DeepRemaster: Temporal Source-Reference Attention Networks for Comprehensive Video Enhancement

SATOSHI IIZUKA, University of Tsukuba, Japan  
EDGAR SIMO-SERRA, Waseda University / JST PRESTO, Japan

Fig. 1. Vintage film remastering results. Our approach is able to remaster 700 frames of video using only 6 reference color images in a single processing step. The first row shows various frames from the input video, the second row shows the restored black and white frames, the third row shows the variation between the input and restored black and white frames, and the fourth row shows the final colorized output. We show the reference color images used on the right. Using source-reference attention, our model automatically matches similar regions to the reference color images, and using self-attention with temporal convolutions it is able to enforce temporal consistency. Our approach is able to restore the noisy and blurring input, and, afterwards, with the few manually colored reference images, we are able to obtain a temporally-consistent natural looking color video. Images are taken from “A-Bomb Blast Effects” (1952) and licensed under the public domain. Figure best viewed in color.

The remastering of vintage film comprises of a diversity of sub-tasks including super-resolution, noise removal, and contrast enhancement which aim to restore the deteriorated film medium to its original state. Additionally, due to the technical limitations of the time, most vintage film is either recorded in black and white, or has low quality colors, for which colorization becomes necessary. In this work, we propose a single framework to tackle the entire remastering task semi-interactively. Our work is based on temporal convolutional neural networks with attention mechanisms trained on videos with data-driven deterioration simulation. Our proposed source-reference attention allows the model to handle an arbitrary number of reference color images to colorize long videos without the need for segmentation while maintaining temporal consistency. Quantitative analysis shows that our framework outperforms existing approaches, and that, in contrast to existing approaches, the performance of our framework increases with longer videos and more reference color images.

CCS Concepts: • Computing methodologies → Image processing; Neural networks.

Additional Key Words and Phrases: remastering, restoration, colorization, convolutional network, source-reference attention

ACM Reference Format:
Satoshi Iizuka and Edgar Simo-Serra. 2019. DeepRemaster: Temporal Source-Reference Attention Networks for Comprehensive Video Enhancement. ACM Trans. Graph. 38, 6, Article 176 (November 2019), 13 pages. https://doi.org/10.1145/3355089.3356570
1 INTRODUCTION

Since the invention of motion pictures in the late 19th century, an incredible amount of hours of film have been recorded and released. However, in addition to visual artifacts and the low quality of the film technology at the time, many of the earlier works of significant historical value have suffered from degradation or been lost. Restoration of such important films, given their analogue nature, is complicated, with the initial efforts beginning on restoring the film at a physical level. Afterwards, the content is transferred to the digital medium, where it is remastered by removing noise and artifacts in addition to adding color to the film frames. However, such remastering processes require a significant amount of both time and money, and is currently done manually by experts with a single film costing in the order of hundreds of thousands to millions of dollars. Under these circumstances, huge industries such as publishers, TV, and the print industry, which own an enormous quantity of archived deteriorated old videos, show a great demand for efficient remastering techniques. In this work, we propose a semi-automatic approach for remastering old black and white films that have been converted to digital data.

Remastering an old film is not as simple as using a noise removal algorithm followed by colorization approach in a pipeline fashion: the noise and colorization processes are intertwined and affect each other. Furthermore, most old films suffer from blurring and low resolution, for which increasing the sharpness also becomes important. We propose a full pipeline for remastering black and white motion pictures, made of several trainable components which we train in a single end-to-end framework. By using a careful data creation and augmentation scheme, we are able to train the model to remaster videos by not only removing noise and adding color, but also increasing the resolution and sharpness, and improving the contrast with temporal consistency.

Our approach is based on fully convolutional networks. In contrast to many recent works that use recursive models for processing videos [Liu et al. 2018; Vondrick et al. 2018], we use temporal convolutions that allow for processing video frames by taking account information from multiple frames of the input video at once. In addition, we propose using an attention mechanism, which we denote as source-reference attention, that allows using multiple reference frames in an interactive manner. In particular, we use this source-reference attention to provide the model with an arbitrary number of color mages to be used as references when adding color. The model is able to not only dynamically choose what reference frames to use when coloring each output frame, but also choose what regions of the reference frames to use for each output region in a computationally efficient manner. We show how this approach can be used to remaster long sequences comprising of multiple different scenes (close-up, panorama, etc.), using an assortment of reference frames as shown in Fig. 1. The number of reference frames used is not fixed and it is even possible to remaster in a fully automatic way by not providing reference frames. Additionally, by manually creating and/or colorizing reference frames, it is possible for the user to control the colorization results when remastering, which is necessary for practical applications.

We perform an in-depth evaluation of our approach both quantitatively and qualitatively, and find the results of our framework to be favorable in comparison with existing approaches. Furthermore, the performance of our approach increases on longer sequences with more reference color images, which proves to be a challenge for existing approaches. Our experiments show that using source-reference attention it is possible to remaster thousands of frames with a small set of reference images in a efficiently with stable and consistent colors.

