

# Scene-adaptive Temporal Stabilisation for Video Colourisation using Deep Video Priors

Marc Gorriz Blanch<sup>1</sup>, Noel O'Connor<sup>2</sup>, and Marta Mrak<sup>1</sup>

<sup>1</sup> BBC Research & Development, London, UK

{marc.gorrizblanch, marta.mrak}@bbc.co.uk

<sup>2</sup> Dublin City University, Dublin, Ireland

noel.oconnor@dcu.ie

**Abstract.** Automatic image colourisation methods applied independently to each video frame usually lead to flickering artefacts or propagation of errors because of differences between neighbouring frames. While this can be partially solved using optical flow methods, complex scenarios such as the appearance of new objects in the scene limit the efficiency of such solutions. To address this issue, we propose application of blind temporal consistency, learned during the inference stage, to consistently adapt colourisation to the given frames. However, training at test time is extremely time-consuming and its performance is highly dependent on the content, motion, and length of the input video, requiring a large number of iterations to generalise to complex sequences with multiple shots and scene changes. This paper proposes a generalised framework for colourisation of complex videos with an optimised few-shot training strategy to learn scene-aware video priors. The proposed architecture is jointly trained to stabilise the input video and to cluster its frames with the aim of learning scene-specific modes. Experimental results show performance improvement in complex sequences while requiring less training data and significantly fewer iterations.

**Keywords:** Video colourisation, temporal consistency, deep video prior, few-shot learning

## 1 Introduction

Video restoration is in increasing demand in the production industry in order to both deliver historical content in high quality and to support innovation in the creative sector [24]. Video colourisation in particular is still a challenging task due to its ambiguity in the solution space and the requirement of global spatio-temporal consistency. Prior to automatic colourisation methods, producers relied on specialists to perform manual colourisation, resulting in a time consuming and sometimes a prohibitively expensive manual process. Researchers have thus endeavoured to develop computer-assisted methodologies in order to automate the colourisation process and reduce production costs. Early methods relied on frame-to-frame image colourisation techniques propagating colour scribbles [15,

20, 33] or reference colours [6, 28, 31]. The problem that typically occurs when processing is applied on a single frame without consideration of the neighbouring frames is temporal flickering. Similarly, propagation of errors can occur if the temporal dimension is not taken into account when characteristics (e.g colour) of previous frame are transferred to the current frame. Improved results can be obtained by considering a more robust propagation and imposing refinements with temporal constraints [1, 34].

Instead of improving temporal consistency using task-specific solutions, methods that generalise to various tasks can be applied. An example is the work in [5], which proposes a general approach agnostic to a specific image processing algorithm. The method takes the original video (black and white in the case of colourisation) and the per-frame processed counterpart (initially colourised version) and solves a gradient domain optimisation problem to minimise the temporal warping error between consecutive frames. An extension of such an approach takes into account object occlusions by leveraging information from a set of key-frames [32]. Another example was proposed in [17], adopting a perceptual loss to maintain perceptual similarity between output and processed frames. However, most methods rely on a dense correspondence backend (e.g. optical flow or PatchMatch [2]), which quickly becomes impractical in real-world scenarios due to the increased processing time needed. A novel solution proposed the use of Deep Video Prior by training a convolutional network on video content to enforce consistency between pairs of corresponding output patches [19]. The method solves multimodal consistency by means of Iteratively Reweighted Training, which learns to select a main mode among multiple inconsistent ones and discard those outliers leading to flickering artifacts. The main limitation is the requirement to train in test time, which makes the method extremely time-consuming in practice. For instance, training depends on the content, motion and length of the input video, requiring a large number of iterations to generalise to complex sequences with multiple shots and scene changes.

This paper proposes a framework for temporal stabilisation of frame-to-frame colourised videos with an optimised few-shot training strategy to learn scene-aware video priors. The proposed architecture is jointly trained to stabilise the input video and to cluster the input frames with the aim of learning scene-specific modes. Learnt embeddings are posteriorly injected into the decoder process to guide the stabilisation of specific scenes. A clustering algorithm for scene segmentation is used to select meaningful frames and to generate pseudo-labels to supervise the scene-aware training. Experimental results demonstrate the generalisation of the Deep Video Prior baseline [19], obtaining improved performance in complex sequences with small amounts of training data and fewer iterations.

## 2 Related work

### 2.1 Video colourisation

Although several works attempted to solve video colourisation problem as an end-to-end fully automatic task [18], most rely on single frame colourisation. This

