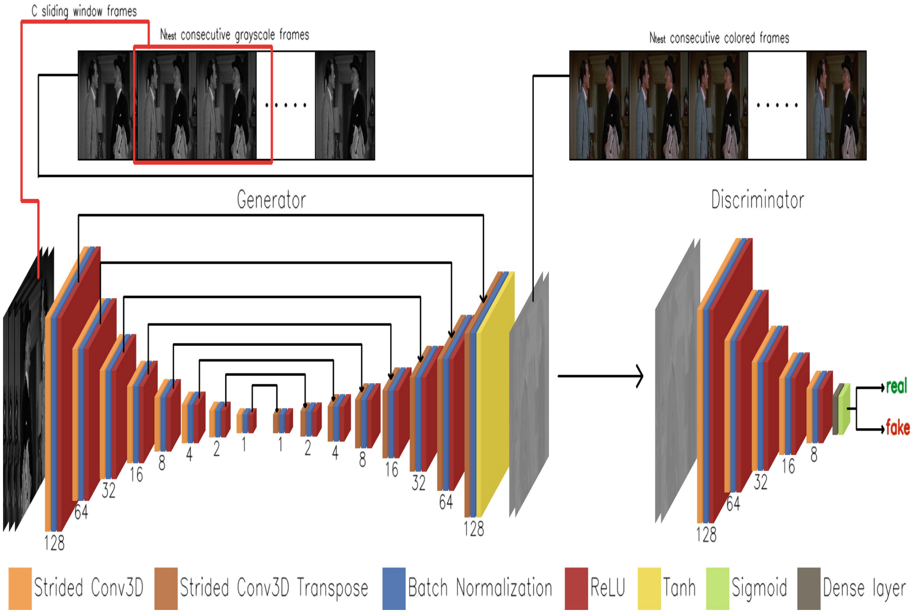

Fig. 1. Architecture of the proposed model. Sequences of luminance frames (‘L’ channels) are fed to the generator network, which creates chrominance channels for the corresponding frames (‘a’, ‘b’ channels). The discriminator network quantifies to what extend the generated chrominance sequence corresponds to a plausible colorization. As inputs and outputs are frame sequences, all convolutions are three-dimensional. During training, model weights are estimated through optimization of a GAN loss that effectively combines these constraints: how plausible the generated colorization is per frame *and as a sequence*, and how close the chrominance estimate is to the ground truth. In test time, the video is processed in a sliding window manner, producing C colorization estimates (the size of the sliding window) for each frame, which are then aggregated to obtain a single estimate per frame. The output chrominance is finally combined with the luminance input in order to recreate a sequence of colorized frames.

testing phase, the model is run on the input monochrome video in a sliding window manner. Windows are overlapping and move by a single frame at a time, thereby producing a set of C colorization estimates *for each* monochrome frame, hence C video colorization proposals. These C estimates are then combined to produce a single colorized output. In what follows we discuss the details of this process.

A GAN is a generative, neural network-based model that consists of two components, the generator network and the discriminator network. The cGAN architecture [5] that is employed as part of the proposed method, is a supervised variant of the original unsupervised GAN [4]. A cGAN learns a mapping from observed input x to target output y[1].

Formally, the objective to be optimized is:

$$\begin{aligned} \arg\min_{G}\max_{D}[\mathbb{E}_y[logD(y)] \\ +\mathbb{E}_x[log(1-D(G(x)))]] + \lambda\,\mathbb{E}_{x,y}[\|y-G(x)\|_1], \end{aligned} \tag{1}$$

with hyperparameter λ controlling the trade-off between the GAN (discriminator) loss and the L_1 loss. $D(\cdot)$ and $G(\cdot)$ correspond to the discriminator and generator respectively. The GAN loss quantifies how plausible the colorization output, while the L_1 loss forces the colorization to be close to the ground truth. We use representations in the CIE Lab color space (following e.g. [17]). For a monochrome frame sequence, only luminance is known beforehand. Input is a sequence of luminance channels (channel L) of C consecutive frames $x \in \mathbb{R}^{H \times W \times C \times 1}$, and the objective is to learn a mapping from luminance to chrominance (channels a,b) $y \in \mathbb{R}^{H \times W \times C \times 2}$ where H, W are frame dimensions.