To summarize, our contributions are as follows: (1) the first single framework for remastering vintage film, (2) source-reference attention that can handle an arbitrary number of reference images, (3) an example-based film degradation simulation approach for generating training data for film restoration, and (4) an in-depth evaluation with favorable results with respect to existing approaches and strong baselines. Models, code, and additional results are made available at http://iizuka.cs.tsukuba.ac.jp/projects/remastering/.

2 RELATED WORK

2.1 Denoising and Restoration

One of the more classical approaches to denoising and restoration is the family of Block-Matching and 3D filtering (BM3D) algorithms [Dabov et al. 2007; Maggioni et al. 2012, 2014], which are based on collaborative filtering in the in the transform domain. Although fairly limited in the types of noise patterns they can eliminate, these approaches have wide applicability to both images and video. Besides noise removal, other restoration related applications such as image super-resolution and deblurring [Danielyan et al. 2012] have also been explored with the BM3D algorithm.

More recently, Convolutional Neural Networks have been used for denoising-type applications, and, in particular, for single images [Lefkimmiatis 2018; Zhang et al. 2018b]. However, these generally assume simple additive Gaussian noise [Lefkimmiatis 2018], blurring [Fan et al. 2018; Shi et al. 2016; Yu et al. 2018], or JPEG-deblocking [Zhang et al. 2017b], or are applied to specialized tasks such as Monte Carlo rendering denoising [Bako et al. 2017; Chaitanya et al. 2017; Vogels et al. 2018] for which it is easy to create supervised training data. Extensions for video based on optical flow and transformer networks have also been proposed [Kim et al. 2018]. However, restoration of old film requires more than being able to remove Gaussian noise or blur: it requires being able to remove film artifacts that can be both local, affecting a small region of the image, or global, affecting the contrast and brightness of the entire frame, as shown in Fig. 2. For this it is necessary to create higher quality and realistic film noise as we propose in our approach.

2.2 Colorization

Colorization of black and white images is an ill-posed problem in which there is no single solution. Most approaches have relied on user inputs, either in the form of scribbles [Huang et al. 2005; Levin et al. 2004], reference images similar to the image being colorized [Irons et al. 2005; Pitié et al. 2007; Reinhard et al. 2001; Tai et al. 2005; Welsh et al. 2002; Wu et al. 2013], or internet queries [Chia et al. 2011; Liu et al. 2008]. While most traditional approaches have focused on solving an optimization problem using
(a) Synthetic noisy images. (b) Vintage black and white movies from the early 20th century.

Fig. 2. Comparison between denoising and restoration tasks. (a) Example of generated synthetic images for denoising tasks [Martin et al. 2001]. The top row shows the original images and the bottom row shows them with added Gaussian noise. (b) Example of vintage film which requires restoration. The old movies suffer from a plethora of deterioration issues such as film grain noise, scratches, dampness, vignetting, and contrast bleed, which make them challenging to restore to their original quality. (a) Images are taken from [Martin et al. 2001], and (b) videos licensed in the Public Domain.

both the input greyscale image and the user provided hints or references images [An and Pellacini 2008; Levin et al. 2004; Xu et al. 2013], recent approaches have opted to leverage large datasets and employ learning-based models such as Convolutional Neural Networks (CNN) to colorize images automatically [Iizuka et al. 2016; Larsson et al. 2016; Zhang et al. 2016]. Analogous to the optimization-based approaches, CNN-based approaches have been extended to handle user inputs as both scribbles [Sangkloy et al. 2017; Zhang et al. 2017a], and a single reference images [He et al. 2018; Meyer et al. 2018]. Our approach, while related to existing CNN-based methods, extends the colorization to video and an arbitrary number of reference images, in addition to performing restoration of the video.

Related to the current work are Recursive Neural Network (RNN) approaches for colorizing videos [Liu et al. 2018; Vondrick et al. 2018]. They process the video frame-by-frame by propagating the color from an initial colored key frame to rest of the scene. While this is a simple way to colorize videos, it can fail to propagate the color when there are abrupt changes in scene. In particular, RNN-based methods have the following limitations:

1. They require the first frame to be colored and cannot use related frames.
2. They are unable to propagate between scene changes, and thus require precise scene segmentation. This doesn’t allow handling scenes that alternate back and forth, as commonly done in movies, which end up requiring many additional colorized references.
3. Once they make an error they continue amplifying it. This severely limits the number of frames that can be propagated.

In contrast to RNN-based approaches, our approach is able to handle multiple scenes or entire videos seamlessly as shown in Fig. 3. Instead of using a RNN, we use a CNN with temporal convolutions and attention, which allows incorporating non-local information from multiple input frames to colorize a single output frame.

