

Learning Temporal Coherence via Self-Supervision for GAN-based Video Generation

MENGYU CHU*, YOU XIE*, JONAS MAYER, LAURA LEAL-TAIXÉ, and NILS THUERUY,
Technical University of Munich, Germany

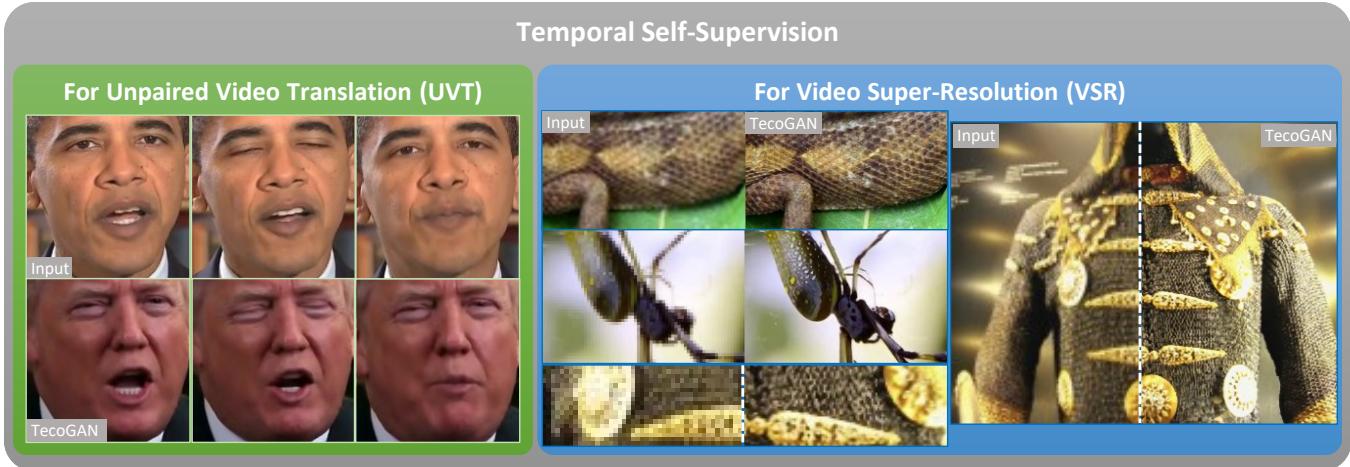


Fig. 1. Using the proposed approach for temporal self-supervision, we achieve realistic results with natural temporal evolution for two inherently different video generation tasks: unpaired video translation (left) and video super-resolution (right). While the resulting sharpness can be evaluated via the still images above, the corresponding videos in [our supplemental web-page](#) (Sec. 1 and Sec.2) highlight the high quality of the temporal changes. Obama and Trump video courtesy of the White House (public domain).

Our work explores temporal self-supervision for GAN-based video generation tasks. While adversarial training successfully yields generative models for a variety of areas, temporal relationships in the generated data are much less explored. Natural temporal changes are crucial for sequential generation tasks, e.g. video super-resolution and unpaired video translation. For the former, state-of-the-art methods often favor simpler norm losses such as L^2 over adversarial training. However, their averaging nature easily leads to temporally smooth results with an undesirable lack of spatial detail. For unpaired video translation, existing approaches modify the generator networks to form spatio-temporal cycle consistencies. In contrast, we focus on improving learning objectives and propose a temporally self-supervised algorithm. For both tasks, we show that temporal adversarial learning is key to achieving temporally coherent solutions without sacrificing spatial detail. We also propose a novel Ping-Pong loss to improve the long-term temporal consistency. It effectively prevents recurrent networks from accumulating artifacts temporally without depressing detailed features. Additionally, we propose a first set of metrics to quantitatively evaluate the accuracy as well as the perceptual quality of the temporal evolution. A series of user studies

confirm the rankings computed with these metrics. Code, data, models, and results are provided at <https://github.com/thunil/TecoGAN>. The project page <https://ge.in.tum.de/publications/2019-tecogan-chu/> contains supplemental materials.

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

Additional Key Words and Phrases: Generative adversarial network, temporal cycle-consistency, self-supervision, video super-resolution, unpaired video translation.

ACM Reference Format:

Mengyu Chu, You Xie, Jonas Mayer, Laura Leal-Taixé, and Nils Thuerey. 2020. Learning Temporal Coherence via Self-Supervision for GAN-based Video Generation. *ACM Trans. Graph.* 39, 4, Article 75 (July 2020), 19 pages. <https://doi.org/10.1145/3386569.3392457>

1 INTRODUCTION

Generative adversarial networks (GANs) have been extremely successful at learning complex distributions such as natural images [Isola et al. 2017; Zhu et al. 2017]. However, for sequence generation, directly applying GANs without carefully engineered constraints typically results in strong artifacts over time due to the significant difficulties introduced by the temporal changes. In particular, conditional video generation tasks are very challenging learning problems where generators should not only learn to represent the data distribution of the target domain but also learn to correlate the

*Both authors contributed equally to the paper

Authors' address: Mengyu Chu, mengyu.chu@tum.de; You Xie, you.xie@tum.de; Jonas Mayer, jonas.a.mayer@tum.de; Laura Leal-Taixé, leal.taixe@tum.de; Nils Thuerey, nils.thuerey@tum.de,
Technical University of Munich, Department of Computer Science, Munich, Germany.

© 2020 Copyright held by the owner/author(s). Publication rights licensed to ACM. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *ACM Transactions on Graphics*, <https://doi.org/10.1145/3386569.3392457>.

output distribution over time with conditional inputs. Their central objective is to faithfully reproduce the temporal dynamics of the target domain and not resort to trivial solutions such as features that arbitrarily appear and disappear over time.

In our work, we propose a novel adversarial learning method for a recurrent training approach that supervises both spatial contents as well as temporal relationships. As shown in Fig. 1, we apply our approach to two video-related tasks that offer substantially different challenges: *video super-resolution* (VSR) and *unpaired video translation* (UVT). With no ground truth motion available, the spatio-temporal adversarial loss and the recurrent structure enable our model to generate realistic results while keeping the generated structures coherent over time. With the two learning tasks we demonstrate how spatio-temporal adversarial training can be employed in paired as well as unpaired data domains. In addition to the adversarial network which supervises the short-term temporal coherence, long-term consistency is self-supervised using a novel bi-directional loss formulation, which we refer to as “Ping-Pong” (PP) loss in the following. The PP loss effectively avoids the temporal accumulation of artifacts, which can potentially benefit a variety of recurrent architectures. We also note that most existing image metrics focus on spatial content only. We fill the gap of temporal assessment with a pair of metrics that measures the perceptual similarity over time and the similarity of motions with respect to a ground truth reference. User studies confirm these metrics for both tasks.

The central contributions of our work are:

- a spatio-temporal discriminator unit together with a careful analysis of training objectives for realistic and coherent video generation tasks,
- a novel PP loss supervising long-term consistency,
- in addition to a set of metrics for quantifying temporal coherence based on motion estimation and perceptual distance.

Together, our contributions lead to models that outperform previous work in terms of temporally-coherent detail, which we qualitatively and quantitatively demonstrate with a wide range of content.

2 RELATED WORK

Deep learning has made great progress for image generation tasks. While regular losses such as L^2 [Kim et al. 2016; Lai et al. 2017] offer good performance for image super-resolution (SR) tasks in terms of PSNR metrics, previous work found adversarial training [Goodfellow et al. 2014] to significantly improve the perceptual quality in multi-modal settings such as image generation [Brock et al. 2019], colorization [He et al. 2018], super-resolution [Ledig et al. 2016], and translation [Isola et al. 2017; Zhu et al. 2017] tasks. Besides representing natural images, GAN-based frameworks are also successful at static graphic representations including geometry synthesis [Wu et al. 2019] and city modeling [Kelly et al. 2018].

Sequential generation tasks, on the other hand, require the generation of realistic content that changes naturally over time [Kim et al. 2019; Xie et al. 2018]. It is especially so for conditional video generation tasks [Jamriška et al. 2019; Sitzmann et al. 2018; Wronski et al. 2019; Zhang et al. 2019], where specific correlations between the input and the generated spatio-temporal evolution are required

when ground-truth motions are not provided. Hence, motion estimation [Dosovitskiy et al. 2015; Liu et al. 2019] and compensation become crucial for video generation tasks. The compensation can take various forms, e.g., explicitly using variants of optical flow networks [Caballero et al. 2017; Sajjadi et al. 2018; Shi et al. 2016] and implicitly using deformable convolution layers [Wang et al. 2019a; Zhu et al. 2019] or dynamic up-sampling [Jo et al. 2018]. In our work, a network is trained to estimate the motion and we show that it can help generators and discriminators in spatio-temporal adversarial training.

For VSR, recent work improve the spatial detail and temporal coherence by either using multiple low-resolution (LR) frames as inputs [Haris et al. 2019; Jo et al. 2018; Liu et al. 2017; Tao et al. 2017] or recurrently using previously estimated outputs [Sajjadi et al. 2018]. The latter has the advantage to re-use high-frequency details over time. In general, adversarial learning is less explored for VSR and applying it in conjunction with a recurrent structure gives rise to a special form of temporal mode collapse, as we will explain below. For video translation tasks, GANs are more commonly used but discriminators typically only supervise the spatial content. E.g., Zhu et al. [2017] focuses on images without temporal constraints and generators can fail to learn the temporal cycle-consistency for videos. In order to learn temporal dynamics, RecycleGAN [Bansal et al. 2018] proposes to use a prediction network in addition to a generator, while a concurrent work [Chen et al. 2019] chose to learn motion translation in addition to the spatial content translation. Being orthogonal to these works, we propose a spatio-temporal adversarial training for both VSR and UVT and we show that temporal self-supervision is crucial for improving spatio-temporal correlations without sacrificing spatial detail.

While L^1 and L^2 temporal losses based on warping are generally used to enforce temporal smoothness in video style transfer tasks [Chen et al. 2017; Ruder et al. 2016] and concurrent GAN-based VSR [Pérez-Pellitero et al. 2018] and UVT [Park et al. 2019] work, it leads to an undesirable smooth over spatial detail and temporal changes in outputs. Likewise, the L^2 temporal metric represents a sub-optimal way to quantify temporal coherence. For image similarity evaluation, perceptual metrics [Prashnani et al. 2018; Zhang et al. 2018] are proposed to reliably consider semantic features instead of pixel-wise errors. However, for videos, perceptual metrics that evaluate natural temporal changes are unavailable up to now. To address this open issue, we propose two improved temporal metrics and demonstrate the advantages of temporal self-supervision over direct temporal losses. Due to its complexity, VSR has also led to workshop challenges like NTIRE19 [Nah et al. 2019], where algorithms such as EDVR [Wang et al. 2019a] perform best w.r.t. PSNR-based metrics. We compare to these methods and give additional details in Appendix A.

Previous work, e.g. tempoGAN for fluid flow [Xie et al. 2018] and vid2vid for video translation [Wang et al. 2018a], has proposed adversarial temporal losses to achieve time consistency. While tempoGAN employs a second temporal discriminator with multiple aligned frames to assess the realism of temporal changes, it is not suitable for videos, as it relies on ground truth motions and employs single-frame processing that is sub-optimal for natural images. On the other hand, vid2vid focuses on paired video translations and

proposes a video discriminator based on a conditional motion input that is estimated from the paired ground-truth sequences. We focus on more difficult unpaired translation tasks instead and demonstrate the gains in the quality of our approach in the evaluation section. Bashkirova et al. [2018] solve UVT tasks as a 3D extension of the 2D image translation. In DeepFovea [Kapulyan et al. 2019], a 3D discriminator is used to supervise video in-painting results with 32 frames as a single 3D input. Since temporal evolution differs from a spatial distribution, we show how a separate handling of the temporal dimension can reduce computational costs, remove training restrictions, and most importantly improve inference quality.

For tracking and optical flow estimation, L^2 -based time-cycle losses [Wang et al. 2019b] were proposed to constrain motions and tracked correspondences using symmetric video inputs. By optimizing indirectly via motion compensation or tracking, this loss improves the accuracy of the results. For video generation, we propose a PP loss that also makes use of symmetric sequences. However, we directly constrain the PP loss via the generated video content, which successfully improves the long-term temporal consistency in the video results. The PP loss is effective by offering valid information in forward as well as backward passes of image sequences. This concept is also used in robotic control algorithms, where reversed trajectories starting from goal positions have been used as training data [Nair et al. 2018].

3 LEARNING TEMPORALLY COHERENT CONDITIONAL VIDEO GENERATION

We first propose the concepts of temporal self-supervision for GAN-based video generation (Sec. 3.1 and Sec. 3.2), before introducing solutions for VSR and UVT tasks (Sec. 3.3 and Sec. 3.4) as example applications.

3.1 Spatio-Temporal Adversarial Learning

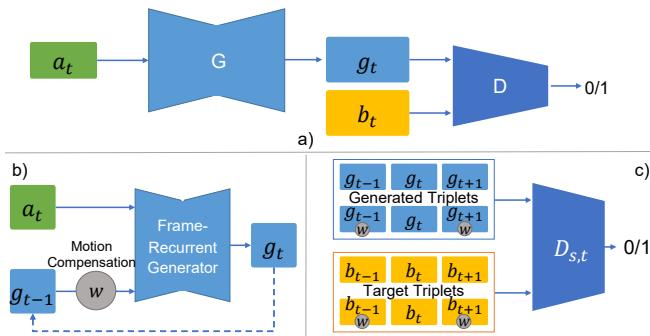


Fig. 2. a) A spatial GAN for image generation. b) A frame recurrent Generator. c) A spatio-temporal Discriminator. In these figures, letter a , b , and g , stand for the input domain, the output domain and the generated results respectively. G and D stand for the generator and the discriminator.

While GANs are popular and widely used in image generation tasks to improve perceptual quality, their spatial adversarial learning inherently introduces temporal problems for tasks such as video generation. Thus, we propose an algorithm for spatio-temporal

adversarial learning that is easy to integrate into existing GAN-based image generation approaches. Starting from a standard GAN for images, as shown in Fig. 2 a), we propose to use a frame-recurrent generator (b) together with a spatio-temporal discriminator (c).