The generator network is comprised of a series of convolutional and deconvolutional layers. Skip connections are added in the manner introduced by UNet [10]. As inputs and outputs are sequences of fixed-size frames, all convolutions and deconvolutions are three-dimensional (frame height, width and time dimensions). The encoder and decoder stacks comprise 8 strided convolutional/deconvolutional layers each (stride = 2), followed iteratively by batch normalization (BN) and rectified linear unit (ReLU) activation layers. Following [2], outputs are forced to lie in the $(-1, 1)$ range with a tanh activation layer at the end of the generator network, and only later renormalized to valid a, b chrominance values. The discriminator network is a 3D convolutional network comprising 5 convolutional layers iteratively followed by BN and ReLU layers. The discriminator is topped by a fully connected ("dense") layer and a sigmoid activation unit in order to map the image to a real/fake probability figure.

At test time, we use the generator in a sliding window fashion over the footage to be colorized. Hence, each frame is given as input to the generator at a total of C times, since C is the size of the sliding window. The produced C chrominance estimates $\chi_1, \chi_2, \cdots, \chi_C$[2] then need to be used to produce a single estimate $\hat{\chi}$. We can write $\hat{\chi}$ as a maximum-a-posteriori (MAP) estimate as:

$$\hat{\chi} = \arg\max_{\chi} p(\chi|\chi_1, \cdots, \chi_C) \tag{2}$$

where a prior distribution $p(\chi)$ can be assumed over possible a, b values in order to favor a particular chrominance setup. If identical distributions centered around each χ_i and an uninformative prior is used, the above formula simplifies as an average over all chrominance values per frame pixel: $\hat{\chi} = {}^1\!/_C \sum_{i=1}^{C} \chi_i$.

[1] Other variants of a cGAN are possible; for example, a noise variable z could be added to produce a non-deterministic output [5]. We employ a deterministic cGAN variant in this work.

[2] χ_i denotes the i^{th} colorization estimate for a frame. y denotes a colorization estimate for a sequence of C frames.