2.3 Attention

Attention mechanisms for neural networks were original developed for Natural Language Translation (NLT) [Bahdanau et al. 2015]. Similar to human attention, attention for neural network allows the model to focus on different parts of the input. For NLT, attention allows to find a mapping between the input language words and the output language words, which can be in different orders. For natural language processing, many different variants have been
3 APPROACH

Our approach is based on fully convolutional networks, which are a variant of convolutional neural networks in which only convolutional layers are employed. This allows processing images and videos of any resolution. We employ a mix of temporal and spatial convolution layers, in addition to attention-based mechanisms that allow us to use an arbitrary number of reference color images during the remastering. An overview of the proposed approach can be seen in Fig. 4.

3.1 Source-Reference Attention

We employ source-reference attention to be able to supply an arbitrary number of reference color images that the model can use as hints for the remastering of videos. In particular, source-reference attention layers take as an input two different variable length volumetric feature maps, one corresponding to the source data and the other to the reference data, and allow the model to exploit non-local similarities between the source data and the reference data. The model can thus use the color from the reference data to colorize similar areas of the source data.

More formally, let the source data feature representation be $h_s \in \mathbb{R}^{C \times T_s \times H_s \times W_s}$ with $C$ channels, $T_s$ frames of height $H_s$ and width
\( W_s \), and let the reference data features be \( h_r \in \mathbb{R}^{C_r \times N_r \times H_r \times W_r} \) with \( C_r \) channels, \( N_r \) maps of height \( H_r \) and width \( W_r \). The source-reference attention layer \( A_{sr}(\cdot, \cdot) \) can be defined as

\[ A_{sr}(h_s, h_r) = h_s + \gamma d(e_s(h_r) \text{ softmax}(e_r(h_r)\top e_s(h_r))) \]

(1)

where \( \gamma \in \mathbb{R} \) is a learnt parameter and

\[ e_s : \mathbb{R}^{C \times T_s \times H_s \times W_s} \to \mathbb{R}^{C \times T_s \times H_s \times W_s} \]
\[ e_r : \mathbb{R}^{C \times N_r \times H_r \times W_r} \to \mathbb{R}^{C \times N_r \times H_r \times W_r} \]
\[ e_t : \mathbb{R}^{C \times N_r \times H_r \times W_r} \to \mathbb{R}^{C \times N_r \times H_r \times W_r} \]

(2)

are encoding functions that map the input source and reference feature tensors to matrices with a reduced number of channels, and \( d : \mathbb{R}^{C \times T_s \times H_s \times W_s} \to \mathbb{R}^{C \times T_s \times H_s \times W_s} \) is a decoding function that simply reshapes the tensor without modifying any values. For the encoding functions, we use temporal convolution operators with \( 1 \times 1 \times 1 \) pixel kernels followed by reshaping to the appropriate output dimensions. A visual overview of the source-reference attention layer is shown in Fig. 5.

Note that if reference data features are not provided, the output of the source-reference attention layer becomes simply the source data features. Furthermore, in the case the same features are used for both the source and reference features, the source-reference attention layer becomes simply the source data encoding function. For self-attention, we use the source-reference model architecture is shown in Table 2.

### 3.2.1 Pre-Processing Network

The pre-processing network is formed exclusively by temporal convolution layers, and uses a skip connection between the input and output. The main objective of the pre-processing network is to remove artefacts and noise from the input greyscale video. The network uses an encoder-decoder architecture in which the resolution is halved twice and restored to the full size at the end with trilinear upsampling. A full overview of the pre-processing model architecture is shown in Table 1. Most of the processing is done at the low resolution to decrease the computational burden, and the output of this network is used as the luminance channel of the final output image.

#### Table 1. Overview of the pre-processing model architecture

| Layer Type | Output Resolution | Notes |
|------------|-------------------|-------|
| Input      | \( 1 \times T_s \times W_s \times H_s \) | Input greyscale image |
| TConv.     | \( 64 \times T_s \times W_s/2 \times H_s/2 \) | Replication padding, spatial stride of 2 |
| TConv. \((\times 2)\) | \( 128 \times T_s \times W_s/2 \times H_s/2 \) | |
| TConv.     | \( 256 \times T_s \times W_s/4 \times H_s/4 \) | Spatial stride of 2 |
| TConv. \((\times 4)\) | \( 256 \times T_s \times W_s/4 \times H_s/4 \) | |
| TConv.     | \( 128 \times T_s \times W_s/2 \times H_s/2 \) | Trilinear upsampling |
| TConv. \((\times 2)\) | \( 64 \times T_s \times W_s/2 \times H_s/2 \) | |
| TConv.     | \( 16 \times T_s \times W_s \times H_s \) | Trilinear upsampling |
| TConv.     | \( 1 \times T_s \times W_s \times H_s \) | TanH output, input is added, and finally clamped to \([0, 1]\) range |

3.2.2 Source-Reference Network

The source-reference network forms the core of the model and takes as an input the output of the pre-processing network along with an arbitrary number of user-provided reference color images. Two forms of attention are employed to allow non-local information to be used when computing the output chrominance maps: source-reference attention allows information from reference color images to be used, giving the user indirect control of the colorization; and self-attention allows non-local temporal information to be used, increasing the temporal consistency of the colorization. For self-attention, we use the source-reference attention layer implementation and use the same features for both the source and reference feature maps. An overview of the source-reference model architecture is shown in Table 2.