As shown in Fig. 2 b), our generator produces an output g_t from an input frame a_t and recursively uses the previously generated output g_{t-1} . Following previous work [Sajjadi et al. 2018], we warp this frame-recurrent input to align it with the current frame. This allows the network to more easily re-use previously generated details. The high-level structure of the generator can be summarized as:

$$v_t = F(a_{t-1}, a_t), \quad g_t = G(a_t, W(g_{t-1}, v_t)). \quad (1)$$

Here, the network F is trained to estimate the motion v_t from frame a_{t-1} to a_t and W denotes warping.

The central building block of our approach is a novel *spatio-temporal* discriminator $D_{S,t}$ that receives triplets of frames, shown in Fig. 2 c). This contrasts with typically used *spatial* discriminators that supervise only a single image. By concatenating multiple adjacent frames along the channel dimension, the frame triplets form an important building block for learning as they can provide networks with gradient information regarding the realism of spatial structures as well as short-term temporal information, such as first- and second-order time derivatives.

We propose a $D_{S,t}$ architecture that primarily receives two types of triplets: three adjacent frames and the corresponding warped ones. We warp later frames backward and previous ones forward. The network F is likewise used to estimate the corresponding motions. While original frames contain the full spatio-temporal information, warped frames more easily yield temporal information with their aligned content. For the input variants we use the following notations: $I_g = \{g_{t-1}, g_t, g_{t+1}\}$, $I_b = \{b_{t-1}, b_t, b_{t+1}\}$; $I_{wg} = \{W(g_{t-1}, v_t), g_t, W(g_{t+1}, v'_t)\}$, $I_{wb} = \{W(b_{t-1}, v_t), b_t, W(b_{t+1}, v'_t)\}$. A subscript a denotes the input domain, while the b subscript denotes the target domain. The quotation mark in v' indicates that quantities are estimated from the backward direction.

Although the proposed concatenation of several frames seems like a simple change that has been used in a variety of other contexts, we show that it represents an important operation that allows discriminators to understand spatio-temporal data distributions. As will be shown below, it can effectively reduce temporal problems encountered by spatial GANs. While L^2 -based temporal losses are widely used in the field of video generation, the spatio-temporal adversarial loss is crucial for preventing the inference of blurred structures in multi-modal data-sets. Compared to GANs using multiple discriminators, the single $D_{S,t}$ network that we propose can learn to balance the spatial and temporal aspects according to the reference data and avoid inconsistent sharpness as well as overly smooth results. Additionally, by extracting shared spatio-temporal features, it allows for smaller network sizes.

3.2 Self-Supervision for Long-term Temporal Consistency

When relying on a previous output as input, i.e., for frame-recurrent architectures, generated structures easily accumulate frame by frame. In adversarial training, generators learn to heavily rely on previously generated frames and can easily converge towards strongly reinforcing spatial features over longer periods of time. For videos,

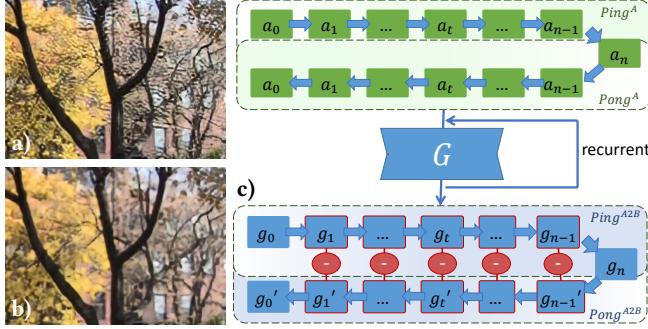


Fig. 3. a) Result without PP loss. The VSR network is trained with a recurrent frame-length of 10. When inference on long sequences, frame 15 and latter frames of the foliage scene show the drifting artifacts. b) Result trained with PP loss. These artifacts are removed successfully for the latter. c) When inferring a symmetric PP sequence with a forward pass (Ping) and its backward counterpart (Pong), our PP loss constrains the output sequence to be symmetric. It reduces the L^2 distance between g_t and g'_t , the corresponding frames in the forward and backward passes, shown via red circles with a minus sign. The PP loss reduces drifting artifacts and improves temporal coherence.

this especially occurs along directions of motion and these solutions can be seen as a special form of temporal mode collapse, where the training converges to a mostly constant temporal signal as a sub-optimal, trivial equilibrium. We have noticed this issue in a variety of recurrent architectures, examples are shown in Fig. 3 a) and the Dst version in Fig. 8.

While this issue could be alleviated by training with longer sequences, it is computationally expensive and can fail for even longer sequences, as shown in Appendix D. We generally want generators to be able to work with sequences of arbitrary length for inference. To address this inherent problem of recurrent generators, we propose a new bi-directional “Ping-Pong” loss. For natural videos, a sequence with the forward order as well as its reversed counterpart offer valid information. Thus, from any input of length n , we can construct a symmetric PP sequence in form of $a_1, \dots, a_{n-1}, a_n, a_{n-1}, \dots, a_1$ as shown in Fig. 3 c). When inferring this in a frame-recurrent manner, the generated result should not strengthen any invalid features from frame to frame. Rather, the result should stay close to valid information and be symmetric, i.e., the forward result $g_t = G(a_t, g_{t-1})$ and the one generated from the reversed part, $g'_t = G(a_t, g'_{t+1})$, should be identical.

Based on this observation, we train our networks with extended PP sequences and constrain the generated outputs from both “legs” to be the same using the loss: $\mathcal{L}_{pp} = \sum_{t=1}^{n-1} \|g_t - g'_t\|_2$. Note that in contrast to the generator loss, the L^2 norm is a correct choice here: We are not faced with multi-modal data where an L^2 norm would lead to undesirable averaging, but rather aim to constrain the recurrent generator to its own, unique version over time without favoring smoothness. The PP terms provide constraints for short term consistency via $\|g_{n-1} - g_{n-1}'\|_2$, while terms such as $\|g_1 - g_1'\|_2$ prevent long-term drifts of the results. This bi-directional loss formulation also helps to constrain ambiguities due to disocclusions that can occur in regular training scenarios.

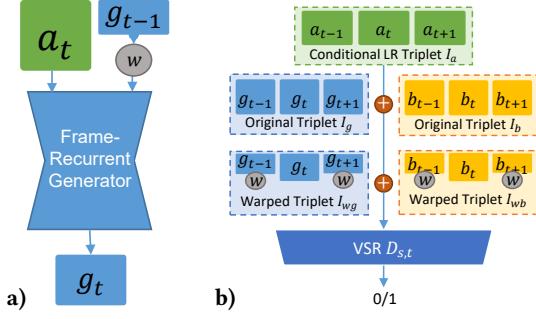


Fig. 4. a) The frame-recurrent VSR Generator. b) Conditional VSR $D_{s,t}$.

As shown in Fig. 3 b), the PP loss successfully removes drifting artifacts while appropriate high-frequency details are preserved. In addition, it effectively extends the training data set, and as such represents a useful form of data augmentation. A comparison is given in Appendix D to disentangle the effects of the augmentation of PP sequences and the temporal constraints. The results show that the temporal constraint is the key to reliably suppressing the temporal accumulation of artifacts, achieving consistency, and allowing models to infer much longer sequences than seen during training.

The majority of related work for video generation focuses on network architectures. Being orthogonal to architecture improvements, our work explores temporal self-supervision. The proposed spatio-temporal discriminator and the PP loss can be used in video generation tasks to replace simple temporal losses, e.g. the ones based on L^2 differences and warping. In the following subsections, solutions for VSR and UVT are presented as examples in paired and unpaired data domains.

3.3 Network Architecture for VSR

For video super-resolution (VSR) tasks, the input domain contains LR frames while the target domain contains high-resolution (HR) videos with more complex details and motions. Since one pattern in low-resolution can correspond to multiple structures in high-resolution, VSR represents a multimodal problem that benefits from adversarial training. In the proposed spatio-temporal adversarial training, we use a ResNet architecture for the VSR generator G . Similar to previous work, an encoder-decoder structure is applied to F for motion estimation. We intentionally keep the generative part simple and in line with previous work, in order to demonstrate the advantages of the temporal self-supervision.

The VSR discriminator $D_{s,t}$ should guide the generator to learn the correlation between the conditional LR inputs and HR targets. Therefore, three LR frames $I_a = \{a_{t-1}, a_t, a_{t+1}\}$ from the input domain are used as a conditional input. The input of $D_{s,t}$ can be summarized as $I_{s,t}^b = \{I_b, I_{wb}, I_a\}$ labelled as *real* and the generated inputs $I_{s,t}^g = \{I_g, I_{wg}, I_a\}$ labelled as *fake*, as shown in Fig. 4. We concatenate all triplets together. In this way, the conditional $D_{s,t}$ will penalize G if I_g contains less spatial details or unrealistic artifacts in comparison to I_a, I_b . At the same time, temporal relationships between the generated images I_{wg} and those of the ground truth I_{wb} should match. With our setup, the discriminator profits from the

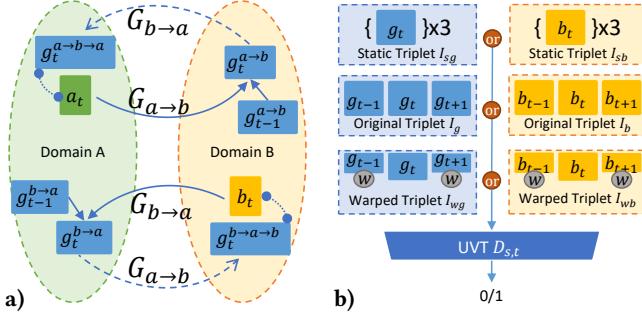


Fig. 5. a) The UVT cycle link formed by two recurrent generators. b) Unconditional UVT $D_{s,t}$.

warped frames to classify realistic and unnatural temporal changes, and for situations where the motion estimation is less accurate, the discriminator can fall back to the original, i.e. not warped, images.

3.4 Network Architecture for UVT

While one generator is enough to map data from A to B for tasks such as VSR, unpaired generation tasks require a second generator to establish cycle consistency [Zhu et al. 2017]. For the UVT task, we use two recurrent generators, mapping from domain A to B and back. As shown in Fig. 5 a), given $g_t^{a \rightarrow b} = G_{ab}(a_t, W(g_{t-1}^{a \rightarrow b}, v_t))$, we can use a_t as the labeled data of $g_t^{a \rightarrow b \rightarrow a} = G_{ba}(g_t^{a \rightarrow b}, W(g_{t-1}^{a \rightarrow b \rightarrow a}, v_t))$ to enforce consistency. An encoder-decoder structure is applied to UVT generators and F .

In UVT tasks, we demonstrate that the temporal cycle-consistency between different domains can be established using the supervision of unconditional spatio-temporal discriminators. This is in contrast to previous work which focuses on the generative networks to form spatio-temporal cycle links. Our approach actually yields improved results, as we will show below. In practice, we found it crucial to ensure that generators first learn reasonable spatial features, and only then improve their temporal correlation. Therefore, different to the $D_{s,t}$ of VST that always receives 3 concatenated triplets as an input, the unconditional $D_{s,t}$ of UVT only takes one triplet at a time. Focusing on the generated data, the input for a single batch can either be a static triplet of $I_{sg} = \{g_t, g_t, g_t\}$, the warped triplet I_{wg} , or the original triplet I_g . The same holds for the reference data of the target domain, as shown in Fig. 4 b). Here, the warping is again performed via F . With sufficient but complex information contained in these triplets, transition techniques are applied so that the network can consider the spatio-temporal information step by step, i.e., we initially start with 100% static triplets I_{sg} as the input. Then, over the course of training, 25% of them transit to I_{wg} triplets with simpler temporal information, with another 25% transition to I_g afterwards, leading to a (50%, 25%, 25%) distribution of triplets. Details of the transition calculations are given in Appendix C. Sample triplets are visualized in the supplemental web-page (Sec. 7).

While non-adversarial training typically employs loss formulations with static goals, the GAN training yields dynamic goals due to discriminators discovering learning objectives over the course of the training run. Therefore, their inputs have a strong influence on the training process and the final results. Modifying the inputs

in a controlled manner can lead to different results and substantial improvements if done correctly, as will be shown in Sec. 4.

3.5 Loss Functions

Perceptual Loss Terms. As perceptual metrics, both pre-trained NNs [Johnson et al. 2016; Wang et al. 2018b] and GAN discriminators [Xie et al. 2018] were successfully used in previous work. Here, we use feature maps from a pre-trained VGG-19 network [Simonyan and Zisserman 2014], as well as $D_{s,t}$ itself. In the VSR task, we can encourage the generator to produce features similar to the ground truth ones by increasing the cosine similarity of their feature maps. In UVT tasks without paired ground truth data, the generators should match the distribution of features in the target domain. Similar to a style loss for traditional style transfer tasks [Johnson et al. 2016], we thus compute the $D_{s,t}$ feature correlations measured by the Gram matrix for UVT tasks. The $D_{s,t}$ features contain both spatial and temporal information and hence are especially well suited for the perceptual loss.

Loss and Training Summary. We now explain how to integrate the spatio-temporal discriminator into the paired and unpaired tasks. We use a standard discriminator loss for the $D_{s,t}$ of VSR and a least-square discriminator loss for the $D_{s,t}$ of UVT. Correspondingly, a non-saturated \mathcal{L}_{adv} is used for the G and F of VSR and a least-squares one is used for the UVT generators. As summarized in Table 1, G and F are trained with the mean squared loss $\mathcal{L}_{content}$, adversarial losses \mathcal{L}_{adv} , perceptual losses \mathcal{L}_ϕ , the PP loss \mathcal{L}_{pp} , and a warping loss \mathcal{L}_{warp} , where again g , b and Φ stand for generated samples, ground truth images and feature maps of VGG or $D_{s,t}$. We only show losses for the mapping from A to B for UVT tasks, as the backward mapping simply mirrors the terms. We refer to our full model for both tasks as *TecoGAN* below. The UVT data-sets are obtained from previous work [Bansal et al. 2018] and each data domain has around 2400 to 3600 unpaired frames. For VSR, we download 250 short videos with 120 frames each from [Vimeo.com](https://vimeo.com). In line with other VSR projects, we down-sample these frames by a factor of 2 to get the ground-truth HR frames. Corresponding LR frames are achieved by applying a Gaussian blur and sampling every fourth pixel. A Gaussian blur step is important to mimic the information loss due to the camera sensibility in a real-life capturing scenario. Although the information loss is complex and not unified, a Gaussian kernel with a standard deviation of 1.5 is commonly used for a super-resolution factor of 4. Training parameters and details are given in Appendix F.