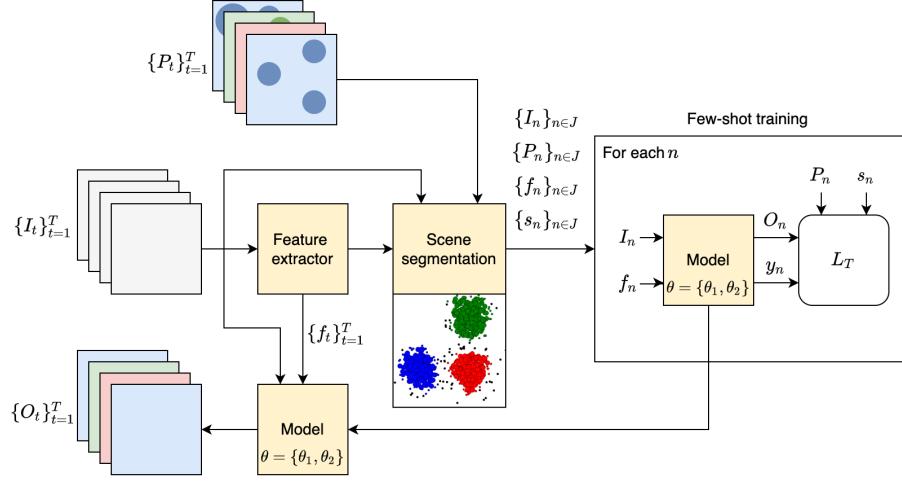
is because image colourisation, compared to video colourisation, achieves higher visual quality and naturalness. Propagation methods are commonly used to stabilise the temporal coherence between frames. For instance, the work in [16] propose Video Propagation Networks (VPN) to process video frames in an adaptive manner. VPN approach applies a neural network for adaptative spatio-temporal filtering. First it connects all the pixels from current and previous frames and propagates associated information across the sequence. Then it uses a spatial network to refine the generated features. Another example is the Switchable Temporal Propagation Network [21], based on a Temporal Propagation Network (TPN), which models the transition-related affinity between a pair of frames in a purely data-driven manner. In this way, a learnable unified framework for propagating a variety of visual properties from video frames, including colour, can be achieved. Aiming at improving the efficiency of deep video processing, colourisation and propagation can be performed at once. An example is the method in [34] that is based on a recurrent video colourisation framework, which combines colourisation and propagation sub-networks to jointly predict and refine results from a previous frame. A direct improvement is the method in [1] that uses masks as temporal correspondences and hence improves the colour leakage between objects by wrapping colours within restricted masked regions over time.

## 2.2 Deep Video Prior

Methods for temporal stabilisation usually promote blind temporal consistency by means of dense matching (optical flow or PatchMatch [2]) to define a regularisation loss that minimises the distance between correspondences in the stabilised output frames [5]. Such methods are trained with large datasets with pairs of grayscale inputs and coloured frames. Notice that such frameworks are blind to the image processing operator and can be used for multiple tasks such as super-resolution, denoising, dehazing, etc. In contrast, Deep Video Prior (DVP) can implicitly achieve such regularisation by training a convolutional neural network [19]. Such method only requires training on the single test video, and no training dataset is needed. To address the challenging multimodal inconsistency problem, an Iteratively Reweighted Training (IRT) strategy is used in DVP approach. The method selects one mode from multiple possible modes for the processed video to ensure temporal consistency and preserve perceptual quality.

## 2.3 Few-shot learning

Few-shot learning was introduced to learn from a limited number of examples with supervised information [10, 11]. For example, although current methods on image classification outperform humans on ImageNet [8], each class needs sufficient amount of labelled images, which can be difficult to obtain. Therefore, few-shot learning can reduce the data gathering effort for data-intensive applications [30]. Many related topics use this methodology, such as meta-learning [12, 25], embedding learning [3, 27] and generative modelling [9, 10]. The method



**Fig. 1.** Proposed framework for temporal stabilisation of frame-to-frame colourised videos. In addition to the DVP baseline, a scene segmentation and a few show training is used to learn scene-aware video priors.

proposed in this paper uses few-shot learning as training strategy to reduce processing time and to generalise to long and complex video sequences.

### 3 Method

This section describes the proposed extension of DVP baseline for multiple scenes, followed by the optimised few-shot training strategy which enables reduced processing time by removing the time response of DVP conditioned to the number of input frames. Finally, DVP architecture is modified by adding a classification sub-network which clusters the input frames with the objective of learning scene-specific priors.

#### 3.1 Extension of DVP to multiple scenes

Given a grayscale input sequence  $\{I_t\}_{t=1}^T$  of  $T$  frames and its colourised counterpart  $\{P_t\}_{t=1}^T$  created using an image colourisation operator  $F$ , the goal is to learn the mapping  $\hat{G}(\theta) : \{P_t\}_{t=1}^T \rightarrow \{O_t\}_{t=1}^T$ , such that  $\{O_t\}_{t=1}^T$  is a temporally stable output without flickering artifacts and  $\theta$  are the network parameters. Due to the superior performance of image colourisation compared to video methods [5, 18], an image operator is applied frame-to-frame and the proposed framework is used to sort out temporal issues. Therefore, from a random initialisation,  $\hat{G}(\theta)$  is optimised in each iteration by means of the reconstruction loss  $L_{data}$  (e.g.  $L_1$  distance) between  $\hat{G}(I_t; \theta)$  and  $P_t$ :