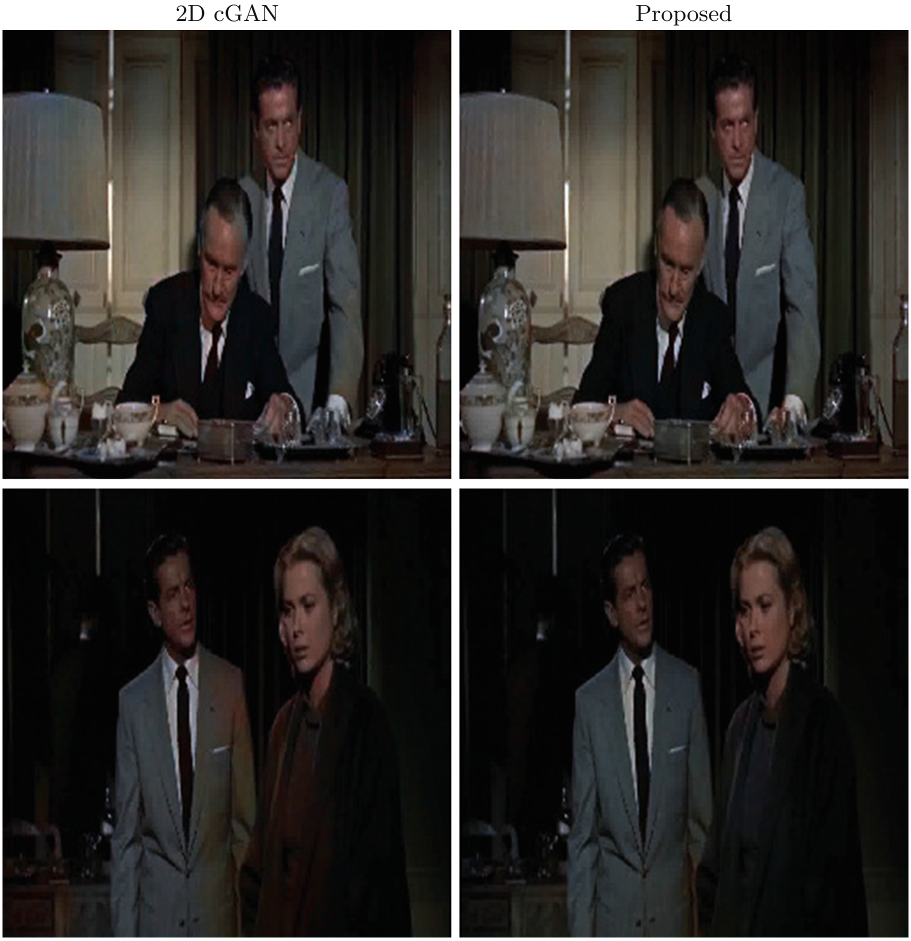

Fig. 2. Comparison of proposed model vs non-sequential 2D cGAN model. The proposed model produces better results than the non-sequential variant, as the former can take advantage of optical flow information, with its 3D convolution/deconvolution layers and estimate aggregation scheme. (Note for example how each method colorizes the hand of the standing actor on the top frame, or the color of the suit on the bottom frame). Depicted frames are samples from the film "Dial M for Murder".

Finally, the chrominance estimate is recombined with input luminance to recreate colored RGB frames for the input video. The architecture of the proposed model is summarized in Fig. 1.