As with the pre-process network, the model is based on a encoder-decoder architecture, where the resolution is reduced to allow for more efficient computation and lower memory usage, and restored for the final output. While temporal convolutions allow for better temporal consistency, they also complicate the learning and increase the computational burden. Unlike the pre-processing network, the source-reference network uses a mix of temporal and spatial convolutions. In particular, the decoder and \( 1/8 \) middle branch use temporal convolutions while the encoders of both the input video and reference images use spatial convolutions, and the \( 1/16 \) middle branch uses a mixture of both, which we found decreases memory usage and simplifies the training, while not sacrificing any remastering accuracy. Furthermore, in the case of the reference color images, there is no temporal coherency to be exploited by using temporal convolutions as the images are not necessarily related.

First, the input video and reference images are separately reduced to \( 1/8 \) of the original width and height in three stages by separate...
encoders. The encoded input video and reference video features are then split into two branches: one processes the video at $1/8$ width and height, and one decreases the resolution another stage to $1/16$ of the original width and height to further process the video. Both branches employ source-reference attention layers, additional temporal convolution layers, and self-attention layers. In particular, the $1/16$ branch is processed with a self-attention layer before being upsampled trilinearly and concatenated to the $1/8$ branch output. The resulting combined features are processed with self-attention to be more temporally uniform. Afterwards, a decoder converts the features to chrominance channels in three stages using trilinear upsampling. Finally the output of the network is used as the image chrominance with two channels corresponding to the $ab$ channels of the Lab color-space, while the output of the pre-processing network is used as the image luminance corresponding to the $L$ channel.

4 TRAINING

We train our model using manually curated supervised training data. In order to improve both the generalization and quality of the results, we perform large amounts of both synthetic data augmentation and example-based film deterioration.

4.1 Objective Function

We train the model in a fully supervised fashion with a linear combination of two $L_1$ losses. In particular, we use a supervised dataset $D$ consisting of pairs of deteriorated black and white videos $x$ and restored color videos split into luminance $y_l$ and chrominance $y_{ab}$ using the Lab color-space, and reference color images $z$, and optimize the following expression:

$$\arg \min_{\theta, \phi} \mathbb{E}_{(x,y_l,y_{ab},z) \in D} \left[ \| P(x; \theta) - y_l \| + \beta \| S(P(x; \theta), z; \phi) - y_{ab} \| \right]. \quad (3)$$

where $P$ is the pre-processing model with weights $\theta$, $S$ is the source-reference model with weights $\phi$, and $\beta \in \mathbb{R}$ is a weighting hyperparameter.

Training is done using batches of videos with 5 sequential frames each, that are chosen randomly from the training data. For each 5-frame video, a random number of color references images $z$ is.

### Table 2. Overview of the source-reference model architecture.

This model takes as an input both the output of the pre-processing model and a set of reference images. Both these inputs are processed by separate encoders (a), then processed in two different middle branches corresponding to $1/8$ width and height (b), and $1/16$ width and height (c), before being decoded to the chrominance channels of the output video with a decoder (d). We abbreviate Spatial Convolutions with “SConv,” Temporal Convolutions with “TConv,” and Source-Reference Attention with “SR Attn.” For the source and reference encoders, we refer to the temporal dimension generically as $T$, where $T = T_s$ for the reference encoder and $T = T_r$ for the source encoder. We specify layer irregularities in the notes column. When the same layer is repeated several times consecutively, we indicate this with the number of times in parenthesis.