$$\arg \min_{\theta} L_{data}(\hat{G}(I_t; \theta), P_t). \quad (1)$$

As shown in Figure 1, the proposed method extends the DVP framework [19] for video sequence with multiple scenes. In particular, the proposed method defines a scene as a change of content, e.g. a camera shot, appearance of new objects, etc. In particular, the input sequence  $\{I_t\}_{t=1}^T$  of  $T$  frames is divided into  $S$  scenes, where typically  $S \ll T$ , and  $\{s_t\}_{t=1}^T$  is the scene index for each frame. In order to learn scene-specific modes, the proposed network not only learns to stabilise the input sequence, but also to cluster its frames into different scenes by generating a class distribution vector  $y_t \in \mathbb{R}^S$ . As shown in Figure 2, an external feature vector  $f_t$  (from frame  $I_t$ ) is provided in order to guide the clustering process.  $f_t$  can be obtained from a suitable neural network, e.g. from VGG-16 classification head [26]. Finally,  $y_t$  is used to generate scene-specific priors which are posteriorly injected into the different stages of the network decoder. Therefore, the proposed model combines two different sub-models, denoted by  $\hat{G}(\theta) = \{\hat{G}_1(\theta_1), \hat{G}_2(\theta_2)\}$ , where  $\theta = \{\theta_1, \theta_2\}$  are all the network parameters,  $\hat{G}_1(\theta_1) : \{P_t\}_{t=1}^T \rightarrow \{O_t\}_{t=1}^T$  and  $\hat{G}_2(\theta_2) : \{f_t\}_{t=1}^T \rightarrow \{y_t\}_{t=1}^T$ .

The neural network is then trained to jointly improve the temporal consistency of the input video frames  $\{I_t\}_{t=1}^T$  (enforcing  $\{O_t\}_{t=1}^T$  to be close to  $\{P_t\}_{t=1}^T$ ) and classify them into the corresponding scenes  $\{s_t\}_{t=1}^T$ . Following DVP baseline, an IRT strategy is used to address the problem of averaging when the difference of multiple modes is large (e.g. pixel with more than one possible colourisation solution). In particular, a confidence map  $C_t$  is used to enforce the selection of a main mode per pixel from multiple modes, while it ignores the outliers (minor modes leading to flickering artifacts). In practice, DVP doubles the number of output channels (e.g. 6 channels for RGB images) to obtain two output versions: a main frame  $O_t^{main}$  and an outlier frame  $O_t^{minor}$ . The confidence map  $C_{t,i}$  at iteration  $i$  is calculated by:

$$C_{t,i} = \begin{cases} 1 & d(O_{t,i}^{main}, P_t) < \max\{L_1(O_{t,i}^{minor}, P_t), \delta\} \\ 0 & \text{otherwise} \end{cases}, \quad (2)$$

where  $d$  is the function to measure the distance between pixels and  $\delta$  is a threshold. Therefore, the model parameters at iteration  $(i+1)$  can be optimised using  $C_{t,i}$  which guides the training loss:

$$\begin{aligned} \theta^{i+1} = \arg \min_{\theta} & \{L_{data}(C_{t,i} \odot O_{t,i}^{main}, C_{t,i} \odot P_t) + \\ & + L_{data}((1 - C_{t,i}) \odot O_{t,i}^{minor}, (1 - C_{t,i}) \odot P_t)\}. \end{aligned} \quad (3)$$

Then, a multi-loss function is proposed combining the IRT loss  $L_{IRT}$  between  $\hat{G}_1(I_t; \theta_1)$  and  $P_t$ , and the cross-entropy loss  $L_{class}$  between  $\hat{G}_2(f_t; \theta_2)$  and  $s_t$ :

$$L_T = L_{IRT} \left( \hat{G}_1(I_t; \theta_1), P_t \right) + L_{class} \left( \hat{G}_2(f_t; \theta_2), s_t \right). \quad (4)$$

### 3.2 Few-shot training strategy

The main limitation of DVP is the long processing time due to the need for training at inference time. This fact makes the method impractical for long

sequences. This paper proposes to speed up the training process reducing the number of iterations by means of a few-shot training strategy. Such strategy selects a reduced set of  $N$  frames  $\{I_n\}_{n \in J} \subset \{I_t\}_{t=1}^T$ , where  $J \subset \{1, 2, \dots, T\}$  and  $N < T$ . Notice that for completeness  $I_n \neq I_t$ . Selected few-shot samples are then used to train the model for generalisation to the remaining frames during inference time. The proposed model makes this solution feasible thanks to its scene-aware capacity to generalise to variable content. Such approach makes the model more robust for processing of sequences with changes (e.g. with high motion) as it temporally downsamples the input.