4 ANALYSIS AND EVALUATION OF LEARNING OBJECTIVES

In the following section, we illustrate the effects of temporal supervision using two ablation studies. In the first one, models trained with ablated loss functions show how \mathcal{L}_{adv} and \mathcal{L}_{pp} change the overall learning objectives. Next, full UVT models are trained with different $D_{s,t}$ inputs. This highlights how differently the corresponding discriminators converge to different spatio-temporal equilibriums and the general importance of providing suitable data distributions from the target domain. While we provide qualitative and quantitative

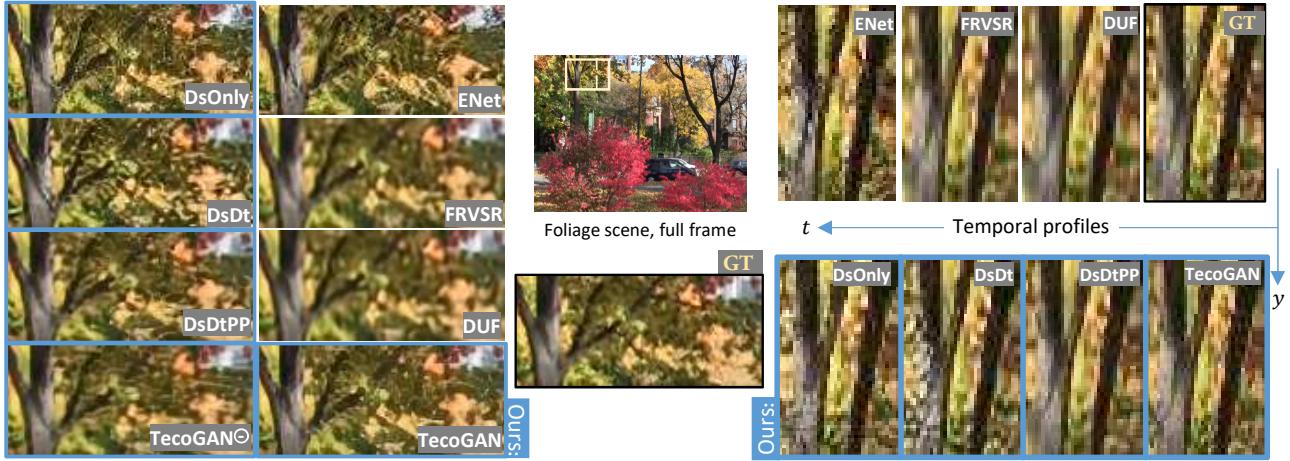


Fig. 6. In VSR of the foliage scene, adversarial models (ENet, DsOnly, DsDt, DsDtPP, TecoGAN[®] and TecoGAN) yield better perceptual quality than methods using L^2 loss (FRVSR and DUF). In temporal profiles on the right, DsDt, DsDtPP and TecoGAN show significantly less temporal discontinuities compared to ENet and DsOnly. The temporal information of our discriminators successfully suppresses these artifacts. Corresponding video clips can be found in Sec. 4.1-4.6 of [the supplemental web-page](#).



Fig. 7. VSR temporal profile comparisons of the calendar scene (time shown along y-axis), cf. Sec. 4.1-4.6 of the supplemental web-page. TecoGAN models lead to natural temporal progression, and our final model closely matches the desired ground truth behavior over time.

Table 1. Summary of loss terms.

$\mathcal{L}_{D_{s,t}}$ for	
VSR, $D_{s,t}$	$-\mathbb{E}_{b \sim p_b(b)}[\log D(I_{s,t}^b)] - \mathbb{E}_{a \sim p_a(a)}[\log(1 - D(I_{s,t}^g))]$
UVT, $D_{s,t}^b$	$\mathbb{E}_{b \sim p(b)}[D(I_{s,t}^b) - 1]^2 + \mathbb{E}_{a \sim p(a)}[D(I_{s,t}^g)]^2$
Loss for	VSR, G & F UVT, G_{ab}
$\mathcal{L}_{G,F}$	$\lambda_w \mathcal{L}_{\text{warp}} + \lambda_p \mathcal{L}_{\text{PP}} + \lambda_a \mathcal{L}_{\text{adv}} + \lambda_\phi \mathcal{L}_\phi + \lambda_c \mathcal{L}_{\text{content}}$
$\mathcal{L}_{\text{warp}}$	$\sum \ a_t - W(a_{t-1}, F(a_{t-1}, a_t))\ _2$
\mathcal{L}_{PP}	$\sum_{t=1}^{n-1} \ g_t - g_t'\ _2$
\mathcal{L}_{adv}	$-\mathbb{E}_{a \sim p_a(a)}[\log D_{s,t}(I_{s,t}^g)] - \mathbb{E}_{a \sim p_a(a)}[D_{s,t}^b(I_{s,t}^{g \rightarrow b})]^2$
\mathcal{L}_ϕ	$1.0 - \frac{\Phi(I_{s,t}^g) * \Phi(I_{s,t}^b)}{\ \Phi(I_{s,t}^g)\ * \ \Phi(I_{s,t}^b)\ } \quad \left\ GM(\Phi(I_{s,t}^g)) - GM(\Phi(I_{s,t}^b)) \right\ _2$
$\mathcal{L}_{\text{content}}$	$\ g_t - b_t\ _2 \quad \ g_t^{a \rightarrow b \rightarrow a} - a_t\ _2 + \ g_t^{b \rightarrow a \rightarrow b} - b_t\ _2$

evaluations below, we also refer the reader to our supplemental material which contains a [web-page](#) with video clips that more clearly highlight the temporal differences.

4.1 Loss Ablation Study

Below we compare variants of our full TecoGAN model to EnhanceNet (ENet) [Sajjadi et al. 2017], FRVSR [Sajjadi et al. 2018], and DUF [Jo et al. 2018] for VSR. CycleGAN [Zhu et al. 2017] and RecycleGAN [Bansal et al. 2018] are compared for UVT. Specifically, ENet and CycleGAN represent state-of-the-art single-image adversarial models without temporal information, FRVSR and DUF are state-of-the-art VSR methods without adversarial losses, and RecycleGAN is a spatial adversarial model with a prediction network learning the temporal evolution.

Video Super-Resolution. For VSR, we first train a *DsOnly* model that uses a frame-recurrent *G* and *F* with a VGG loss and only the regular spatial discriminator. Compared to ENet, which exhibits strong incoherence due to the lack of temporal information, *DsOnly* improves temporal coherence thanks to the frame-recurrent connection, but there are noticeable high-frequency changes between



Fig. 8. When learning a mapping between Trump and Obama, the CycleGAN model gives good spatial features but collapses to essentially static outputs of Obama. It manages to transfer facial expressions back to Trump using tiny differences encoded in its Obama outputs, without understanding the cycle-consistency between the two domains. Being able to establish the correct temporal cycle-consistency between domains, ours, RecycleGAN and STC-V2V can generate correct blinking motions, shown in Sec. 4.7 of the supplemental web-page. Our model outperforms the latter two in terms of coherent detail that is generated. Obama and Trump video courtesy of the White House (public domain).

frames. The temporal profiles of DsOnly in Fig. 6 and 7, correspondingly contain sharp and broken lines.

When adding a temporal discriminator in addition to the spatial one ($D_{s,t}$), this version generates more coherent results, and its temporal profiles are sharp and coherent. However, $D_{s,t}$ often produces the drifting artifacts discussed in Sec. 3, as the generator learns to reinforce existing details from previous frames to fool D_s with sharpness, and satisfying D_t with good temporal coherence in the form of persistent detail. While this strategy works for generating short sequences during training, the strengthening effect can lead to very undesirable artifacts for long-sequence inferences.

By adding the self-supervision for long-term temporal consistency \mathcal{L}_{pp} , we arrive at the $DsDtPP$ model, which effectively suppresses these drifting artifacts with an improved temporal coherence. In Fig. 6 and Fig. 7, $DsDtPP$ results in continuous yet detailed temporal profiles without streaks from temporal drifting. Although $DsDtPP$ generates good results, it is difficult in practice to balance the generator and the two discriminators. The results shown here were achieved only after numerous runs manually tuning the weights of the different loss terms. By using the proposed $D_{s,t}$ discriminator instead, we get a first complete model for our method, denoted as $Tecogan^{\circ}$. This network is trained with a discriminator that achieves an excellent quality with an effectively halved network size, as illustrated on the right of Fig. 14. The single discriminator correspondingly leads to a significant reduction in resource usage. Using two discriminators requires ca. 70% more GPU memory, and leads to a reduced training performance by ca. 20%. The

$Tecogan^{\circ}$ model yields similar perceptual and temporal quality to $DsDtPP$ with a significantly faster and more stable training.

Since the $Tecogan^{\circ}$ model requires less training resources, we also trained a larger generator with 50% more weights. In the following, we will focus on this larger single-discriminator architecture with PP loss as our full $Tecogan$ model for VSR. Compared to the $Tecogan^{\circ}$ model, it can generate more details, and the training process is more stable, indicating that the larger generator and $D_{s,t}$ are more evenly balanced. Result images and temporal profiles are shown in Fig. 6 and Fig. 7. Video results are shown in Sec. 4 of the supplemental web-page.

Unpaired Video Translation. We carry out a similar ablation study for the UVT task. Again, we start from a single-image GAN-based model, a *CycleGAN* variant which already has two pairs of spatial generators and discriminators. Then, we train the *DsOnly* variant by adding flow estimation via F and extending the spatial generators to frame-recurrent ones. By augmenting the two discriminators to use the triplet inputs proposed in Sec. 3, we arrive at the *Dst* model with spatio-temporal discriminators, which does not yet use the PP loss. By adding the PP loss we complete the $Tecogan$ model for UVT. Although UVT tasks substantially differ from VSR tasks, the comparisons in Fig. 8 and Sec. 4.7 of our supplemental web-page illustrate that UVT tasks profit from the proposed approach in a very similar manner to VSR.

We use renderings of 3D fluid simulations of rising smoke as our unpaired training data. These simulations are generated with randomized numerical simulations using a resolution of 64^3 for

domain A and 256^3 for domain B, and both are visualized with images of size 256^2 . Therefore, video translation from domain A to B is a tough task, as the latter contains significantly more turbulent and small-scale motions. With no temporal information available, the CycleGAN variant generates HR smoke that strongly flickers. The DsOnly model offers better temporal coherence by relying on its frame-recurrent input, but it learns a solution that largely ignores the current input and fails to keep reasonable spatio-temporal cycle-consistency links between the two domains. On the contrary, our $D_{s,t}$ enables the Dst model to learn the correlation between the spatial and temporal aspects, thus improving the cycle-consistency. However, without \mathcal{L}_{pp} , the Dst model (like the DsDt model of VSR) reinforces detail over time in an undesirable way. This manifests itself as inappropriate smoke density in empty regions. Using our full TecoGAN model which includes \mathcal{L}_{pp} , yields the best results, with detailed smoke structures and very good spatio-temporal cycle-consistency.

For comparison, a DsDtPP model with a larger number of networks, i.e. four discriminators, two frame-recurrent generators and the F , is trained. By weighting the temporal adversarial losses from Dt with 0.3 and the spatial ones from Ds with 0.5, we arrived at a balanced training run. Although this model performs similarly to the TecoGAN model on the smoke dataset, the proposed spatio-temporal $D_{s,t}$ architecture represents a more preferable choice in practice, as it learns a natural balance of temporal and spatial components by itself, and requires fewer resources. Continuing along this direction, it will be interesting future work to evaluate variants, such as a shared $D_{s,t}$ for both domains, i.e. a multi-class classifier network. Besides the smoke dataset, an ablation study for the Obama and Trump dataset from Fig. 8 shows a very similar behavior, as can be seen in the supplemental web-page (Sec. 4.7).

4.2 Spatio-temporal Adversarial Equilibria

Our evaluation so far highlights that temporal adversarial learning is crucial for achieving spatial detail that is coherent over time for VSR, and for enabling the generators to learn the spatio-temporal correlation between domains in UVT. Next, we will shed light on the complex spatio-temporal adversarial learning objectives by varying the information provided to the discriminator network. In the following tests, shown in Fig. 9 and Sec. 5 of the supplemental document, $D_{s,t}$ networks are identical apart from changing inputs, and we focus on the smoke dataset.

In order to learn the spatial and temporal features of the target domain as well as their correlation, the simplest input for $D_{s,t}$ consists of only the original, unwarped triplets, i.e. $\{I_g \text{ or } I_b\}$. Using these, we train a *baseline* model, which yields a sub-optimal quality: it lacks sharp spatial structures and contains coherent but dull motions. Despite containing the full information, these input triplets prevent $D_{s,t}$ from providing the desired supervision. For paired video translation tasks, the *vid2vid* network achieves improved temporal coherence by using a video discriminator to supervise the output sequence conditioned with the ground-truth motion. With no ground-truth data available, we train a *vid2vid* variant by using the estimated motions and original triplets, i.e. $\{I_g + F(g_{t-1}, g_t) + F(g_{t+1}, g_t) \text{ or } I_b + F(b_{t-1}, b_t) + F(b_{t+1}, b_t)\}$, as the input for $D_{s,t}$. However, the

result do not significantly improve. The motions are only partially reliable, and hence don't help for the difficult unpaired translation task. Therefore, the discriminator still fails to fully correlate spatial and temporal features.