| (a) Source and Reference Encoders | (b) Middle $1/8$ branch | (c) Middle $1/16$ branch | (d) Decoder |
|----------------------------------|-------------------------|--------------------------|-------------|
| **Layer Type** | **Output Resolution** | **Notes** | **Layer Type** | **Output Resolution** | **Notes** | **Layer Type** | **Output Resolution** | **Notes** |
| Input | $(1 \ or \ 3) \times X \times W \times H$ | $3$ channels (RGB) for reference, $1$ channel (grayscale) for source | TConv | $512 \times T_s \times W_s \times H_s$ | Input is source encoder output, spatial stride of $2$ | TConv | $512 \times T_s \times W_s \times H_s$ | Uses $1/8$ source and reference as inputs |
| SConv | $64 \times T \times W/2 \times H/2$ | Spatial stride of $2$ | SConv | $512 \times T_s \times W_s \times H_s$ | Input is reference encoder output, spatial stride of $2$ | SConv | $512 \times T_s \times W_s \times H_s$ | Outputs $1/8$ source |
| SConv | $128 \times T \times W/2 \times H/2$ | | SConv | $512 \times T_s \times W_s \times H_s$ | | | |
| SConv | $256 \times T \times W/4 \times H/4$ | Spatial stride of $2$ | SConv | $512 \times T_s \times W_s \times H_s$ | Outputs $1/16$ reference | |
| SConv | $256 \times T \times W/4 \times H/4$ | | SR Attn. | $512 \times T_s \times W_s \times H_s$ | | | |
| SConv | $512 \times T \times W/8 \times H/8$ | Spatial stride of $2$ | TConv | $512 \times T_s \times W_s \times H_s$ | | |
| SConv | $512 \times T \times W/8 \times H/8$ | | Self Attn. | $512 \times T_s \times W_s \times H_s$ | | |
| | | | | | | | |
| **Layer Type** | **Output Resolution** | **Notes** | **Layer Type** | **Output Resolution** | **Notes** | **Layer Type** | **Output Resolution** | **Notes** |
| SR Attn. | $512 \times T_s \times W_s \times H_s$ | Input is source and reference encoder output | TConv | $256 \times T_s \times W_s \times H_s$ | | |
| TConv | $512 \times T_s \times W_s \times H_s$ | | TConv | $128 \times T_s \times W_s \times H_s$ | | | |
| TConv | $512 \times T_s \times W_s \times H_s$ | | TConv | $64 \times T_s \times W_s \times H_s$ | | | |
| TConv | $512 \times T_s \times W_s \times H_s$ | | TConv | $32 \times T_s \times W_s \times H_s$ | | | |
| TConv | $512 \times T_s \times W_s \times H_s$ | | TConv | $16 \times T_s \times W_s \times H_s$ | | | |
| TConv | $512 \times T_s \times W_s \times H_s$ | | TConv | $8 \times T_s \times W_s \times H_s$ | | | |
| TConv | $2 \times T_s \times W_s \times H_s$ | | TConv | $2 \times T_s \times W_s \times H_s$ | | | |

ACM Trans. Graph., Vol. 38, No. 6, Article 176. Publication date: November 2019.
We perform large amounts of data augmentation to the input video, based on the YouTube-8M dataset \cite{abu2016youtube} which consists of roughly 8 million videos corresponding to about 500 thousand hours of video data. The dataset is annotated with 4,803 visual entities which we do not use. We convert the videos totalling 10,243,010 frames, of which we use 1,219 (7,993,132 frames) for testing.

### 4.3 Data Augmentation

We apply large amounts of transformations that affect the input video \(x\) and ground truth video \(y\), the same transformation is done to all the frames in the video, while in the case of the reference images \(z\), the transformation is done independently for each image as they are not related to each other.

| Name          | Target | Prob. | Range         | Notes                  |
|---------------|--------|-------|---------------|------------------------|
| Horiz. Flip   | \((x, y, z)\) | 50%   | \((-\infty, \infty)\) | -                      |
| Scaling       | \((x, y)\) | 100% | \([0, 256]\)   | Size of the smallest edge \((px)\), randomly crops |
| Rotation      | \((x, y)\) | 100% | \([0, 2\pi]\) | In degrees              |
| Brightness    | \((x, y)\) | 20%  | \([0, 1]\)    | Encoding quality        |
| Contrast      | \((x, y)\) | 20%  | \([0, 1]\)    | Encoding quality        |
| JPEG Noise    | \((x, y)\) | 90%  | \([0, 256]\)  | Encoding quality        |
| Noise         | \((x, y)\) | 10%  | \([0, 256]\)  | Gaussian                |
| Blur          | \((x, y)\) | 50%  | \([0, 256]\)  | Bicubic down-sampling   |
| Contrast      | \((x, y)\) | 33%  | \([0, 256]\)  | Bicubic down-sampling   |
| Scaling       | \((x, y)\) | 100% | \([0, 256]\)  | Size of the smallest edge \((px)\), randomly crops |
| Saturation    | \((x, y)\) | 10%  | \([0, 256]\)  | Size of the smallest edge \((px)\), randomly crops |

| Name          | Target | Prob. | Range         | Notes                  |
|---------------|--------|-------|---------------|------------------------|
| Brightness    | \((x, y)\) | 20%  | \([0, 1]\)    | Encoding quality        |
| Contrast      | \((x, y)\) | 20%  | \([0, 1]\)    | Encoding quality        |
| JPEG Noise    | \((x, y)\) | 90%  | \([0, 256]\)  | Encoding quality        |
| Noise         | \((x, y)\) | 10%  | \([0, 256]\)  | Gaussian                |
| Blur          | \((x, y)\) | 50%  | \([0, 256]\)  | Bicubic down-sampling   |
| Contrast      | \((x, y)\) | 33%  | \([0, 256]\)  | Bicubic down-sampling   |
| Saturation    | \((x, y)\) | 10%  | \([0, 256]\)  | Size of the smallest edge \((px)\), randomly crops |