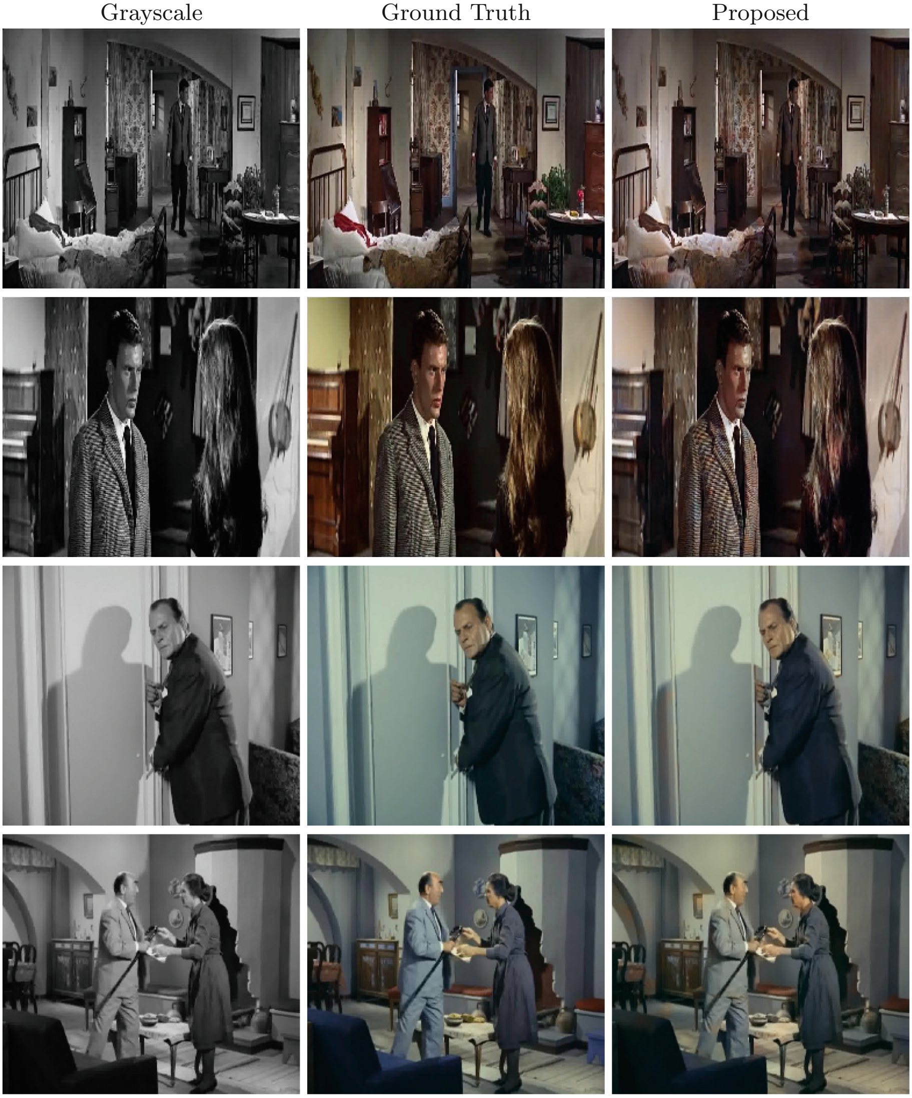

Fig. 3. Colorization results using our method. Depicted frames are samples from films: "Et Dieu..créa la femme" and "Tzéni, Tzéni" (2 top and 2 bottom rows respectively).

3 Metrics for Numerical Evaluation of Video Colorization

In this section we describe the metrics we use for numerical evaluation of video colorization. We use two metrics that measure per-frame colorization quality, also usable in single-image colorization. Furthermore, we propose a new metric suitable for video colorization in particular.

Peak Signal-to-Noise Ratio (PSNR): PSNR is calculated per each test frame in the RGB colorspace, and their mean is reported as a benchmark over the whole video.

Raw Accuracy (RA): Raw Accuracy, used in [17] to evaluate image colorization, is defined in terms of accuracy of predicted colors over a varying threshold. Colors are classified as correctly predicted if their Euclidean distance in the *ab* space is lower than a threshold. Accuracy is computed over color values for every pixel position and frame. Integrating over the curve that is produced by taking into account varying threshold yields the RA metric. We integrated from 0 to 150 distance units as in [17].

Color Consistency (CC): The aforementioned metrics measure strictly the quality of colorization of each frame separately. We propose and use a metric to measure both per-frame quality and also the consistency of the choice of colors between consecutive frames. Such a metric can, for example, penalize erratic differences in colorization from frame to frame, that would otherwise be "invisible" to the other metrics, borrowed from single image restoration/colorization. We define color consistency over sets of two consecutive colorization predictions $\hat{\chi}^{(t)}, \hat{\chi}^{(t+1)}$ and corresponding ground truth values $\chi^{(t)}, \chi^{(t+1)}$ as

$$CC^{(t,t+1)} = {}^1\!/_{HW} \sum_{i=1}^{H} \sum_{j=1}^{W} {}^1\!/_2 [A_{ij}^{(t)} + A_{ij}^{(t+1)}] A_{ij}^{(t \times t+1)} \tag{3}$$

where affinity matrices $A^{(t)}$ and $A^{(t \times t+1)}$ are defined as

$$A_{ij}^{(t)} = \phi(||\chi_{ij}^{(t)} - \hat{\chi}_{ij}^{(t)}||),$$
$$A_{ij}^{(t \times t+1)} = \phi\left(\left\| ||\chi_{ij}^{(t)} - \chi_{ij}^{(t+1)}|| - ||\hat{\chi}_{ij}^{(t)} - \hat{\chi}_{ij}^{(t+1)}|| \right\|\right),$$

with function $\phi(\cdot)$ a positive, strictly decreasing function that is used to convert distances to similarities. We use $\phi_{ij}(X) = \lfloor 60 X_{ij}/(\max(X) + \epsilon) + 1 \rfloor^{-1}$. Total CC over a video sequence is calculated as the average CC over all consecutive frames. Higher values correspond to better results.

4 Experiments

We have tested our method over a collection of old films: (a) "Dial M for Murder" (USA, 1954; 63,243 frames) [18] (b) "Et Dieu..créa la femme" (France, 1956; 54,922 frames) [19] (c) "Tzéni, Tzéni" (Greece, 1965; 58,932 frames) [20] (d) "A streetcar named desire" (USA, 1951; 18,002 frames) [21] (e) "Twelve angry men" (USA, 1957; 12,000 frames) [22]. Frames were sampled off these films at 10 fps. Films $(a), (b), (c)$ are colored, while (d) and (e) are originally black-and-white. Consequently, only the colored films could be used for training, while the black-and-white ones could be used only for testing with a colorizer trained on another film.