We then train a third model, *concat*, using the original triplets and the warped ones, i.e. $\{I_g + I_{wg} \text{ or } I_b + I_{wb}\}$. In this case, the model learns to generate more spatial details with a more vivid motion. I.e., the improved temporal information from the warped triplets gives the discriminator important cues. However, the motion still does not fully resemble the target domain. We arrive at our final TecoGAN model for UVT by controlling the composition of the input data: as outlined above, we first provide only static triplets $\{I_{sg} \text{ or } I_{sb}\}$, and then apply the transitions of warped triplets $\{I_{wg} \text{ or } I_{wb}\}$, and original triplets $\{I_g \text{ or } I_b\}$ over the course of training. In this

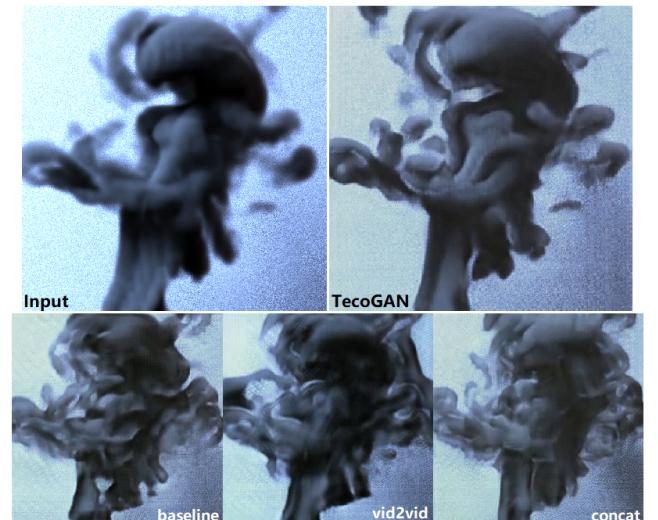


Fig. 9. Adversarial training arrives at different equilibria when discriminators use different inputs. The baseline model (supervised on original triplets) and the vid2vid variant (supervised on original triplets and estimated motions) fail to learn the complex temporal dynamics of a high-resolution smoke. The warped triplets improve the result of the concat model and the full TecoGAN model performs better spatio-temporally. Video comparisons are shown in Sec 5. of the supplemental web-page.



Fig. 10. Video translations between renderings of smoke simulations and real-world captures for smokes.



Fig. 11. Additional VSR comparisons, with videos in Sec 2 of the supplemental web-page. The TecoGAN model generates sharp details in both scenes.

way, the network can first learn to extract spatial features and build on them to establish temporal features. Finally, discriminators learn features about the correlation of spatial and temporal content by analyzing the original triplets and provide gradients such that the generators learn to use the motion information from the input and establish a correlation between the motions in the two unpaired domains. Consequently, the discriminator, despite receiving only a single triplet at once, can guide the generator to produce detailed structures that move coherently.

5 RESULTS AND METRIC EVALUATION

For the VSR task, we test our model on a wide range of video data, including the widely used Vid4 dataset shown in Fig. 6, 7 and 12, detailed scenes from the movie Tears of Steel (ToS) [2011] shown in Fig. 12, and others shown in Fig. 11. Besides ENet, FRVSR and DUF as baselines, we further compare our TecoGAN model to RBPN [Haris et al. 2019] and EDVR [Wang et al. 2019a]. Note that in contrast to TecoGAN with 3 million trainable weights, the latter two use substantially larger networks which have more than 12 and 20 million weights respectively, and EDVR is trained using bi-cubic down-sampling. Thus, in the following quantitative and qualitative comparisons, results and metrics for EDVR are calculated using bi-cubic down-sampled images, while other models use LR inputs with a Gaussian blur. In the supplemental web-page, Sec. 2 contains video results for the stills shown in Fig. 11, while Sec. 3 shows video comparisons of Vid4 scenes.

Trained with down-sampled inputs with Gaussian blur, the VSR TecoGAN model can similarly work with original images that were not down-sampled or filtered, such as a data-set of real-world photos. In Fig. 13, we compared our results to two other methods [Liao et al. 2015; Tao et al. 2017] that have used the same dataset. With the help of adversarial learning, our model is able to generate improved and realistic details in down-sampled images as well as captured images.

For UVT tasks, we train models for Obama and Trump translations, LR- and HR- smoke simulation translations, as well as translations between smoke simulations and real-smoke captures. While smoke simulations usually contain strong numerical viscosity with details limited by the simulation resolution, the real smoke from Eckert et al. [2018] contains vivid motions with many vortices and high-frequency details. As shown in Fig. 10, our method can be used to narrow the gap between simulations and real-world phenomena.

While visual results discussed above provide a first indicator of the quality our approach achieves, quantitative evaluations are crucial for automated evaluations across larger numbers of samples. Below we focus more on the VSR task as ground-truth data is available. We conduct user studies and present evaluations of the different models w.r.t. established spatial metrics. We also motivate and propose two novel temporal metrics to quantify temporal coherence.

For evaluating image SR, Blau and Michaeli [2018] demonstrated that there is an inherent trade-off between the perceptual quality of the result and the distortion measured with vector norms or low-level structures such as PSNR and SSIM. On the other hand, metrics based on deep feature maps such as LPIPS [Zhang et al. 2018] can capture more semantic similarities. We measure the PSNR and LPIPS using the Vid4 scenes. With a PSNR decrease of less than 2dB over

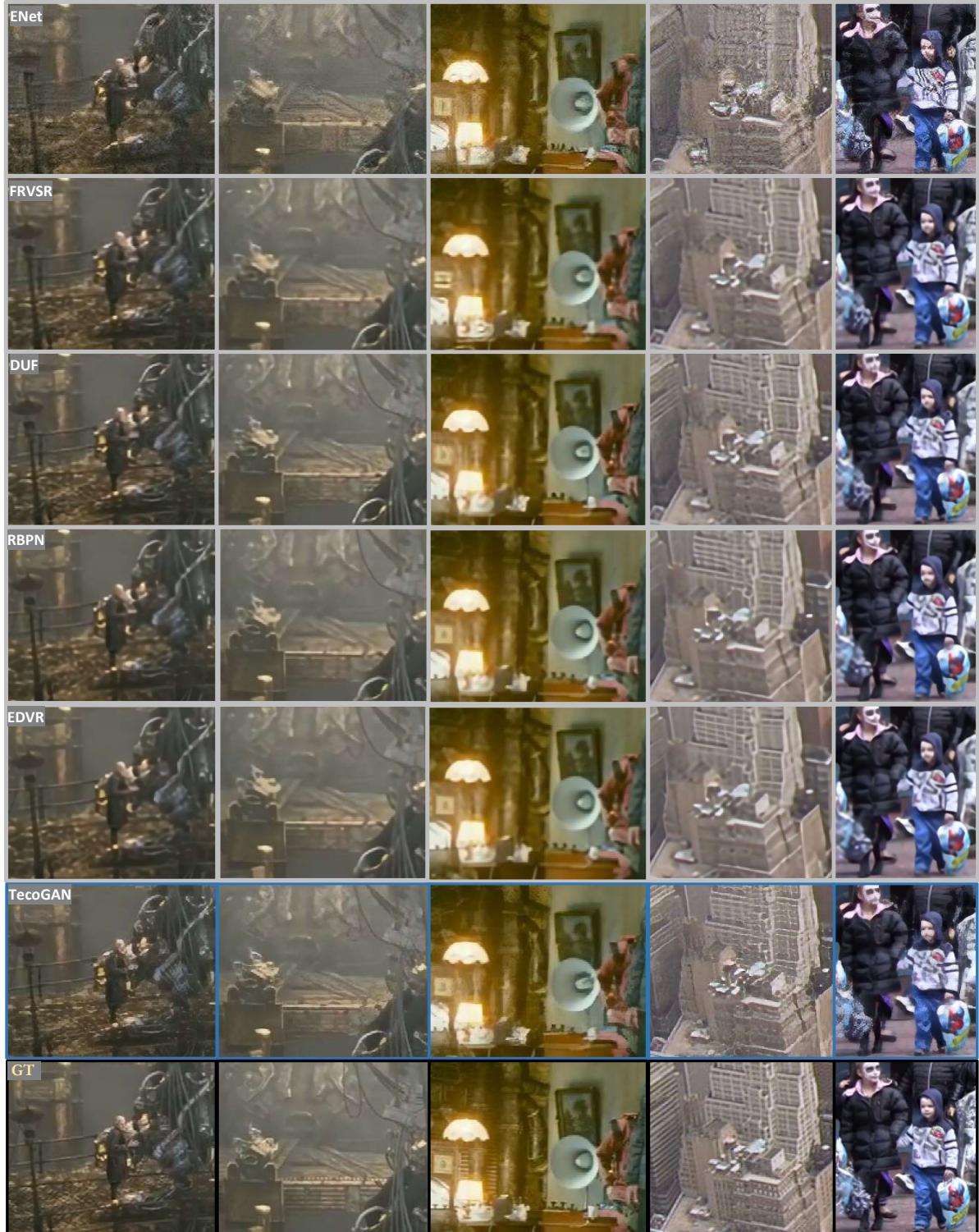


Fig. 12. Detail views of the VSR results of ToS scenes (first three columns) and Vid4 scenes (two right-most columns) generated with different methods: from top to bottom. ENet [Sajjadi et al. 2017], FRVSR [Sajjadi et al. 2018], DUF [Jo et al. 2018], RBPN [Haris et al. 2019], EDVR [Wang et al. 2019a], TecoGAN, and the ground truth. Tears of Steel (ToS) movie (CC) Blender Foundation | mango.blender.org.

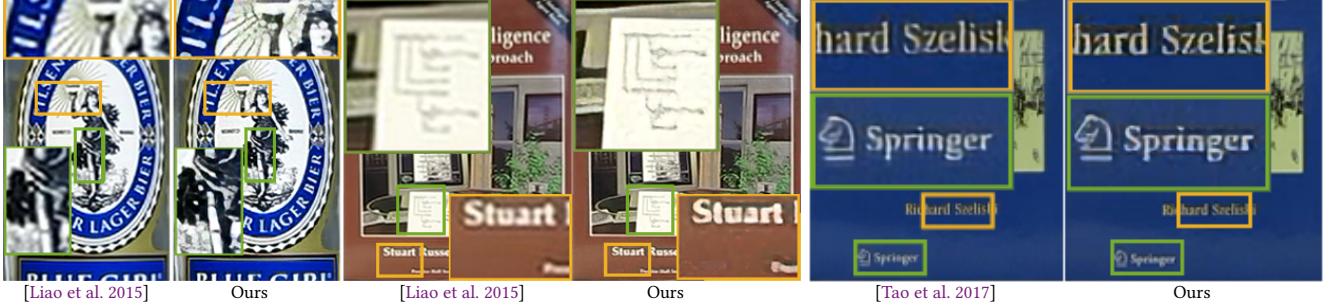


Fig. 13. VSR comparisons for different captured images in order to compare to previous work [Liao et al. 2015; Tao et al. 2017].

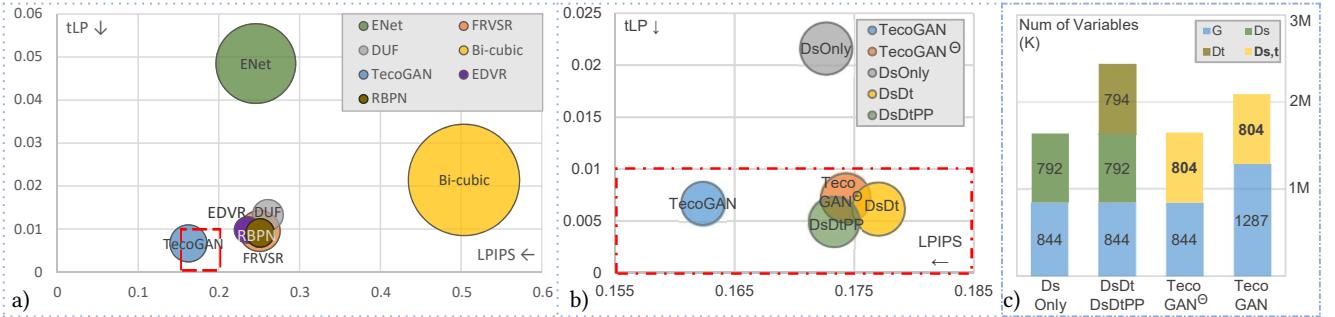


Fig. 14. Visual summary of VSR models. a) LPIPS (x-axis) measures spatial detail and temporal coherence is measured by tLP (y-axis) and tOF (bubble size with smaller as better). b) The red-dashed-box region of a), containing our ablated models. c) The network sizes.

Table 2. Averaged VSR metric evaluations for the Vid4 data set with the following metrics, PSNR: pixel-wise accuracy. LPIPS (AlexNet): perceptual distance to the ground truth. T-diff: pixel-wise differences of warped frames. tOF: pixel-wise distance of estimated motions. tLP: perceptual distance between consecutive frames. User study: Bradley-Terry scores [Bradley and Terry 1952]. Performance is averaged over 500 images up-scaled from 320x134 to 1280x536. More details can be found in Appendix A and Sec. 3 of the supplemental web-page.

Methods	PSNR↑	LPIPS↓ ×10	T-diff↓ ×100	tOF↓ ×10	tLP↓ ×100	User Study ↑	Model Size(M) ↓	Processing Time(ms/frame) ↓
DsOnly	24.14	1.727	6.852	2.157	2.160	-	-	-
DsDt	24.75	1.770	5.071	2.198	0.614	-	-	-
DsDtPP	25.77	1.733	4.369	2.103	0.489	-	-	-
TecoGAN [⊖]	25.89	1.743	4.076	2.082	0.718	-	0.8(G)+1.7(F)	37.07
TecoGAN	25.57	1.623	4.961	1.897	0.668	3.258	1.3(G)+1.7(F)	41.92
ENet	22.31	2.458	9.281	4.009	4.848	1.616	-	-
FRVSR	26.91	2.506	3.648	2.090	0.957	2.600	0.8(SRNet)+1.7(F)	36.95
DUF	27.38	2.607	3.298	1.588	1.329	2.933	6.2	942.21
Bi-cubic	23.66	5.036	3.152	5.578	2.144	0.0	-	-
RBPN	27.15	2.511	-	1.473	0.911	-	12.7	510.90
EDVR	27.34	2.356	-	1.367	0.982	-	20.7	299.71

DUF (which has twice the model size), the LPIPS score of TecoGAN shows an improvement of more than 40%. The other baselines are outperformed by similar margins. Even compared to the large EDVR model using down-sampled inputs without Gaussian blur, TecoGAN still yields a 30% improvement in terms of LPIPS.