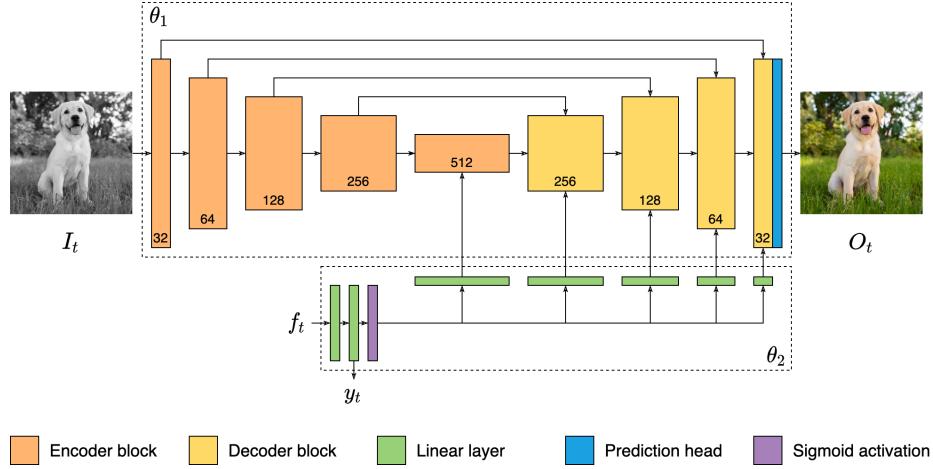
The selection of  $N$  frames for few-shot training is based on a twofold process: scene segmentation and selection of representative frames per scene. Scene segmentation is performed in an unsupervised way via clustering of deep features  $\{f_t\}_{t=1}^T$  with KMeans algorithm [22]. Dimensionality reduction is performed by Principal Components Analysis (PCA) in order to reduce complexity and shorten the clustering time. The number of scenes (e.g. number of clusters) is unknown and variable for each input video. Hence a suitable number of clusters is computed by running KMeans  $K$  times and selecting the elbow of the averaged distortion curve, where the distortion of each sample is computed relative the centroid of its cluster. This method allows a fast and effective scene segmentation approach.

Unsupervised clustering of input frames allows the generation of pseudo-labels for training the proposed classification sub-model. Notice that clustering errors will be mitigated thanks to the few-shot training, since the trained classifier will generalise to unseen frames (and potential uncertainties between scenes) during inference time. After segmentation of the input video into the scenes, suitable frames are selected from each scene by sub-clustering frames in that scene to cover a balanced span of different content. KMeans is applied again with a fixed number of clusters and a number of frames is randomly sampled from each sub-cluster. The number of selected frames per cluster and sub-clusters is proportional to the total number of frames in the given sub-cluster.

### 3.3 Network architecture

As shown in Figure 2, the architecture of the model proposed at Section 3.1 is composed of two sub-networks (denoted by  $\hat{G}_1(\theta_1)$ ,  $\hat{G}_2(\theta_2)$ ). Its inputs are a frame  $I_t \in \mathbb{R}^{1 \times H \times W}$ , where  $H \times W$  are the input dimensions, and its feature vector  $f_t \in \mathbb{R}^{1 \times d}$  (from VGG-16 classification head), where  $d$  are the number of its dimensions. The proposed architecture outputs two colour stabilised versions (main and minor frames)  $O_t \in \mathbb{R}^{6 \times H \times W}$  of the input frame, and a class distribution vector  $y_t \in \mathbb{R}^{1 \times S}$ , which is the product of clustering the input to a particular scene.

$I_t$  is processed by 4 encoder blocks which downsample the input by a factor of 2, generating  $I_t^b \in \mathbb{R}^{e_b \times H_b \times W_b}$ , where  $b = 1, \dots, 4$  is the block index,  $e_b$  is the number of dimensions and  $\{H_b, W_b\} = \max\left(2^5, \frac{\{H, W\}}{2^b}\right)$ . The bottleneck block converts  $I_t^4$  into  $O_t^5 \in \mathbb{R}^{o_5 \times H_5 \times W_5}$ , where  $o_5$  are the number of output



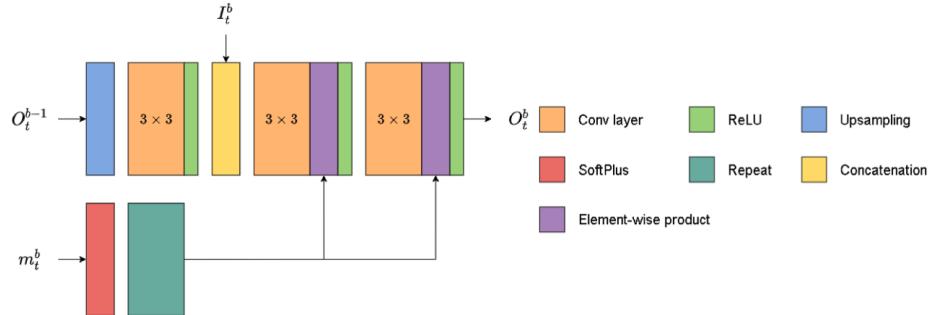
**Fig. 2.** Proposed architecture for stabilisation of frame-to-frame colourised videos. The model not only learns to stabilise an input sequence, but also to cluster the input frames into different scenes, by generating a class distribution vector  $y_t$ .