### 4.4 Example-based Deterioration

In addition to all the different data augmentation techniques, we also simulate deterioration of the film medium from a dataset of 6,152 examples, such as fractal noise, grain noise, dust, and scratches. These deterioration examples are manually collected by web search using the keywords “film noise”, and also generated using software such as Adobe After Effects. For generated noise, fractal noise is used to generate a base noise pattern, which can then be improved by modifying the contrast, brightness, and tone curves to obtain scratch and dust-like noise. In total, 4,340 noise images were downloaded and 1,812 were generated. Some of the deterioration examples are shown in Fig. 6. In particular, as these deterioration effects simulate the degradation of the physical medium which is supporting the film, they are implemented as additive noise: the noise data is randomly added to the input greyscale video, independently for each frame. Furthermore, they are added independently of each other and combined to create the augmented input videos.

For all the noise, we use similar data augmentation techniques as used for the input video. In particular, the noise images are scaled randomly such that the shortest edge is between \([256, 720]\) pixels,
We train our model on our dataset with $\gamma$.
Training is initially done of the pre-processing network and source-network algorithm [Zeiler 2012], which is a variant of stochastic gradient compression artifacts and film scratches, are randomly added. Video licensed in the Public Domain.

both horizontally and vertically flipped with 50% probability, rotated randomly between $[-5,5]$ degrees, cropped to $256 \times 256$ pixels, randomly scaled by $U(0.5, 1.5)$, and randomly either subtracted or added to the original image. Some generated training examples are shown in Fig. 7.

4.5 Optimization
Training is initially done of the pre-processing network and source-reference network separately for 500,000 iterations. Afterwards, they are trained together in an end-to-end fashion by optimizing Eq. (3). For the optimization method, we rely on the ADADELTA algorithm [Zeiler 2012], which is a variant of stochastic gradient descent which heuristically estimates the learning rate parameter, thus requiring no hyper-parameters to tune.

5 RESULTS
We train our model on our dataset with $\gamma = 10^{-4}$ and a batch-size of 20. We use the model with the lowest validation loss as our final model. We evaluate both quantitative and qualitative and compare with existing methods.

5.1 Comparison with Existing Approaches
We compare the results of our approach with both existing approaches and strong baselines with a quantitative evaluation. In particular, for restoration, we compare against the approach of [Zhang et al. 2017b] and [Yu et al. 2018], and for colorization we compare against the propagation-based approach of [Vondrick et al. 2018] and single-image interactive approach of [Zhang et al. 2017a]. For both remastering, i.e., joint restoration and colorization, we compare against all possible combinations of restoration and colorization approaches, e.g., the combination of [Zhang et al. 2017b] and [Vondrick et al. 2018] used together. The approach of [Zhang et al. 2017b] and [Yu et al. 2018] consists of a deep residual convolutional neural network for single image restoration. We note that the approach of [Yu et al. 2018] is an extension of [Fan et al. 2018] and winner of the NTIRE 2018 super resolution image challenge. We modified the model of [Yu et al. 2018] by removing the up-sampling layer at the end as the target task is restoration and not super-resolution. The approach of [Vondrick et al. 2018] is a recursive convolutional neural network that can propagate color information.

The approach of [Zhang et al. 2017a] is a single-image convolutional neural network approach that can use user-provided hints, which we use to provide the reference image color information. We also compare against two strong colorization baselines consisting of our proposed model with the temporal convolution layers replaced with spatial convolution layers, and of our proposed model without self-attention layers. For restoration, we compare to a baseline consisting of our pre-processing network without the skip connection. Finally, we also compare against a baseline consisting of the restoration and colorization networks of our approach trained independently, i.e., without joint training. All approaches are trained using exactly the same training data and training approach for fair comparison.

We compare using our test set consisting of 300 videos from the Youtube-8M dataset. For each video we randomly sample a subset of either 90 or 300 frames, and use the subset as the ground truth. Given that these videos are not noisy nor degraded, we follow the same approach for generating training data to generate deteriorated inputs for evaluation. For the example-based deterioration effects, we use a different set of images from those of the training set to evaluate generalization. We use Peak Signal-to-Noise Ratio (PSNR) as an evaluation metric, and compute the PSNR using the luminance channel only for the restoration task, using the chrominance channels only for the colorization task, and using all the image channels for the remastering task.

For the reference color images, in the case of the 90 frame subset, we only provide the first frame as a reference image, while in the case of the 300 frame subset, we provide every 60th frame starting from the first frame as a reference image. For our approach, all the reference frames are provided at all times. In the case of the approach of [Vondrick et al. 2018], as it only propagates the color and is unable to naturally handle multiple reference images, we replace the output image with the new reference image when necessary as shown in Fig. 3. We note that the same random subset of all videos is used for all the approaches.