While traditional temporal metrics based on vector norm differences of warped frames, e.g. T-diff = $\|g_t - W(g_{t-1}, v_t)\|_1$ [Chen et al. 2017], can be easily deceived by very blurry results, e.g. bi-cubic interpolated ones, we propose to use a tandem of two new metrics, tOF and tLP, to measure the consistence over time. tOF

measures the pixel-wise difference of motions estimated from sequences, and tLP measures perceptual changes over time using deep feature map:

$$\begin{aligned} \text{tOF} &= \|OF(b_{t-1}, b_t) - OF(g_{t-1}, g_t)\|_1 \quad \text{and} \\ \text{tLP} &= \|LP(b_{t-1}, b_t) - LP(g_{t-1}, g_t)\|_1. \end{aligned} \quad (2)$$

OF represents an optical flow estimation with the Farneback [2003] algorithm and LP is the perceptual LPIPS metric. In tLP, the behavior of the reference is also considered, as natural videos exhibit a certain degree of change over time. In conjunction, both pixel-wise

differences and perceptual changes are crucial for quantifying realistic temporal coherence. While they could be combined into a single score, we list both measurements separately, as their relative importance could vary in different application settings.

Our evaluation with these temporal metrics in Table 2 shows that all temporal adversarial models outperform spatial adversarial ones and the full TecoGAN model performs very well: With a large amount of spatial detail, it still achieves good temporal coherence, on par with non-adversarial methods such as DUF, FRVSR, RBPN and EDVR. These results are also visualized in Fig. 14. For VSR, we have confirmed these automated evaluations with several user studies (details in Appendix B). Across all of them, we find that the majority of the participants considered the TecoGAN results to be closest to the ground truth, when comparing to bi-cubic interpolation, ENet, FRVSR and DUF.

For the UVT tasks, where no ground-truth data is available, we can still evaluate tOF and tLP metrics by comparing the motion and the perceptual changes of the output data w.r.t. the ones from the input data, i.e., $tOF = \|OF(a_{t-1}, a_t) - OF(g_{t-1}^{a \rightarrow b}, g_t^{a \rightarrow b})\|_1$ and $tLP = \|LP(a_{t-1}, a_t) - LP(g_{t-1}^{a \rightarrow b}, g_t^{a \rightarrow b})\|_1$. With sharp spatial features and coherent motion, TecoGAN outperforms CycleGAN and RecycleGAN on the Obama&Trump dataset, as shown in Table 3, although it is worth pointing out that tOF is less informative in this case, as the motion in the target domain is not necessarily pixel-wise aligned with the input. While RecycleGAN uses an L2-based cycle loss that leads to undesirable smoothing, Park et al. [2019] propose to use temporal-cycle losses in together with a VGG-based content preserving loss (we will refer to this method as STC-V2V below). While the evaluation of temporal metrics for TecoGAN and STC-V2V is very close, Fig. 8 shows that our results contain sharper spatial details, such as the eyes and eyebrows of Obama as well as the wrinkles of Trump. This is illustrated in Sec. 2.2 of the supplemental web-page. Overall, TecoGAN successfully generates spatial details, on par with CycleGAN. TecoGAN also achieves very good tLP scores thanks to the supervision of temporal coherence, on par with previous work [Bansal et al. 2018; Park et al. 2019], despite inferring outputs with improved spatial complexity.

In line with VSR, a perceptual evaluation by humans in a user study confirms our metric evaluations for the UVT task. The participants consistently prefer TecoGAN results over CycleGAN and RecycleGAN. The corresponding scores are given in the right column of Table 3.

Table 3. For the Obama&Trump dataset, the averaged tLP and tOF evaluations closely correspond to our user studies. The table below summarizes user preferences as Bradley-Terry scores. Details are given in Appendix B and Sec. 3 of the supplemental web-page.

UVT scenes	Trump → Obama		Obama → Trump		AVG		User Studies [†] , ref. to	
	tLP↓	tOF↓	tLP↓	tOF↓	tLP↓	tOF↓	original input	arbitrary target
CycleGAN	0.0176	0.7727	0.0277	1.1841	0.0234	0.9784	0.0	0.0
RecycleGAN	0.0111	0.8705	0.0248	1.1237	0.0179	0.9971	0.994	0.202
STC-V2V	0.0143	0.7846	0.0168	0.927	0.0156	0.8561	-	-
TecoGAN	0.0120	0.6155	0.0191	0.7670	0.0156	0.6913	1.817	0.822

6 DISCUSSION AND LIMITATIONS

In paired as well as unpaired data domains, we have demonstrated that it is possible to learn stable temporal functions with GANs thanks to the proposed discriminator architecture and PP loss. We have shown that this yields coherent and sharp details for VSR problems that go beyond what can be achieved with direct supervision. In UVT, we have shown that our architecture guides the training process to successfully establish the spatio-temporal cycle consistency between two domains. These results are reflected in the proposed metrics and confirmed by user studies.

While our method generates very realistic results for a wide range of natural images, our method can lead to temporally coherent yet sub-optimal details in certain cases such as under-resolved faces and text in VSR, or UVT tasks with strongly different motion between two domains. For the latter case, it would be interesting to apply both our method and motion translation from concurrent work [Chen et al. 2019]. This can make it easier for the generator to learn from our temporal self-supervision. The proposed temporal self-supervision also has potential to improve other tasks such as video in-painting and video colorization. In these multi-modal problems, it is especially important to preserve long-term temporal consistency. For our method, the interplay of the different loss terms in the non-linear training procedure does not provide a guarantee that all goals are fully reached every time. However, we found our method to be stable over a large number of training runs and we anticipate that it will provide a very useful basis for a wide range of generative models for temporal data sets.

ACKNOWLEDGMENTS

This work was supported by the ERC Starting Grant realFlow (StG-2015-637014) and the Humboldt Foundation through the Sofja Kovalevskaja Award. We would like to thank Kiwon Um for helping with the user studies.

REFERENCES

- Aayush Bansal, Shugao Ma, Deva Ramanan, and Yaser Sheikh. 2018. Recycle-GAN: Unsupervised Video Retargeting. In *The European Conference on Computer Vision (ECCV)*.
- Dina Bashkirova, Ben Usman, and Kate Saenko. 2018. Unsupervised video-to-video translation. *arXiv preprint arXiv:1806.03698* (2018).
- Yochai Blau and Tomer Michaeli. 2018. The perception-distortion tradeoff. In *Proc. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, Utah, USA*. 6228–6237.
- Ralph Allan Bradley and Milton E Terry. 1952. Rank analysis of incomplete block designs: I. The method of paired comparisons. *Biometrika* 39, 3/4 (1952), 324–345.
- Andrew Brock, Jeff Donahue, and Karen Simonyan. 2019. Large Scale GAN Training for High Fidelity Natural Image Synthesis. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=B1xsqj09Fm>
- Jose Caballero, Christian Ledig, Andrew P Aitken, Alejandro Acosta, Johannes Totz, Zehan Wang, and Wenzhe Shi. 2017. Real-Time Video Super-Resolution with Spatio-Temporal Networks and Motion Compensation.. In *CVPR*, Vol. 1. 7.
- (CC) Blender Foundation | mango.blender.org. 2011. Tears of Steel. <https://mango.blender.org/>. Online; accessed 15 Nov. 2018.
- Dongdong Chen, Jing Liao, Lu Yuan, Nenghai Yu, and Gang Hua. 2017. Coherent online video style transfer. In *Proc. Intl. Conf. Computer Vision (ICCV)*.
- Yang Chen, Yingwei Pan, Ting Yao, Xinmei Tian, and Tao Mei. 2019. Mocycle-GAN: Unpaired Video-to-Video Translation. *arXiv preprint arXiv:1908.09514* (August 2019).
- Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. 2015. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*. 2758–2766.
- M-L Eckert, Wolfgang Heidrich, and Nils Thuerey. 2018. Coupled fluid density and motion from single views. In *Computer Graphics Forum*, Vol. 37(8). Wiley Online

- Library, 47–58.
- Gunnar Farnebäck. 2003. Two-frame motion estimation based on polynomial expansion. In *Scandinavian conference on Image analysis*. Springer, 363–370.
- Gustav Theodor Fechner and Wilhelm Max Wundt. 1889. *Elemente der Psychophysik: erster Theil*. Breitkopf & Härtel.
- Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in neural information processing systems*. 2672–2680.
- Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. 2017. Improved training of wasserstein gans. In *Advances in neural information processing systems*. 5767–5777.
- Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. 2019. Recurrent back-projection network for video super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3897–3906.
- Mingming He, Dongdong Chen, Jing Liao, Pedro V. Sander, and Lu Yuan. 2018. Deep Exemplar-Based Colorization. *ACM Trans. Graph.* 37, 4, Article 47 (July 2018), 16 pages.
- Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. 2017. Image-To-Image Translation With Conditional Adversarial Networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Ondřej Jamriška, Sárka Sochorová, Ondřej Texler, Michal Lukáč, Jakub Fišer, Jingwan Lu, Eli Shechtman, and Daniel Sýkora. 2019. Stylizing Video by Example. *ACM Transactions on Graphics* 38, 4, Article 107 (2019).
- Younghyun Jo, Seoung Wug Oh, Jaeyeon Kang, and Seon Joo Kim. 2018. Deep Video Super-Resolution Network Using Dynamic Upsampling Filters Without Explicit Motion Compensation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3224–3232.
- Justin Johnson, Alexandre Alahi, and Li Fei-Fei. 2016. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*. Springer, 694–711.
- Anton S Kaplanyan, Anton Sochenov, Thomas Leimkühler, Mikhail Okunev, Todd Goodall, and Gizem Rufo. 2019. DeepFovea: neural reconstruction for foveated rendering and video compression using learned statistics of natural videos. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 1–13.
- Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. 2017. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196* (2017).
- T Kelly, P Guerrero, A Steed, P Wonka, and NJ Mitra. 2018. FrankenGAN: guided detail synthesis for building mass models using style-synchronized GANs. *ACM Transactions on Graphics* 37, 6 (November 2018).
- Byungsoo Kim, Vinicius C. Azevedo, Markus Gross, and Barbara Solenthaler. 2019. Transport-Based Neural Style Transfer for Smoke Simulations. *ACM Trans. Graph.* 38, 6, Article 188 (Nov. 2019), 11 pages.
- Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. 2016. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1646–1654.
- Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. 2017. Deep laplacian pyramid networks for fast and accurate superresolution. In *IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 2. 5.
- Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. 2016. Photo-realistic single image super-resolution using a generative adversarial network. *arXiv:1609.04802* (2016).
- Renjie Liao, Xin Tao, Ruiyu Li, Ziyang Ma, and Jiaya Jia. 2015. Video super-resolution via deep draft-ensemble learning. In *Proceedings of the IEEE International Conference on Computer Vision*. 531–539.
- Ce Liu and Deqing Sun. 2011. A Bayesian approach to adaptive video super resolution. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 209–216.
- Ding Liu, Zhaowen Wang, Yuchen Fan, Xianming Liu, Zhangyang Wang, Shiyu Chang, and Thomas Huang. 2017. Robust video super-resolution with learned temporal dynamics. In *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2526–2534.
- Pengpeng Liu, Michael Lyu, Irwin King, and Jia Xu. 2019. Selfflow: Self-supervised learning of optical flow. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4571–4580.
- Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. 2017. Least squares generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*. 2794–2802.
- Seungjun Nah, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, Radu Timofte, and Kyoung Mu Lee. 2019. NTIRE 2019 Challenge on Video Deblurring and Super-Resolution: Dataset and Study. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.
- Suraj Nair, Mohammad Babaieizadeh, Chelsea Finn, Sergey Levine, and Vikash Kumar. 2018. Time reversal as self-supervision. *arXiv preprint arXiv:1810.01128* (2018).
- Kwanyong Park, Sanghyun Woo, Dahun Kim, Donghyeon Cho, and In So Kweon. 2019. Preserving Semantic and Temporal Consistency for Unpaired Video-to-Video Translation. In *Proceedings of the 27th ACM International Conference on Multimedia (Nice, France) (MM '19)*. Association for Computing Machinery, New York, NY, USA, 1248–1257. <https://doi.org/10.1145/3343031.3350864>
- Eduardo Pérez-Pellitero, Mehdi SM Sajjadi, Michael Hirsch, and Bernhard Schölkopf. 2018. Photorealistic Video Super Resolution. *arXiv preprint arXiv:1807.07930* (2018).
- Ekta Prashnani, Hong Cai, Yasamin Mostofi, and Pradeep Sen. 2018. PieAPP: Perceptual Image-Error Assessment through Pairwise Preference. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1808–1817.
- Manuel Ruder, Alexey Dosovitskiy, and Thomas Brox. 2016. Artistic style transfer for videos. In *German Conference on Pattern Recognition*. Springer, 26–36.
- Mehdi SM Sajjadi, Bernhard Schölkopf, and Michael Hirsch. 2017. Enhancenet: Single image super-resolution through automated texture synthesis. In *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 4501–4510.
- Mehdi SM Sajjadi, Raviteja Vemulapalli, and Matthew Brown. 2018. Frame-Recurrent Video Super-Resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2018)*.
- Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. 2016. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1874–1883.
- Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).
- Vincent Sitzmann, Steven Diamond, Yifan Peng, Xiong Dun, Stephen Boyd, Wolfgang Heidrich, Felix Heide, and Gordon Wetzstein. 2018. End-to-end optimization of optics and image processing for achromatic extended depth of field and super-resolution imaging. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–13.
- Xin Tao, Hongyun Gao, Renjie Liao, Jue Wang, and Jiaya Jia. 2017. Detail-Revealing Deep Video Super-Resolution. In *The IEEE International Conference on Computer Vision (ICCV)*.
- Kiwon Um, Xiangyu Hu, and Nils Thuerey. 2017. Perceptual evaluation of liquid simulation methods. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 143.
- Chaoyue Wang, Chang Xu, Chaochui Wang, and Dacheng Tao. 2018b. Perceptual adversarial networks for image-to-image transformation. *IEEE Transactions on Image Processing* 27, 8 (2018), 4066–4079.
- Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. 2018a. Video-to-Video Synthesis. In *Advances in Neural Information Processing Systems (NIPS)*.
- Xintao Wang, Kelvin C.K. Chan, Ke Yu, Chao Dong, and Chen Change Loy. 2019a. EDVR: Video restoration with enhanced deformable convolutional networks. In *The IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*.
- Xiaolong Wang, Allan Jabri, and Alexei A Efros. 2019b. Learning correspondence from the cycle-consistency of time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2566–2576.
- Bartłomiej Wronski, Ignacio Garcia-Dorado, Manfred Ernst, Damien Kelly, Michael Krainin, Chia-Kai Liang, Marc Levoy, and Peyman Milanfar. 2019. Handheld Multi-Frame Super-Resolution. *ACM Trans. Graph.* 38, 4, Article 28 (July 2019), 18 pages.
- Zhijie Wu, Xiang Wang, Di Lin, Dani Lischinski, Daniel Cohen-Or, and Hui Huang. 2019. SAGNet: Structure-aware Generative Network for 3D-Shape Modeling. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2019)* 38, 4 (2019), 91:1–91:14.
- You Xie, Erik Franz, Mengyu Chu, and Nils Thuerey. 2018. tempoGAN: A Temporally Coherent, Volumetric GAN for Super-resolution Fluid Flow. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 95.
- Bo Zhang, Mingming He, Jing Liao, Pedro V Sander, Lu Yuan, Amine Bermak, and Dong Chen. 2019. Deep Exemplar-based Video Colorization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 8052–8061.
- Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. *arXiv preprint* (2018).
- Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*. 2223–2232.
- Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. 2019. Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 9308–9316.