dimensions. In parallel,  $f_t$  is processed by 2 linear layers to generate deep embeddings  $f_t^1 \in \mathbb{R}^{1 \times d}$ ,  $f_t^2 \in \mathbb{R}^{1 \times S}$ .  $f_t^2$  is both activated with a softmax operation to generate the class distribution vector  $y_t$  and with a sigmoid operation to generate the scene-aware mask  $a_t$  that will be injected into the bottleneck and decoder blocks.  $a_t$  is processed by a sequence of linear layers which generate 5 scene-aware embeddings  $m_t^b \in \mathbb{R}^{1 \times o_b}$ , where  $b = 1, \dots, 5$  and  $o_b$  are the dimensions of the bottleneck and decoder outputs. Finally, as shown in Figure 3,  $m_t^b$  are injected into the corresponding blocks as follows: (1)  $m_t^b$  is activated with a SoftPlus operation (smooth approximation of ReLU) and spatially repeated to generate a volume  $M_t^b \in \mathbb{R}^{e_b \times H_b \times W_b}$ , (2)  $M_t^b$  is element-wise multiplied to each pre-activation within the corresponding block. 4 decoder blocks with skip connections are then applied to upsample the inputs by factor of 2, generating  $O_t^b \in \mathbb{R}^{o_b \times H_b \times W_b}$ , where  $b = 1, \dots, 4$ . Finally, a decoder head is applied to map  $O_t^1$  into the output frames  $O_t$ .

## 4 Experiments

### 4.1 Training strategy

As shown in DVP, the network needs to be initialised with the main mode in order to guide the main outputs towards a specific mode. DVP selects the first image as reference for the main mode and pre-trains the network for a given number of iterations. However, when the reference image contains outliers, and those are treated as main mode, the performance of such approach is not satisfactory. To address that, this work proposes the use of colour histograms to



**Fig. 3.** The proposed decoder block conditioned by the scene-wise embedding  $m_t^b$ . Notice that a similar architecture applies to the bottleneck, injecting the embedding vector into the pre-activations.

detect outliers when specific bins present high variance across the sequence. In particular, colourised frames  $P_t$  are converted into CIE La\*b\* colour space [7], and 2D colour histograms  $\mathcal{H}_t \in \mathbb{R}^{Q \times Q}$  are obtained by matrix multiplication of individual histograms for a\* and b\* channels, where  $Q$  is the number of bins. Next, a mask  $M \in \mathbb{R}^{Q \times Q}$  is computed to locate those bins present in all the frames. Hence, bins out of the mask will represent an outlier.  $\bar{M} = \prod_{t=1}^T M_t$ , where  $M_t$  masks the bins different than zero. Finally, main mode reference frame  $P_{t^*}$  is obtained, where  $t^* = \arg \min_t \sum \mathcal{H}_t \odot (M_t - \bar{M})$ .

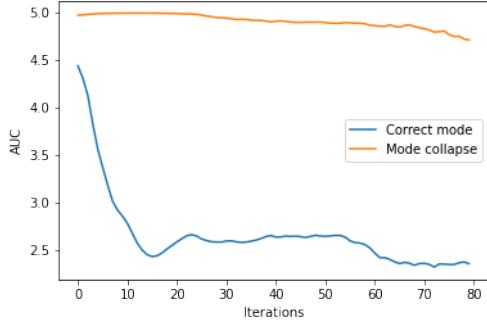
On the other hand, as shown in Figure 4, few-shot training might lead the network into mode collapse, rapidly converging into a random state. Mode collapse is detected when the Area Under the Curve (AUC) of the generated colour histograms vary below a threshold during a given number of iterations. In this case, the initial pre-training is repeated with random initialisation of the network weights. Due to the significant difference of complexity, classifier and stabiliser (U-Net) sub-networks are optimised using different learning rates. Overall, Adam optimiser is adopted, using a learning rate of  $10^{-4}$  for  $\theta_1$  and  $10^{-6}$  for  $\theta_2$ . All the experiments are performed with a single GPU and using a batch size of 8 samples. Initial pre-training iterations are set to 350, and 150 frames are used for few-shot training.

Following DVP [19], this work uses the test set collected by [5], composed by 20 videos of around 200 frames from Videvo dataset<sup>3</sup>, and extended with 8 longer videos from Videvo and Hollywood2 dataset [23], to evaluate the performance for more complex content.

## 4.2 Evaluation metrics

**Temporal inconsistency.** DVP uses wrapping error to measure temporal inconsistency by means of optical flow. However, the quality of optical flow computation and the corresponding occlusion mask might decrease when dealing with

<sup>3</sup> <https://www.videvo.net/>



**Fig. 4.** Example of mode collapse during initial pre-training. The AUC invariance during the initial iterations indicates convergence to a random state, which affects the performance of the posterior IRT training.

flickering content. To mitigate this issue and to better capture colour artifacts, histogram inconsistency is adopted to measure the temporal similarity in the colour domain. Being  $\mathcal{H}_t$  and  $\mathcal{H}_{t-1}$  the colour histograms of frames  $t$  and  $t-1$ , respectively, temporal histogram inconsistency  $E_{hist}$  is defined as a symmetric  $\chi^2$  distance as follows:

$$E_{hist} = 2 \sum_{q=1}^{Q^2} \frac{(\mathcal{H}_{t,q} - \mathcal{H}_{t-1,q})^2}{(\mathcal{H}_{t,q} + \mathcal{H}_{t-1,q}) + \epsilon}, \quad (5)$$

where  $\epsilon$  prevents infinity overflows and  $Q$  is the number of bins.