We have first experimented with training and testing on different parts of the same (colored) film. For training/testing we have used the first 75%/last 25%

from each of the colored films. The proposed 3D cGAN model was used, with model parameters set to $C = 3$ (sliding window size), $\lambda = 100$ (GAN-L_1 loss tradeoff), and compared against a 2D cGAN model that learned to colorize each frame separately. We have also use data augmentation on our training set, with random horizontal flips (50% chance to use a flipped input during training) and gaussian additive noise ($\sim \mathcal{N}(0, 1.2e-3)$). For estimate aggregation (Eq. 2) we present results with an uninformative prior (preliminary tests with priors learned over data statistics did not give any definite improvement). We also compare with a greyscale baseline, i.e. the case where the "colorized" video estimate uses only luminance information. Numerical results can be examined in Table 1.

Table 1. Numerical results for colorization evaluation. Training and testing is performed on different clips of the same film. PSNR is measured in dB; RA and CC values are percentages. Higher values are better. The proposed model performs best, in all cases.

	PSNR	RA	CC
(a) "Dial M for Murder"			
Grayscale	32.69	96.55	73.09
2D cGAN	34.97	96.67	82.07
Proposed	**35.66**	**96.73**	**85.59**
(b) "Et Dieu..créa la femme"			
Grayscale	30.23	94.07	47.82
2D cGAN	32.08	95.17	56.67
Proposed	**32.32**	**95.31**	**58.80**
(c) "Tzéni, Tzéni"			
Grayscale	29.83	92.85	39.17
2D cGAN	31.44	93.87	50.81
Proposed	**31.77**	**94.14**	**55.16**

Qualititative results can be examined in Figs. 2 and 3. While in general both models fare satisfactorily, the proposed model can avoid erroneous colorizations in several cases (cf. Fig. 2). This point is validated by our numerical results, where we calculate the metrics presented in Sect. 3. While w.r.t. to PSNR and RA the proposed model still is better, it could be argued that the difference in the result is statistically insignificant. This is not the case with the proposed CC metric however, where the performance of the proposed model is markedly better. These results validate our expectation, as the 3D structure of the proposed model can take into account the sequential structure of the video, in contrast to its 2D counterpart.

We have also run tests for training and testing on different films. The case that is perhaps closest to a practical application of the current model is using

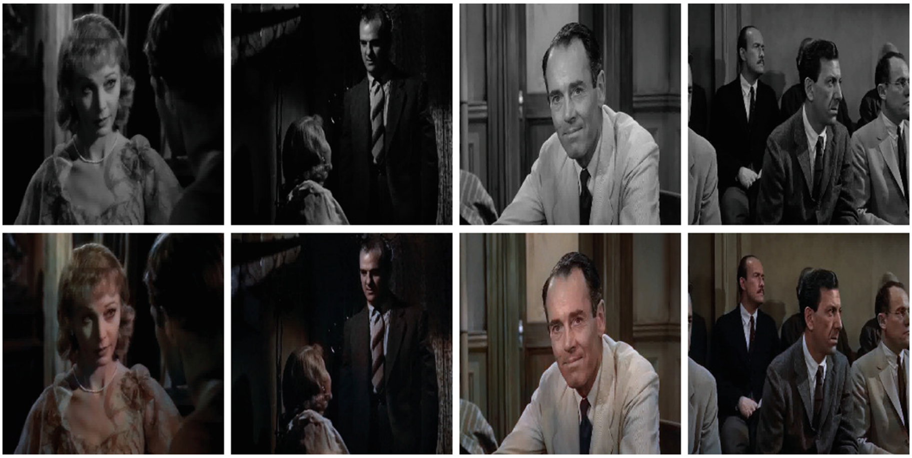

Fig. 4. Colorization results where color ground-truth is unavailable. Depicted are samples from "A streetcar named desire" and "Twelve angry men" (2 leftmost, 2 rightmost columns respectively), colorized with the proposed model trained on "Dial M for Murder".

trained models on one of the colored films to color black-and-white footage, i.e. in our case films (d) and (e). Results for this case can be examined at Fig. 4 (training performed on film (a)). Video colorization demos are available online[3].