LEARNING TEMPORAL COHERENCE VIA SELF-SUPERVISION FOR GAN-BASED VIDEO GENERATION

THE APPENDIX

In the following, we first give details of the proposed temporal evaluation metrics, and present the corresponding quantitative comparison of our method versus a range of others (Appendix A). The user studies we conducted are in support of our TecoGAN network and proposed temporal metrics, and explained in Appendix B. Then, we give technical details of our spatio-temporal discriminator (Appendix C) and the proposed PP loss (Appendix D). Details of network architectures and training parameters are listed (Appendix E, Appendix F). Lastly, we discuss the performance of our approach in Appendix G.

A METRICS AND QUANTITATIVE ANALYSIS

A.1 Spatial Metrics

In order to be able to compare our results with single-image methods, we evaluate all VSR methods with the purely spatial metrics PSNR together with the human-calibrated LPIPS metric [Zhang et al. 2018]. While higher PSNR values indicate a better pixel-wise accuracy, lower LPIPS values represent better perceptual quality and closer semantic similarity. Note that both metrics are agnostic to changes over time, and hence do not suffice to really evaluate video data.

Mean values of the Vid4 scenes [Liu and Sun 2011] are shown on the top of Table 4. Trained with direct vector norms losses, FRVSR, DUF, EDVR, and RBPN achieve high PSNR scores. However, the undesirable smoothing induced by these losses manifests themselves in larger LPIPS distances. ENet, on the other hand, with no information from neighboring frames, yields the lowest PSNR and achieves an LPIPS score that is only slightly better than DUF and FRVSR. The TecoGAN model with adversarial training achieves an excellent LPIPS score, with a PSNR decrease of less than 2dB over DUF. This is very reasonable, since PSNR and perceptual quality were shown to be anti-correlated [Blau and Michaeli 2018], especially in regions where PSNR is very high. Based on good perceptual quality and reasonable pixel-wise accuracy, TecoGAN outperforms all other methods by more than 30% for LPIPS.

A.2 Temporal Metrics

For both VSR and UVT, evaluating temporal coherence without ground-truth motion is very challenging. L^2 -based temporal metrics such as $T\text{-}diff = \|g_t - W(g_{t-1}, v_t)\|_1$ was used [Chen et al. 2017] as a rough assessment of temporal differences, and we give corresponding numbers for comparison. As shown at the bottom of Table 4, T-diff, due to its local nature, is easily deceived by blurry method such as the bi-cubic interrelation and can not correlate well with visual assessments of coherence.

By using the proposed metrics, i.e. measuring the pixel-wise motion differences using tOF together with the perceptual changes over time using tLP, a more nuanced evaluation can be achieved, as

Table 4. Metrics evaluated for the VSR Vid4 scenes.

	BIC	ENet	FRVSR	DUF	RBPN	EDVR	TecoGAN	^{Teco} _{GAN} ⊕	DsOnly	DsDt	DsDtPP
calendar	20.27	19.85	23.86	24.07	23.88	23.97	23.21	23.35	22.23	22.76	22.95
foliage	23.57	21.15	26.35	26.45	26.32	26.42	24.26	25.13	22.33	22.73	25.00
city	24.82	23.36	27.71	28.25	27.51	27.76	26.78	26.94	25.86	26.52	27.03
walk	25.84	24.90	29.56	30.58	30.59	30.92	28.11	28.14	26.49	27.37	28.14
average	23.66	22.31	26.91	27.38	27.15	27.34	25.57	25.89	24.14	24.75	25.77
LPIPS ↓×10	BIC	ENet	FRVSR	DUF	RBPN	EDVR	TecoGAN	^{Teco} _{GAN} ⊕	DsOnly	DsDt	DsDtPP
calendar	5.935	2.191	2.989	3.086	2.654	2.296	1.511	2.142	1.532	2.111	2.112
foliage	5.338	2.663	3.242	3.492	3.613	3.485	1.902	1.984	2.113	2.092	1.902
city	5.451	3.431	2.429	2.447	0.233	2.264	2.084	1.940	2.120	1.889	1.989
walk	3.655	1.794	1.374	1.380	1.362	1.291	1.106	1.011	1.215	1.057	1.051
average	5.036	2.458	2.506	2.607	2.511	2.356	1.623	1.743	1.727	1.770	1.733
tOF ↓×10	BIC	ENet	FRVSR	DUF	RBPN	EDVR	TecoGAN	^{Teco} _{GAN} ⊕	DsOnly	DsDt	DsDtPP
calendar	4.956	3.450	1.537	1.134	1.068	0.986	1.342	1.403	1.607	1.683	1.583
foliage	4.922	3.775	1.489	1.356	1.234	1.144	1.238	1.444	1.543	1.562	1.373
city	7.967	6.225	2.992	1.724	1.584	1.446	2.612	2.905	2.920	2.936	3.062
walk	5.150	3.203	2.569	2.127	1.994	1.871	2.571	2.765	2.745	2.796	2.649
average	5.578	4.009	2.090	1.588	1.473	1.367	1.897	2.082	2.157	2.198	2.103
tLP ↓×100	BIC	ENet	FRVSR	DUF	RBPN	EDVR	TecoGAN	^{Teco} _{GAN} ⊕	DsOnly	DsDt	DsDtPP
calendar	3.258	2.957	1.067	1.603	0.802	0.622	0.165	1.087	0.872	0.764	0.670
foliage	2.434	6.372	1.644	2.034	1.927	1.998	0.894	0.740	3.422	0.493	0.454
city	2.193	7.953	0.752	1.399	0.432	1.060	0.974	0.347	2.660	0.490	0.140
walk	0.851	2.729	0.286	0.307	0.271	0.171	0.653	0.635	1.596	0.697	0.613
average	3.152	4.848	0.957	1.329	0.911	0.982	0.668	0.718	2.160	0.614	0.489
T-diff ↓×100	BIC	ENet	FRVSR	DUF	TecoGAN	TecoGAN [⊕]		DsOnly	DsDt	DsDtPP	GT
calendar	2.271	9.153	3.212	2.750	4.663	3.496	6.287	4.347	4.167	6.478	
foliage	3.745	11.997	3.478	3.115	5.674	4.179	8.961	6.068	4.548	4.396	
city	1.974	7.788	2.452	2.244	3.528	2.965	4.929	3.525	2.991	4.282	
walk	4.101	7.576	5.028	4.687	5.460	5.234	6.454	5.714	5.305	5.525	
average	3.152	9.281	3.648	3.298	4.961	4.076	6.852	5.071	4.369	5.184	

Table 5. Metrics evaluated for VSR of ToS scenes.

	BIC	ENet	FRVSR	DUF	RBPN	EDVR	TecoGAN
room	26.90	25.22	29.80	30.85	26.82	31.12	29.31
bridge	28.34	26.40	32.56	33.02	28.56	32.88	30.81
face	33.75	32.17	39.94	40.23	33.74	41.57	38.60
average	29.58	27.82	34.04	34.60	29.71	35.02	32.75
LPIPS ↓×10	BIC	ENet	FRVSR	DUF	RBPN	EDVR	TecoGAN
room	5.167	2.427	1.917	1.987	2.054	2.232	1.358
bridge	4.897	2.807	1.761	1.684	1.845	1.663	1.263
face	2.241	1.784	0.586	0.517	0.728	0.613	0.590
average	4.169	2.395	1.449	1.414	1.565	1.501	1.086
tOF ↓×10	BIC	ENet	FRVSR	DUF	RBPN	EDVR	TecoGAN
room	1.735	1.625	0.861	0.901	0.821	0.769	0.737
bridge	5.485	4.037	1.614	1.348	1.354	1.140	1.492
face	4.302	2.255	1.782	1.577	1.612	1.383	1.667
average	4.110	2.845	1.460	1.296	1.287	1.113	1.340
tLP ↓×100	BIC	ENet	FRVSR	DUF	RBPN	EDVR	TecoGAN
room	1.320	2.491	0.366	0.307	0.263	0.252	0.590
bridge	2.237	6.241	0.821	0.526	0.821	1.030	0.912
face	1.270	1.613	0.290	0.314	0.364	0.396	0.379
average	1.696	3.827	0.537	0.403	0.531	0.628	0.664

shown for the VSR task in the middle of Table 4. Not surprisingly, the results of ENet show larger errors for all metrics due to their strongly flickering content. Bi-cubic up-sampling, DUF, and FRVSR achieve very low T-diff errors due to their smooth results, representing an easy, but undesirable avenue for achieving coherency. However, the overly smooth changes of the former two are identified by the tLP scores. While our DsOnly model generates sharper results at the expense of temporal coherence, it still outperforms

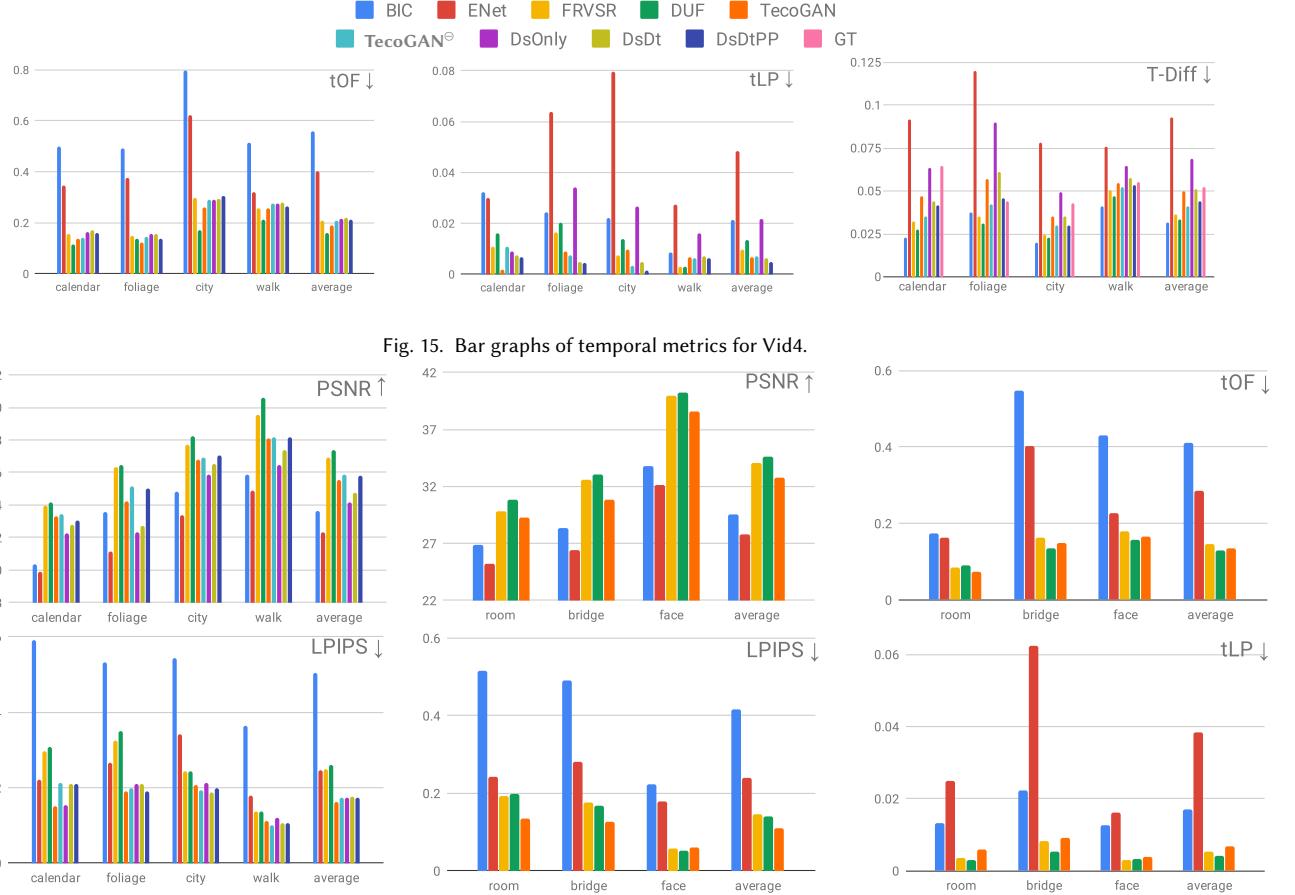


Fig. 16. Spatial metrics for Vid4.

Fig. 17. Metrics for ToS.

ENet there. By adding temporal information to discriminators, our DsDt, DsDt+PP, TecoGAN^θ and TecoGAN improve in terms of temporal metrics. Especially the full TecoGAN model stands out here. For the UVT tasks, temporal motions are evaluated by comparing to the input sequence. With sharp spatial features and coherent motion, TecoGAN outperforms previous work on the Obama&Trump dataset, as shown in Table 3.