**Performance degradation.** Temporal stabilisation has to be achieved without degrading the original colourisation. Since stabilised ground truth is not available, this work uses data fidelity  $F_{data}$  between  $\{O_t\}_{t=1}^T$  and  $\{P_t\}_{t=1}^T$  as follows:

$$F_{data} = \frac{1}{T} \sum_{t=1}^T PSNR(P_t, O_t). \quad (6)$$

Notice that data fidelity can decrease when frames contain large amount of outliers. Therefore, perceptual quality is also evaluated using Fréchet Inception Distance (FID) [14] with the ground truth.

### 4.3 Results

Table 1 shows quantitative comparison results between DVP method [19], our method and the proposed ablations in Section 4.4. Two image-based fully-automatic colourisation methods are considered: colourful image colourisation (CIC) [35] and ChromaGAN (CGAN) [29]. Reference-based image colourisation method



**Fig. 5.** Qualitative comparison with DVP method and the proposed ablations. DVP with few-shot training took around the same processing time as our approach, but failed to generalise to multiple scenes. Moreover, our method achieved better fidelity and perceptual quality than the original DVP.

XCNET [4] is also considered. Such methods which colourise frames based on a reference image introduce even larger flickering issues than fully auto-colourisation based networks. References are sampled from Imagenet dataset [8] using the correspondence recommendation pipeline proposed in [4, 13]. Finally, quality of the original predictions  $\{P_t\}_{t=1}^T$  obtained using CIC method is studied to evaluate the effect of the proposed stabilisation. Moreover, Figure 6 shows the processing time of both DVP and our method in relation to the number of frames.

As can be seen from  $E_{hist}$  results, both DVP and our method significantly increase the temporal consistency compared to the original predictions, and although DVP obtains slightly better results, our method significantly reduces the processing time for long scenes. The drop in performance when using XCNET is due to the colourfulness of the corresponding predictions and the higher concentration of flickering artefacts, compared to CIC or CGAN.

As shown in Figure 5, the frames at different times in the same shot suffer from inconsistent colourisation (notice the same object across various frames with different colour). DVP and DVP with few-shot training temporal both provide more consistent results, but still the main mode is either not correctly chosen or the colours are plain, resulting in less natural appearance. This is reflected in data fidelity results, where our method achieves the best performance. FID also confirm this fact, as DVP lowers the perceptual quality of the original predictions due to its strong stabilisation and degradation of input colours. Finally, as shown in Figure 6, the few-shot strategy allowed a fix amount of

**Table 1.**  $E_{hist}$ ,  $F_{data}$  and FID comparison for different colourisation methods.

Method	$E_{hist} \downarrow$			
	$\{P_t\}_{t=1}^T$	CIC [35]	CGAN [29]	XCNET [4]
DVP [19]		2.30	1.58	3.30
Ours	20.96	3.08	2.54	3.59
DVP (few-shot)		3.75	3.79	3.10
Ours (first frame)		1.39	2.14	2.69

Method	$F_{data} \uparrow [\text{dB}]$		
	CIC [35]	CGAN [29]	XCNET [4]
DVP [19]	19.12	19.32	18.94
Ours	28.63	30.31	26.56
DVP (few-shot)	18.14	18.47	18.67
Ours (first frame)	28.46	30.21	26.40

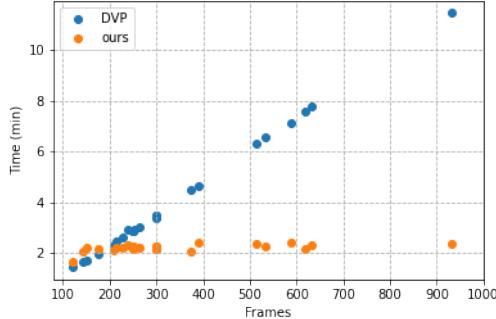
  

Method	$FID \downarrow$			
	$\{P_t\}_{t=1}^T$	CIC [35]	CGAN [29]	XCNET [4]
DVP [19]		126.38	111.16	100.21
Ours	122.74	121.16	105.65	97.96
DVP (few-shot)		129.68	114.98	102.22
Ours (first frame)		119.76	104.03	99.92

training iterations, resulting into a flat time response independent to the length of the input sequence. Note that the total time may increase proportionally to the number of scenes, due to the individual initial pre-training per scene.