5.1.1 Remastering Results. As there is not a single approach that can handle the remastering of videos, we compare against a pipeline approach of first processing the video with the method of either [Zhang et al. 2017b] or [Yu et al. 2018], and then propagating the reference color on the output with the approach of either [Vondrick et al. 2018] or [Zhang et al. 2017a]. We also provide results of a baseline consisting of our full approach without the joint training, i.e., the restoration and colorization networks are trained independently. Results are shown in Table 4. Of the pipeline-based approaches, we find that, while they have similar performance, the combination of [Zhang et al. 2017b] and [Zhang et al. 2017a] gives the highest performance. However, our approach outperforms the existing pipeline
based approaches and the strong baseline that doesn’t use joint training. This shows that even though the restoration and colorization models are first trained independently before being further trained jointly, the joint training plays an important role in improving the quality of the final results. It is also interesting to point out that while the performance of existing approaches degrades with longer videos and more reference color images, our approach improves in performance. This is likely due to all the reference color images being used to remaster each frame. Several randomly chosen examples are shown in Fig. 8, where we can see that existing approaches fail to both remove the noise and propagate the color, while our approach performs well in both cases.

5.1.2 Restoration Results. We compare our approach with that of [Zhang et al. 2017b], [Yu et al. 2018], and a baseline for video restoration. The baseline consists of our pre-processing model without the skip connection that adds the input to the output. As color is not added, no reference color images are provided and the evaluation is done using only the 300 frame subset. Results are shown in Table 5. We can see that the baseline, the approach of [Zhang et al. 2017b], and the approach of [Yu et al. 2018] perform similarly, while our full pre-processing model, with a skip connection, outperforms both. Example results are shown in Fig. 9.

5.1.3 Colorization Results. We compare against the approach of [Zhang et al. 2017a] using global hints, the approach of [Vondrick et al. 2018] and two baselines: one consisting of our source-reference network without temporal convolutions and one without self-attention for colorization. Results are shown in Table 6, and we can see that our approach outperforms existing approaches and the baselines. Similar to the remastering case, our approach performs significantly better on longer videos with additional references images, which is indicative of the capabilities of the source-reference attention: not only is it possible to colorize long sequences with many reference images, it is beneficial for performance. An interesting result is that

Table 4. Quantitative remastering results. We compare the results of our model with that of restoring each frame with the approach of [Zhang et al. 2017b], and propagating reference color with the approach of [Vondrick et al. 2018] on synthetically deteriorated videos from the Youtube-8M dataset, and with a baseline that consists of our model without using joint training. We perform two types of experiments: one using a random 90-frame subset from each video with 1 reference frame, and one using a random 300-frame subset with 5 reference frames.

| Approach | Frames | # Ref. | PSNR   |
|----------|--------|--------|--------|
| Zhang+[2017b]&Zhang+[2017a] | 90 | 1 | 27.13 |
|          | 300 | 5 | 27.31 |
| Yu+[2018]&Zhang+[2017a] | 90 | 1 | 26.43 |
|          | 300 | 5 | 26.59 |
| Zhang+[2017b]&Vondrick+[2018] | 90 | 1 | 26.43 |
|          | 300 | 5 | 26.60 |
| Yu+[2018]&Vondrick+[2018] | 90 | 1 | 26.85 |
|          | 300 | 5 | 26.89 |
| Ours %o joint training | 90 | 1 | 29.07 |
|          | 300 | 5 | 29.23 |
| Ours | 90 | 1 | 30.83 |
|          | 300 | 5 | 31.14 |

ACM Trans. Graph., Vol. 38, No. 6, Article 176. Publication date: November 2019.
Fig. 9. **Restoration results on the Youtube-8M test dataset with degradation noise.** We show one frame from several examples and compare our approach with the approaches of [Zhang et al. 2017b] and [Yu et al. 2018]. The first column shows the input frame which has been deteriorated with noise, the next three columns correspond to the black and white restoration of each approach, and the last column corresponds to the ground truth video. Videos courtesy of Naa Creation (top), and Mayda Tapanes (bottom) and licensed under CC-by.

Fig. 10. **Colorization results on the Youtube-8M test dataset.** We show one frame from several examples and compare our approach with the colorization approach of [Zhang et al. 2017a] without using the reference image and the RNN-based approach [Vondrick et al. 2018] which uses the reference image. The first column shows the input frame, the next three columns correspond to the colorization of each approach, and the last corresponds to the reference image. Note that the input frame is not the same frame as the reference image. Videos courtesy of Naa Creation (top), and Mayda Tapanes (bottom) and licensed under CC-by.