A.3 Spatio-temporal Evaluations

Since temporal metrics can trivially be reduced for blurry image content, we found it important to evaluate results with a combination of spatial and temporal metrics. Given that perceptual metrics are already widely used for image evaluations, we believe it is the right time to consider perceptual changes in temporal evaluations, as we did with our proposed temporal coherence metrics. Although not perfect, they are not as easily deceived as simpler metrics. Specifically, tOF is more robust than a direct pixel-wise metric as it compares motions instead of image content. In the supplemental material (Sec. 6), we visualize the motion difference and it can well reflect the visual inconsistencies. On the other hand, we found that our formulation of tLP is a general concept that can work reliably with different perceptual metrics: When repeating

the tLP evaluation with the PieAPP metric [Prashnani et al. 2018] instead of LP, i.e., $tLP_{Pie} = \|f(b_{t-1}, b_t) - f(g_{t-1}, g_t)\|_1$, where $f(\cdot)$ indicates the perceptual error function of PieAPP, we get close to identical results, as shown in Fig. 18. The conclusions from tLP_{Pie} also closely match the LPIPS-based evaluation: our network architecture can generate realistic and temporally coherent detail, and the metrics we propose allow for a stable, automated evaluation of the temporal perception of a generated video sequence.

Besides the previously evaluated the Vid4 dataset, with graphs shown in Fig. 15, 16, we also get similar evaluation results on the *Tears of Steel* data-sets (room, bridge, and face, in the following referred to as ToS scenes) and corresponding results are shown in Table 5 and Fig. 17. In all tests, we follow the procedures of previous work [Jo et al. 2018; Sajjadi et al. 2018] to make the outputs of all methods comparable, i.e., for all result images, we first exclude spatial borders with a distance of 8 pixels to the image sides, then further shrink borders such that the LR input image is divisible by 8 and for spatial metrics, we ignore the first two and the last two frames, while for temporal metrics, we ignore first three and last two frames, as an additional previous frame is required for inference. Below, we additionally show user study results for the Vid4 scenes. By comparing the user study results and the metric breakdowns

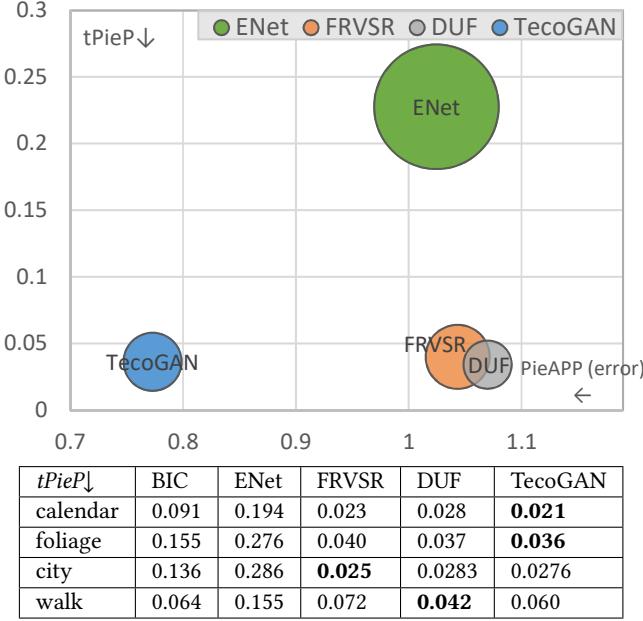


Fig. 18. Tables and visualization of perceptual metrics computed with PieAPP [Prashnani et al. 2018] (instead of LPIPS used in Fig. 14 previously) on ENet, FRVSR, DUF and TecoGAN for the VSR of Vid4. Bubble size indicates the tOF score.

shown in Table 4, we found our metrics to reliably capture the human temporal perception.

B USER STUDIES

We conducted several user studies for the VSR task comparing five different methods: bi-cubic interpolation, ENet, FRVSR, DUF and TecoGAN. The established 2AFC design [Fechner and Wundt 1889; Um et al. 2017] is applied, i.e., participants have a pair-wise choice, with the ground-truth video shown as reference. One example setup can be seen in Fig. 19. The videos are synchronized and looped until participants make the final decision. With no control to stop videos, users cannot stop or influence the playback, and hence can focus more on the whole video, instead of specific spatial details. Videos positions (left/A or right/B) are randomized.

After collecting 1000 votes from 50 users for every scene, i.e. twice for all possible pairs ($5 \times 4/2 = 10$ pairs), we follow common procedure and compute scores for all models with the Bradley-Terry model (1952). The outcomes for the Vid4 scenes can be seen in Fig. 20 (overall scores are listed in Table 2 of the main document).

From the Bradley-Terry scores for the Vid4 scenes we can see that the TecoGAN model performs very well, and achieves the first place in three cases, as well as a second place in the walk scene. The latter is most likely caused by the overall slightly smoother images of the walk scene, in conjunction with the presence of several human faces, where our model can lead to the generation of unexpected details. However, overall the user study shows that users preferred the TecoGAN output over the other two deep-learning methods with a 63.5% probability.

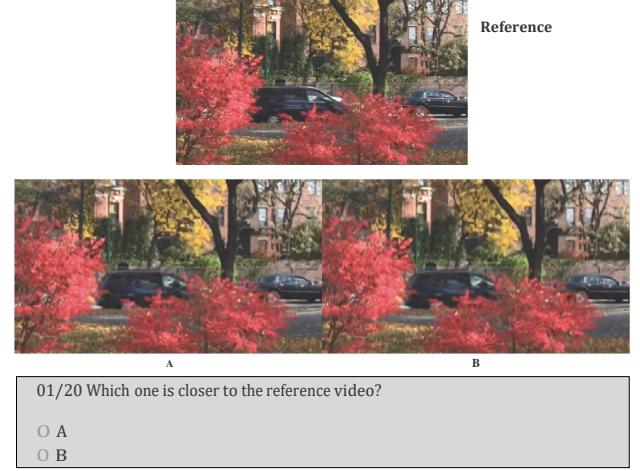


Fig. 19. A sample setup of user study.

Methods	The Bradley-Terry scores (standard error)			
	calendar	foliage	city	walk
Bi-cubic	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
ENet	1.834 (0.228)	1.634 (0.180)	1.282 (0.205)	1.773 (0.197)
FRVSR	3.043 (0.246)	2.177 (0.186)	3.173 (0.240)	2.424 (0.204)
DUF	3.468 (0.252)	2.243 (0.186)	3.302 (0.242)	3.175 (0.214)
TecoGAN	4.091 (0.262)	2.769 (0.194)	4.052 (0.255)	2.693 (0.207)

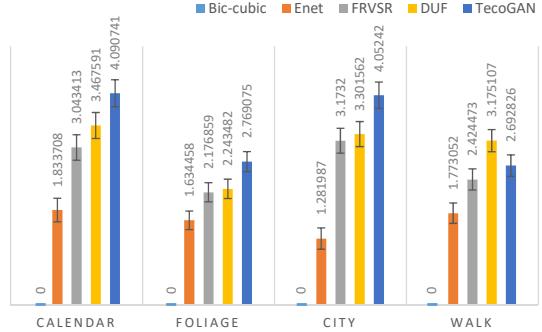


Fig. 20. Tables and bar graphs of Bradley-Terry scores and standard errors for Vid4 VSR.

This result also matches with our metric evaluations. In Table 4, while TecoGAN achieves spatial (LPIPS) improvements in all scenes, DUF and FRVSR are not far behind in the walk scene. In terms of temporal metrics tOF and tLP, TecoGAN achieves similar or lower scores compared to FRVSR and DUF for calendar, foliage and city scenes. The lower performance of our model for the walk scene is likewise captured by higher tOF and tLP scores. Overall, the metrics confirm the performance of our TecoGAN approach and match the results of the user studies, which indicate that our proposed temporal metrics successfully capture important temporal aspects of human perception.

For UVT tasks which have no ground-truth data, we carried out two sets of user studies: One uses an arbitrary sample from the target domain as the reference and the other uses the actual input from the source domain as the reference. On the Obama&Trump data-sets, we evaluate results from CycleGAN, RecycleGAN, and TecoGAN following the same modality, i.e. a 2AFC design with 50 users for each

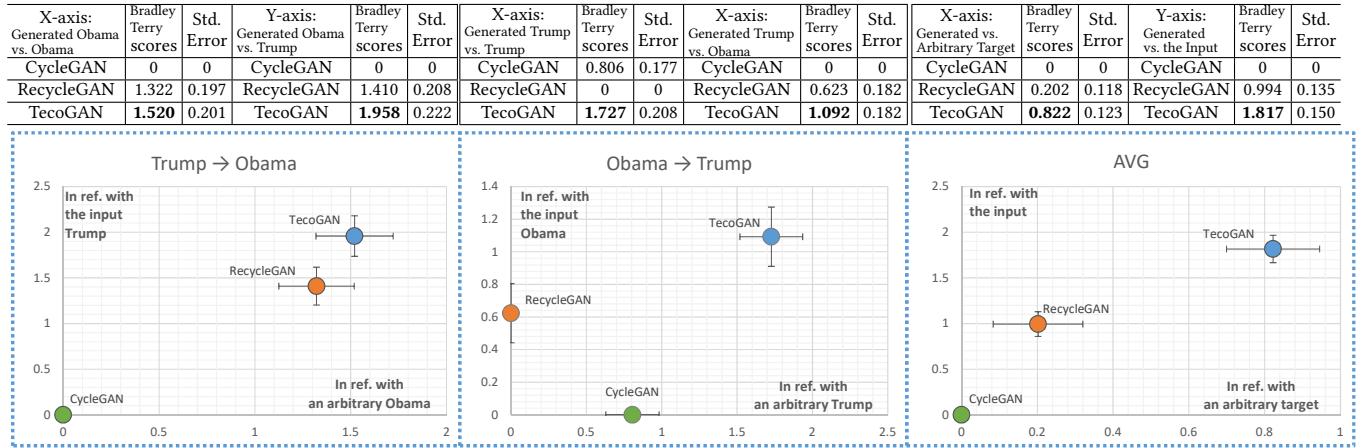


Fig. 21. Tables and graphs of Bradley-Terry scores and standard errors for Obama&Trump UVT.

run. E.g., on the left of Fig. 21, users evaluate the generated Obama in reference with the input Trump on the y-axis, while an arbitrary Obama video is shown as the reference on the x-axis. Ultimately, the y-axis is more important than the x-axis as it indicates whether the translated result preserves the original expression. A consistent ranking of TecoGAN > RecycleGAN > CycleGAN is shown on the y-axis with clear separations, i.e. standard errors don't overlap. The x-axis indicates whether the inferred result matches the general spatio-temporal content of the target domain. Our TecoGAN model also receives the highest scores here, although the responses are slightly more spread out. On the right of Fig. 21, we summarize both studies in a single graph highlighting that the TecoGAN model is consistently preferred by the participants of our user studies.

C TECHNICAL DETAILS OF THE SPATIO-TEMPORAL DISCRIMINATOR

C.1 Motion Compensation Used in Warped Triplet

In the TecoGAN architecture, $D_{s,t}$ detects the temporal relationships between I_{wg} and I_{wb} with the help of the flow estimation network F. However, at the boundary of images, the output of F is usually less accurate due to the lack of reliable neighborhood information. There is a higher chance that objects move into the field of view, or leave suddenly, which significantly affects the images warped with the inferred motion. An example is shown in Fig. 22. This increases the difficulty for $D_{s,t}$, as it cannot fully rely on the images being aligned via warping. To alleviate this problem, we only use the center region of I_{wg} and I_{wb} as the discriminator inputs and we reset a boundary of 16 pixels. Thus, for an input resolution of I_{wg} and I_{wb} of 128×128 for the VSR task, the inner part in size of 96×96 is left untouched, while the border regions are overwritten with zeros.

The flow estimation network F with the loss $\mathcal{L}_{G,F}$ should only be trained to support G in reaching the output quality as determined by $D_{s,t}$, but not the other way around. The latter could lead to F networks that confuse $D_{s,t}$ with strong distortions of I_{wg} and I_{wb} . In order to avoid this undesirable case, we stop the gradient back propagation from I_{wg} and I_{wb} to F. In this way, gradients from $D_{s,t}$ to F are only back propagated through the generated samples g_{t-1}, g_t and g_{t+1} into the generator network. As such, $D_{s,t}$ can guide G to improve the image content, and F learns to warp the previous frame in accordance with the detail that G can synthesize. However, F does not adjust the motion estimation only to reduce the adversarial loss.

C.2 Curriculum Learning for UVT Discriminators

As mentioned in the main document, we train the UVT $D_{s,t}$ with 100% spatial triplets at the very beginning. During training, 25% of them gradually transition to warped triplets and another 25% transition to original triplets. The transitions of the warped triplets are computed with linear interpolation: $(1 - \alpha)I_{cg} + \alpha I_{wg}$, with α

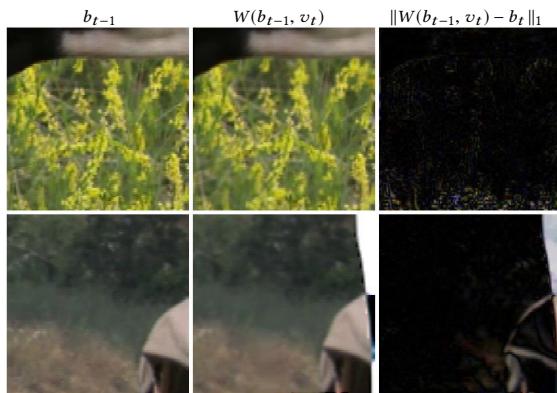


Fig. 22. Near image boundaries, flow estimation is less accurate and warping often fails to align content. The first two columns show original and warped frames, the third one shows differences after warping (ideally all black). The top row shows that structures moving into the view can cause problems, visible at the bottom of the images. The second row has objects moving out of the view.