#### 4.4 Ablations

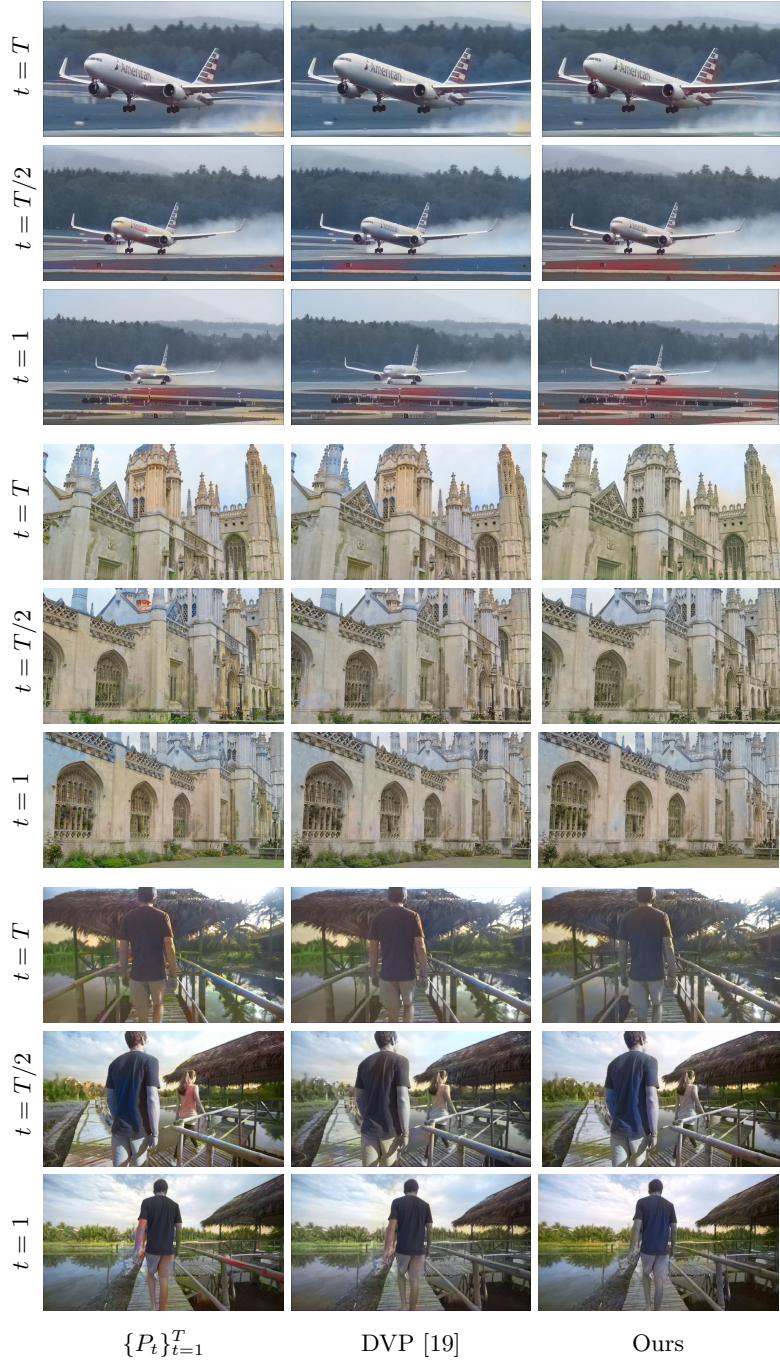
An ablation study is performed to analyse the importance of the proposed scene-aware architecture. First, DVP is tested with the proposed few-shot training strategy. As shown in Table 1 and Figure 5 (DVP few-shot), without using a classification sub-network, DVP is unable to generalise to complex sequences and the input colours are significantly degraded. This drop in performance proves the importance of the classification sub-network to perform effective few-shot training. Finally, a second ablation is performed to evaluate the proposed initialisation mechanism in Section 4.1, which proposes the best reference for main mode per scene by means of histogram characteristics. As shown in Table 1 (ours first frame), a drop in performance is observed when using the first frame as main mode reference (as DVP proposes), proving the effectiveness of the proposed methodology. Notice that original DVP performance could be improved by using the same initialisation mechanism.



**Fig. 6.** Comparison of processing time for all test sequences. Notice the significant increase of the processing time for DVP when the number of frames increases.

## 5 Conclusions

This paper proposed a general framework for temporal stabilisation of frame-to-frame colourised videos using scene-aware deep video priors. The framework includes an optimised few-shot training strategy to reduce the processing time of DVP baseline [19] by removing its time response conditioned on the number of input frames. In order to handle complex sequences with multiple scenes, the DVP architecture is modified by adding a classification sub-network which clusters the input frames with the objective of learning scene-specific priors. Experimental results show that our method improves data fidelity and perceptual quality and achieves similar temporal consistency to DVP while reducing the processing time in long sequences. As future work, model efficiency can be further improved by simplifying the network architecture or by using techniques such as pruning or weights quantisation. Moreover, finer tuning of colourisation could be achieved by improving the scene segmentation process in order to obtain more precise scene priors. Finally, an unified framework for video colourisation can be obtained by integrating the deep video prior methodology into an end-to-end video colourisation pipeline.