Table 5. **Quantitative restoration results.** We compare the results of our pre-processing network with the approach of [Zhang et al. 2017b], [Yu et al. 2018], and a baseline of our approach without the skip connection for restoring synthetically deteriorated videos from the Youtube-8M dataset.

| Approach                | Frames | Ref. | PSNR  |
|-------------------------|--------|------|-------|
| [Zhang et al. 2017b]    | 300    | -    | 25.08 |
| [Yu et al. 2018]        | 300    | -    | 24.49 |
| Ours w/o skip connection| 300    | -    | 24.73 |
| Ours                    | 300    | -    | **26.13** |

Table 6. **Quantitative colorization results.** We compare the colorization results of our source-reference network with the approach of [Zhang et al. 2017a] using global hints, and [Vondrick et al. 2018] for the colorization of videos from the Youtube-8M dataset. We perform two types of experiments: one using a random 90-frame subset from each video with 1 reference frame, and one using a random 300-frame subset with 5 reference frames.

| Approach                | Frames | Ref. | PSNR  |
|-------------------------|--------|------|-------|
| [Zhang et al. 2017a]    | 90     | 1    | 31.28 |
|                         | 300    | 5    | 31.16 |
| [Vondrick et al. 2018]  | 90     | 1    | 31.55 |
|                         | 300    | 5    | 31.70 |
| Ours w/o temporal conv. | 90     | 1    | 28.46 |
|                         | 300    | 5    | 28.51 |
| Ours w/o self-attention | 90     | 1    | 29.00 |
|                         | 300    | 5    | 28.72 |
| Ours                    | 90     | 1    | **34.94** |
|                         | 300    | 5    | **36.26** |

self-attention plays a critical role in our model. We believe this is due to the fact it allows each output pixel to be computed using information from the entire image, which would require many more convolutional layers if self-attention was not employed. Example results are shown in Fig. 10.

5.2 Qualitative Results
We show qualitative results in Fig. 11 on diverse challenging real world vintage film examples. As the videos are originally color, we...
Fig. 11. **Qualitative comparison with the combined approach of Zhang+[2017b] and Vondrick+[2018].** We show the reference color images in the first row with their timestamps. Afterwards four different frames taken from the input video and output videos are shown. Note that the example of (d) is remastered with 41 reference images of which we only show a subset. "Right to Health, A (Part I)", "Freedom Highway", "Color Craziness", and "The Jungle Book" are licensed in the public domain.
We have presented an approach for the remastering of vintage film. We can see how the approach of [Zhang et al. 2017b] can restore with a Nvidia GTX 1080Ti GPU, with 4ms corresponding to the restoration stage, and 65ms corresponding to the colorization stage.

We also perform a qualitative comparison of restoration results on vintage film in Fig. 12 with the approach of [Zhang et al. 2017b]. We can see how our approach is able to perform a consistent remastering, while existing approaches lose track of the colorization and fail to produce pleasing results, which is consistent with our quantitative evaluation.

We perform a qualitative comparison of restoration results on vintage film, which is not possible to remaster with the current approach. The first row shows frames from the original input video and the second row shows the output of our approach. Images taken from the movie "Metropolis" (1925) which is licensed in the public domain.

Our model has a temporal resolution of 15 frames, corresponding to roughly half a second in most videos, which can lead to small temporal consistencies in the output video. For reference, existing approaches use a smaller amount such as 4 frames [Vondrick et al. 2018] or 10 frames [Lai et al. 2018]. While it should be possible to increase the temporal resolution, this leads to slower convergence and slower computation. While blind video temporal consistency techniques can alleviate this issue [Bonneel et al. 2015; Lai et al. 2018], we found that while they are able to slightly improve the temporal consistency, it comes at the cost of significantly worse results. We believe that integrating such an approach with our model and training end-to-end is a possible way to improve the temporal consistency without sacrificing the quality of the results.

We note that despite the progress in this work on remastering vintage film, due to the complexity of the task, it is still an open problem in computer graphics. Unlike most of the image and video research up until now, vintage film poses a much more difficult and realistic problem as highlighted in Fig. 2, and we hope that this work can further stimulate research in this topic.

ACKNOWLEDGMENTS

This work was partially supported by JST ACT-I (Iizuka, Grant Number: JPMJPR16U3), JST PRESTO (Simo-Serra, Grant Number: JPMJPR1756), and JST CREST (Iizuka and Simo-Serra, Grant Number: JPMJCR14D1).

REFERENCES

Bonneel, Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Apostol (Paul) Natesy, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. 2016. YouTube-8M: A Large-Scale Video Classification Benchmark. In arXiv:1609.08675. https://arxiv.org/pdf/1609.08675v1.pdf

An, Xiaobo and Fabio Pellacini. 2008. AppProp: All-pairs Appearance-space Edit Propagation. ACM Transactions on Graphics (Proceedings of SIGGRAPH) 27, 3 (Aug. 2008), 40:1–40:9.

Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. (2015).

Bako, Steve, Thijs Vogels, Brian McWilliams, Mark Meyer, Jan Novák, Alex Harvill, Pradeep Sen, Tony Derose, and Fabrice Rousselle. 2017. Kernel-predicting convolutional networks for denoising Monte Carlo renderings. ACM Transactions on Graphics (Proceedings of SIGGRAPH) 36, 4 (2017), 97:1–97:12.