5 Conclusion and Future Work

We have presented a method for automatic video colorization, based on a novel cGAN-based model with 3D convolutional and deconvolutional layers and an estimate aggregation scheme. The usefulness of our model has been validated with tests on colorizing old black-and-white film footage. Model performance has also been evaluated with single-image based metrics as well as a newly proposed metric that measures sequential color consistency. As future work, we envisage exploring the uses of the color prior in our aggregation scheme.

Acknowledgements. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan XP GPU used for this research.

References

1. Ben-Zrihem, N., Zelnik-Manor, L.: Approximate nearest neighbor fields in video. In: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5233–5242 (2015)

[3] http://www.cs.uoi.gr/~sfikas/video_colorization.

2. Chintala, S., Denton, E., Arjovsky, M., Mathieu, M.: How to train a GAN? Tips and tricks to make GANs work (2016). http://github.com/soumith/ganhacks. Accessed 25 January 2018
3. Daskalakis, C., Ilyas, A., Syrgkanis, V., Zeng, H.: Training GANs with optimism. CoRR abs/1711.00141 (2017). http://arxiv.org/abs/1711.00141
4. Goodfellow, I., et al.: Generative adversarial nets. In: Advances in Neural Information Processing Systems (NIPS), pp. 2672–2680 (2014)
5. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. arXiv preprint arXiv:1611.07004 (2016)
6. Juliani, A.: Pix2Pix-Film (2017). http://github.com/awjuliani/Pix2Pix-Film. Accessed 2 January 2018
7. Levin, A., Lischinski, D., Weiss, Y.: Colorization using optimization. ACM Trans. Graph. (TOG) **23**, 689–694 (2004)
8. Meyer, S., Cornillère, V., Djelouah, A., Schroers, C., Gross, M.: Deep video color propagation. arXiv preprint arXiv:1808.03232 (2018)
9. Otani, M., Hioki, H.: Video colorization based on optical flow and edge-oriented color propagation. In: Computational Imaging XII. vol. 9020, p. 902002. International Society for Optics and Photonics (2014)
10. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
11. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training GANs. In: Advances in neural information processing systems (NIPS), pp. 2234–2242 (2016)
12. Sheng, B., Sun, H., Magnor, M., Li, P.: Video colorization using parallel optimization in feature space. IEEE Trans. Circuits Syst. Video Technol. **24**(3), 407–417 (2014)
13. Veeravasarapu, V.R., Sivaswamy, J.: Fast and fully automated video colorization. In: 2012 International Conference on Signal Processing and Communications (SPCOM), pp. 1–5. IEEE (2012)
14. Welsh, T., Ashikhmin, M., Mueller, K.: Transferring color to greyscale images. ACM Trans. Graph. (TOG) **21**, 277–280 (2002)
15. Xia, S., Liu, J., Fang, Y., Yang, W., Guo, Z.: Robust and automatic video colorization via multiframe reordering refinement. In: IEEE International Conference on Image Processing, pp. 4017–4021. IEEE (2016)
16. Yatziv, L., Sapiro, G.: Fast image and video colorization using chrominance blending. IEEE Trans. Image Process. **15**(5), 1120–1129 (2006)
17. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9907, pp. 649–666. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46487-9_40
18. Dial M for murder. https://www.imdb.com/title/tt0046912/ (1954)
19. Et Dieu.créa la femme. https://www.imdb.com/title/tt0049189/ (1956)
20. Tzéni, tzéni. https://www.imdb.com/title/tt0145006/ (1966)
21. A streetcar named desire. https://www.imdb.com/title/tt0044081/ (1951)
22. Twelve angry men. https://www.imdb.com/title/tt0050083/ (1957)