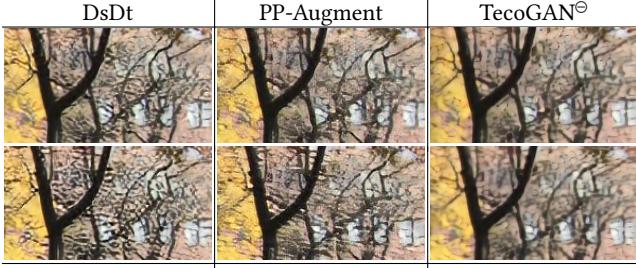


Fig. 23. 1st & 2nd row: Frame 15 & 40 of the *Foliage* scene. While DsDt leads to strong recurrent artifacts early on, PP-Augment shows similar artifacts later in time (2nd row, middle). TecoGAN^O model successfully removes these artifacts.

growing from 0 to 1. For the original triplets, we additionally fade the “warping” operation out by using $(1 - \alpha)I_{cg} + \alpha\{W(g_{t-1}, v_t * \beta), g_t, W(g_{t+1}, v'_t * \beta)\}$, again with α growing from 0 to 1 and β decreasing from 1 to 0. We found this smooth transition to be helpful for a stable training run.

D DATA AUGMENTATION AND TEMPORAL CONSTRAINTS IN THE PP LOSS

Since training with sequences of arbitrary length is not possible with current hardware, problems such as the “streaking artifacts” discussed above generally arise for recurrent models. In the proposed PP loss, both the Ping-Pong data augmentation and the temporal consistency constraint contribute to solving these problems. In order to show their separated contributions, we trained another TecoGAN variant that only employs the data augmentation without the constraint (i.e., $\lambda_p = 0$ in Table 1). Denoted as “PP-Augment”, we show its results in comparison with the DsDt and TecoGAN^O models in Fig. 23. Video results are shown in the in the supplemental material (Sec. 4.5.).

During training, the generator of DsDt receives 10 frames, and generators of PP-Augment and TecoGAN^O see 19 frames. While DsDt shows strong recurrent accumulation artifacts early on, the PP-Augment version slightly reduces the artifacts. In Fig. 23, it works well for frame 15, but shows artifacts from frame 32 on. Only our regular model (TecoGAN^O) successfully avoids temporal accumulation for all 40 frames. Hence, with the PP constraint, the model avoids recurrent accumulation of artifacts and works well for sequences that are substantially longer than the training length. Among others, we have tested our model with ToS sequences of lengths 150, 166 and 233. For all of these sequences, the TecoGAN model successfully avoids temporal accumulation or streaking artifacts.

E NETWORK ARCHITECTURE

In this section, we use the following notation to specify all network architectures used: $\text{conc}()$ represents the concatenation of two tensors along the channel dimension; $C/CT(\text{input}, \text{kernel_size}, \text{output_channel}, \text{stride_size})$ stands for the convolution and transposed convolution operation, respectively; “+” denotes element-wise addition; BilinearUp2 up-samples input tensors by a factor of 2 using bi-linear interpolation; $\text{BicubicResize4}(\text{input})$ increases the resolution of the input tensor to 4 times higher via bi-cubic up-sampling;

$\text{Dense}(\text{input}, \text{output_size})$ is a densely-connected layer, which uses Xavier initialization for the kernel weights.

The architecture of our VSR generator G is:

$$\text{conc}(a_t, W(g_{t-1}, v_t)) \rightarrow l_{in}; C(l_{in}, 3, 64, 1), \text{ReLU} \rightarrow l_0;$$

$$\text{ResidualBlock}(l_i) \rightarrow l_{i+1} \text{ with } i = 0, \dots, n-1;$$

$$CT(l_n, 3, 64, 2), \text{ReLU} \rightarrow l_{up2}; CT(l_{up2}, 3, 64, 2), \text{ReLU} \rightarrow l_{up4};$$

$$C(l_{up4}, 3, 3, 1), \text{ReLU} \rightarrow l_{res}; \text{BicubicResize4}(a_t) + l_{res} \rightarrow g_t .$$

In TecoGAN^O, there are 10 sequential residual blocks in the generator ($l_n = l_{10}$), while the TecoGAN generator has 16 residual blocks ($l_n = l_{16}$). Each $\text{ResidualBlock}(l_i)$ contains the following operations: $C(l_i, 3, 64, 1)$, $\text{ReLU} \rightarrow r_i$; $C(r_i, 3, 64, 1) + l_i \rightarrow l_{i+1}$.

The VSR $D_{s,t}$ ’s architecture is:

$$\begin{aligned} I_{s,t}^g \text{ or } I_{s,t}^b \rightarrow l_{in}; C(l_{in}, 3, 64, 1), \text{Leaky ReLU} \rightarrow l_0; \\ C(l_0, 4, 64, 2), \text{BatchNorm}, \text{Leaky ReLU} \rightarrow l_1; \\ C(l_1, 4, 64, 2), \text{BatchNorm}, \text{Leaky ReLU} \rightarrow l_2; \\ C(l_2, 4, 128, 2), \text{BatchNorm}, \text{Leaky ReLU} \rightarrow l_3; \\ C(l_3, 4, 256, 2), \text{BatchNorm}, \text{Leaky ReLU} \rightarrow l_4; \\ \text{Dense}(l_4, 1), \text{sigmoid} \rightarrow l_{out} . \end{aligned}$$

VSR discriminators used in our variant models, DsDt, DsDtPP and DsOnly, have a the same architecture as $D_{s,t}$. They only differ in terms of their inputs.

The flow estimation network F has the following architecture:

$$\begin{aligned} \text{conc}(a_t, a_{t-1}) \rightarrow l_{in}; C(l_{in}, 3, 32, 1), \text{Leaky ReLU} \rightarrow l_0; \\ C(l_0, 3, 32, 1), \text{Leaky ReLU}, \text{MaxPooling} \rightarrow l_1; \\ C(l_1, 3, 64, 1), \text{Leaky ReLU} \rightarrow l_2; \\ C(l_2, 3, 64, 1), \text{Leaky ReLU}, \text{MaxPooling} \rightarrow l_3; \\ C(l_3, 3, 128, 1), \text{Leaky ReLU} \rightarrow l_4; \\ C(l_4, 3, 128, 1), \text{Leaky ReLU}, \text{MaxPooling} \rightarrow l_5; \\ C(l_5, 3, 256, 1), \text{Leaky ReLU} \rightarrow l_6; \\ C(l_6, 3, 256, 1), \text{Leaky ReLU}, \text{BilinearUp2} \rightarrow l_7; \\ C(l_7, 3, 128, 1), \text{Leaky ReLU} \rightarrow l_8; \\ C(l_8, 3, 128, 1), \text{Leaky ReLU}, \text{BilinearUp2} \rightarrow l_9; \\ C(l_9, 3, 64, 1), \text{Leaky ReLU} \rightarrow l_{10}; \\ C(l_{10}, 3, 64, 1), \text{Leaky ReLU}, \text{BilinearUp2} \rightarrow l_{11}; \\ C(l_{11}, 3, 32, 1), \text{Leaky ReLU} \rightarrow l_{12}; \\ C(l_{12}, 3, 2, 1), \tanh \rightarrow l_{out}; l_{out} * \text{MaxVel} \rightarrow v_t . \end{aligned}$$

Here, MaxVel is a constant vector, which scales the network output to the normal velocity range.

While F is the same for UVT tasks, UVT generators have an encoder-decoder structure:

$$\begin{aligned} \text{conc}(a_t, W(g_{t-1}, v_t)) \rightarrow l_{in}; C(l_{in}, 7, 32, 1), \text{InstanceNorm}, \text{ReLU} \rightarrow l_0; \\ C(l_0, 3, 64, 2), \text{InstanceNorm}, \text{ReLU} \rightarrow l_1; \\ C(l_1, 3, 128, 2), \text{InstanceNorm}, \text{ReLU} \rightarrow l_2; \\ \text{ResidualBlock}(l_2 + i) \rightarrow l_{3+i} \text{ with } i = 0, \dots, n-1; \\ CT(l_{n+2}, 3, 64, 2), \text{InstanceNorm}, \text{ReLU} \rightarrow l_{n+3}; \\ CT(l_{n+3}, 3, 32, 2), \text{InstanceNorm}, \text{ReLU} \rightarrow l_{n+4}; \\ CT(l_{n+4}, 7, 3, 1), \tanh \rightarrow l_{out} \end{aligned}$$

$\text{ResidualBlock}(l_2 + i)$ contains the following operations: $C(l_{2+i}, 3, 128, 1)$, $\text{InstanceNorm}, \text{ReLU} \rightarrow t_{2+i}$; $C(t_{2+i}, 3, 128, 1)$, $\text{InstanceNorm} \rightarrow r_{2+i}$; $r_{2+i} + l_{2+i} \rightarrow l_{3+i}$. We use 10 residual blocks for all UVT generators.

Since UVT generators are larger than the VSR generator, we also use a larger $D_{s,t}$ architecture:

$$I_{s,t}^g \text{ or } I_{s,t}^b \rightarrow l_{in}; C(l_{in}, 4, 64, 24), \text{ReLU} \rightarrow l_0;$$

$$\begin{aligned}
& C(l_0, 4, 128, 2), \text{InstanceNorm, Leaky ReLU} \rightarrow l_1; \\
& C(l_1, 4, 256, 2), \text{InstanceNorm, Leaky ReLU} \rightarrow l_2; \\
C(l_2, 4, 512, 2), \text{InstanceNorm, Leaky ReLU} & \rightarrow l_3; \text{Dense}(l_3, 1) \rightarrow l_{out}.
\end{aligned}$$

Again, all ablation studies use the same architecture and only differ in terms of their inputs.

F TRAINING DETAILS

We use the non-saturated GAN for VSR and LSGAN [Mao et al. 2017] for UVT and both of them can prevent the gradient vanishing problem of a standard GAN [Goodfellow et al. 2014]. We employ a dynamic discriminator updating strategy, i.e. discriminators are not updated when there is a large difference between $D(I_{s,t}^g)$ and $D(I_{s,t}^b)$. While our training runs are generally very stable, the training process could potentially be further improved with modern GAN algorithms, e.g. Wasserstein GAN [Gulrajani et al. 2017].

To improve the stability of the adversarial training for the VSR task, we pre-train G and F together with a simple L^2 loss of $\sum \|g_t - b_t\|_2 + \lambda_w \mathcal{L}_{warp}$ for 500k batches. Based on the pre-trained models, we use 900k batches for the proposed spatio-temporal adversarial training stage. Our training sequences has a length of 10 and a batch size of 4. A black image is used as the first previous frame of each video sequence. I.e., one batch contains 40 frames and with the PP loss formulation, the NN receives gradients from 76 frames in total for every training iteration.

In the pre-training stage of VSR, we train the F and a generator with 10 residual blocks. An ADAM optimizer with $\beta = 0.9$ is used throughout. The learning rate starts from 10^{-4} and decays by 50% every 50k batches until it reaches 2.5×10^{-5} . This pre-trained model is then used for all TecoGAN variants as initial state. In the adversarial training stage of VSR, all TecoGAN variants are trained with a fixed learning rate of 5×10^{-5} . The generators in DsOnly, DsDt, DsDtPP and TecoGAN[⊖] have 10 residual blocks, whereas the TecoGAN model has 6 additional residual blocks in its generator. Therefore, after loading 10 residual blocks from the pre-trained model, these additional residual blocks are faded in smoothly with a factor of 2.5×10^{-5} . We found this growing training methodology [Karras et al. 2017], to be stable and efficient in our tests. When training the VSR DsDt and DsDtPP, extra parameters are used to balance the two cooperating discriminators properly. Through experiments, we found D_t to be stronger. Therefore, we reduce the learning rate of D_t to 1.5×10^{-5} in order to keep both discriminators balanced. At the same time, a factor of 0.0003 is used on the temporal adversarial loss to the generator, while the spatial adversarial loss has a factor of 0.001. During the VSR training, input LR video frames are cropped to a size of 32×32 . In all VSR models, the Leaky ReLU operation uses a tangent of 0.2 for the negative half space. Additional training parameters are listed in Table 6.

For the UVT task, a pre-training is not necessary for generators and discriminators since temporal triplets are gradually faded in. Only a pre-trained F model is reused. Trained on specialized datasets, we found UVT models to converge well with 100k batches of sequences in length of 6 and batch size of 1.

For all UVT tasks, we use a learning rate of 10^{-4} to train the first 90k batches and the last 10k batches are trained with the learning rate decay from 10^{-4} to 0. Images of the input domain are cropped

Table 6. Training parameters

VSR Param	DsOnly	DsDt	DsDtPP	TecoGAN [⊖]	TecoGAN
λ_a	1e-3	Ds: 1e-3, Dt: 3e-4		1e-3	1e-3
λ_p	0.0	0.0		0.5	
λ_ϕ			0.2 for VGG and 1.0 for Discriminator		
$\lambda_\omega, \lambda_c$			1.0, 1.0		
learning-rate	5e-5	5e-5 for Ds, 1.5e-5 for Dt, 5e-5 for G, F.		5e-5	5e-5
UVT Param	DsOnly	Dst	DsDtPP	TecoGAN	
λ_a		0.5		Ds: 0.5 Dt: 0.3	0.5
λ_p	0.0	0.0		100.0	
λ_ϕ			from 10^6 decays to 0.0		
λ_ω	0.0, a pre-trained F is used for UST tasks				
λ_c			10.0		

into a size of 256×256 when training, the original size being 288×288 . The additional training parameters are also listed in Table 6. For UVT, $\mathcal{L}_{content}$ and \mathcal{L}_ϕ are only used to improve the convergence of the training process. We fade out $\mathcal{L}_{content}$ in the first 10k batches and \mathcal{L}_ϕ is used for the first 80k and faded out in last 20k.

G PERFORMANCE

TecoGAN is implemented in *TensorFlow*. While generator and discriminator are trained together, we only need the trained generator network for the inference of new outputs after training, i.e., the whole discriminator network can be discarded. We evaluate the models on a Nvidia GeForce GTX 1080Ti GPU with 11G memory, the resulting VSR performance for which is given in Table 2.

The VSR TecoGAN[⊖] model and FRVSR have the same number of weights (843587 in the generator network and 1.7M in F), and thus show very similar performance characteristics with around 37 ms per frame. The larger VSR TecoGAN model with 1286723 weights in the generator is slightly slower than them, spending 42 ms per frame. In the UVT task, generators spend around 60 ms per frame with a size of 512×512 . However, compared with models of DUF, EDVR and RBPN, which have 6 to 20 million weights, the performance of TecoGAN is significantly better thanks to its reduced size.