**Fig. 7.** Evaluation in comparison with DVP method and processed frames at different timestamps.



**Fig. 8.** Evaluation in comparison with DVP method and processed frames at different timestamps.

## References

1. Akimoto, N., Hayakawa, A., Shin, A., Narihira, T.: Reference-based video colorization with spatiotemporal correspondence. arXiv preprint arXiv:2011.12528 (2020)
2. Barnes, C., Shechtman, E., Finkelstein, A., Goldman, D.B.: Patchmatch: A randomized correspondence algorithm for structural image editing. ACM Trans. Graph. **28**(3), 24 (2009)
3. Bertinetto, L., Henriques, J.F., Valmadre, J., Torr, P., Vedaldi, A.: Learning feed-forward one-shot learners. Advances in neural information processing systems **29** (2016)
4. Blanch, M.G., Khalifeh, I., O'Connor, N.E., Mrak, M.: Attention-based stylisation for exemplar image colourisation. In: 2021 IEEE 23rd International Workshop on Multimedia Signal Processing (MMSP). pp. 1–6 (2021). <https://doi.org/10.1109/MMSP53017.2021.9733506>
5. Bonneel, N., Tompkin, J., Sunkavalli, K., Sun, D., Paris, S., Pfister, H.: Blind video temporal consistency. ACM Transactions on Graphics (TOG) **34**(6), 1–9 (2015)
6. Bugeau, A., Ta, V.T., Papadakis, N.: Variational exemplar-based image colorization. IEEE Transactions on Image Processing **23**(1), 298–307 (2013)
7. Connolly, C., Fleiss, T.: A study of efficiency and accuracy in the transformation from rgb to cielab color space. IEEE transactions on image processing **6**(7), 1046–1048 (1997)
8. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255 (2009). <https://doi.org/10.1109/CVPR.2009.5206848>
9. Edwards, H., Storkey, A.: Towards a neural statistician. arXiv preprint arXiv:1606.02185 (2016)
10. Fei-Fei, L., Fergus, R., Perona, P.: One-shot learning of object categories. IEEE transactions on pattern analysis and machine intelligence **28**(4), 594–611 (2006)
11. Fink, M.: Object classification from a single example utilizing class relevance metrics. Advances in neural information processing systems **17** (2004)
12. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: International conference on machine learning. pp. 1126–1135. PMLR (2017)
13. He, M., Chen, D., Liao, J., Sander, P.V., Yuan, L.: Deep exemplar-based colorization. ACM Transactions on Graphics (TOG) **37**(4), 1–16 (2018)
14. 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. Advances in neural information processing systems **30** (2017)
15. Huang, Y.C., Tung, Y.S., Chen, J.C., Wang, S.W., Wu, J.L.: An adaptive edge detection based colorization algorithm and its applications. In: Proceedings of the 13th annual ACM international conference on Multimedia. pp. 351–354 (2005)
16. Jampani, V., Gadde, R., Gehler, P.V.: Video propagation networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 451–461 (2017)
17. Lai, W.S., Huang, J.B., Wang, O., Shechtman, E., Yumer, E., Yang, M.H.: Learning blind video temporal consistency. In: Proceedings of the European conference on computer vision (ECCV). pp. 170–185 (2018)
18. Lei, C., Chen, Q.: Fully automatic video colorization with self-regularization and diversity. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3753–3761 (2019)

19. Lei, C., Xing, Y., Chen, Q.: Blind video temporal consistency via deep video prior. *Advances in Neural Information Processing Systems* **33**, 1083–1093 (2020)
20. Levin, A., Lischinski, D., Weiss, Y.: Colorization using optimization. In: ACM SIGGRAPH 2004 Papers, pp. 689–694 (2004)
21. Liu, S., Zhong, G., De Mello, S., Gu, J., Jampani, V., Yang, M.H., Kautz, J.: Switchable temporal propagation network. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 87–102 (2018)
22. MacQueen, J., et al.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. vol. 1, pp. 281–297. Oakland, CA, USA (1967)
23. Marszalek, M., Laptev, I., Schmid, C.: Actions in context. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. pp. 2929–2936. IEEE (2009)
24. Mrak, M.: Ai gets creative. In: Proceedings of the 1st International Workshop on AI for Smart TV Content Production, Access and Delivery. p. 1–2. AI4TV ’19, Association for Computing Machinery, New York, NY, USA (2019). <https://doi.org/10.1145/3347449.3357490>
25. Ravi, S., Larochelle, H.: Optimization as a model for few-shot learning (2016)
26. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
27. Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P.H., Hospedales, T.M.: Learning to compare: Relation network for few-shot learning. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1199–1208 (2018)
28. Tai, Y.W., Jia, J., Tang, C.K.: Local color transfer via probabilistic segmentation by expectation-maximization. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05). vol. 1, pp. 747–754. IEEE (2005)
29. Vitoria, P., Raad, L., Ballester, C.: Chromagan: Adversarial picture colorization with semantic class distribution. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 2445–2454 (2020)
30. Wang, Y., Yao, Q., Kwok, J.T., Ni, L.M.: Generalizing from a few examples: A survey on few-shot learning. *ACM computing surveys (csur)* **53**(3), 1–34 (2020)
31. Welsh, T., Ashikhmin, M., Mueller, K.: Transferring color to greyscale images. In: Proceedings of the 29th annual conference on Computer graphics and interactive techniques. pp. 277–280 (2002)
32. Yao, C.H., Chang, C.Y., Chien, S.Y.: Occlusion-aware video temporal consistency. In: Proceedings of the 25th ACM international conference on Multimedia. pp. 777–785 (2017)
33. Yatziv, L., Sapiro, G.: Fast image and video colorization using chrominance blending. *IEEE transactions on image processing* **15**(5), 1120–1129 (2006)
34. Zhang, B., He, M., Liao, J., Sander, P.V., Yuan, L., Bermak, A., Chen, D.: Deep exemplar-based video colorization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8052–8061 (2019)
35. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: European conference on computer vision. pp. 649–666. Springer (2016)