SCALE SPACE FLOW WITH AUTOREGRESSIVE PRIORS

Ruihan Yang[1], **Yibo Yang**[1], **Joseph Marino**[2] **and Stephan Mandt**[1]
Department of Computer Science, University of California, Irvine[1]
Computation & Neural Systems, California Institute of Technology[2]
{ruihan.yang,yibo.yang,mandt}@uci.edu, jmarino@caltech.edu

ABSTRACT

There has been a recent surge of interest in neural video compression models that combine data-driven dimensionality reduction with learned entropy coding. Scale Space Flow (SSF) is among the most popular variants due to its favorable rate-distortion(RD) performance. Recent work showed that this approach could be further improved by structured priors and stochastic temporal autoregressive transforms on the frame level. However, as of early 2021, most state-of-the-art compression approaches work with time-independent priors. Assuming that frame latents are still temporally correlated, further compression gains should be expected by conditioning the priors on temporal information. We show that the naive way of conditioning priors on previous stochastic latent states degrades performance, but temporal conditioning on a deterministic quantity does lead to a consistent improvement over all baselines. Evaluating the benefits of the temporal prior given the involved challenges in training and deployment remains an open question.

1 INTRODUCTION

The last two years have seen a surge in exploring deep latent variable models for neural video compression (Agustsson et al., 2020; Yang et al., 2020a;b; 2021; 2020d; Habibian et al., 2019; Lu et al., 2019; Liu et al., 2019). In contrast to neural *image* codecs that assume i.i.d. data, neural video codecs exploit temporal redundancies in the frame sequence. Just as their classical analogs, neural codecs typically involve two operations: motion compensation and residual compression. In the motion compensation step, the model exploits information about already decoded frames to predict the most likely next frame in sequence (e.g., by compressing an optical flow field). In the residual compression step, the deviation from this prediction is encoded.

With few exceptions (Han et al., 2019; Yang et al., 2020b), most of the currently proposed methods do not exploit temporal information on the *prior* level. Instead, they use time-independent image compression models to compress the flow fields and residuals. In this paper, we analyze how these models can be improved by temporal conditioning. Our paper thereby considers a state-of-the-art model for low-latency video compression: Scale Space Flow (SSF) (Agustsson et al., 2020). Recent work has shown that this model can be further improved with hierarchical design by introducing structured priors and stochastic temporal autoregressive transforms on the frame level (Yang et al., 2020c). However, none of these versions use temporally conditioned priors for entropy coding.

We consider two types of temporally conditioned priors. In a first attempt, we condition the prior of the residual on the previous time step's latent residual. Despite being based on first principles in latent time series modeling, we find that such temporal conditioning degrades the performance in some cases, which we attribute to optimization problems. To circumvent the problem, we propose to condition the prior on the previous reconstructed frame, which refers to a more complicated but stable way of performing temporal conditioning. We show that this way of including temporal information leads to consistent improvements over SSF and slight improvement over the newly proposed hierarchical model by Yang et al. (2021).

2 Temporal Priors for Video Compression Models

We follow the probabilistic modeling formalism of low-latency video compression proposed in (Yang et al., 2020c), in which a learned compression model consists of a *prior* distribution $p(\mathbf{z}_{1:T})$ of latent variables, a *decoder* that parameterizes the mean of Gaussian likelihood distribution $p(\mathbf{x}_{1:T}|\mathbf{z}_{1:T})$ of video frames $\mathbf{x}_{1:T}$, and an *encoder* that parameterizes the mean of a box-shaped variational posterior $q(\mathbf{z}_{1:T}|\mathbf{x}_{1:T})$. Based on the variational compression approach (Ballé et al., 2017) that approximates rounding operations ($\lfloor \cdot \rceil$) by uniform noise injection, the rate-distortion function of such a compression model then corresponds to the β-VAE objective:

$$\tilde{\mathcal{L}} = \mathbb{E}_{q(\mathbf{z}_{1:T}|\mathbf{x}_{1:T})}[-\log p(\mathbf{x}_{1:T}|\mathbf{z}_{1:T}) - \beta \log p(\mathbf{z}_{1:T})]. \tag{1}$$

In this work, we explore various forms of the latent prior $p(\mathbf{z}_{1:T})$ that extend beyond the temporally factorized prior $\prod_t p(\mathbf{z}_t)$ used in most of previous neural video compression methods.

2.1 Low-latency Video Compression Models

A learned video compression model described above is primarily determined by the form of latent variables $\mathbf{z}_t$, as well as the likelihood $p(\mathbf{x}_{1:T}|\mathbf{z}_{1:T})$. The dominant low-latency compression methods employ a latent variable with two components, $\mathbf{z}_t = (\mathbf{w}_t, \mathbf{v}_t)$, where $\mathbf{w}_t$ corresponds to the encoded motion information, and $\mathbf{v}_t$ the encoded residual information, and a Gaussian likelihood $p(\mathbf{x}_t|\mathbf{z}_{\leq t})$ whose mean $\hat{\mathbf{x}}_t$ is the reconstructed version of the ground truth frame $\mathbf{x}_t$ as would be computed by the decoder. The Stochastic Temporal Autoregressive Transform (Yang et al., 2021) is a general decoding framework for computing $\hat{\mathbf{x}}_t$ that captures a variety of existing low-latency video compression frameworks as special cases:

$$\hat{\mathbf{x}}_t \;=\; h_\mu(\hat{\mathbf{x}}_{t-1}, g_w(\mathbf{w}_t)) + h_\sigma(\hat{\mathbf{x}}_{t-1}, \mathbf{w}_t) \odot g_v(\mathbf{v}_t, \mathbf{w}_t). \qquad \text{(Yang et al., 2021)} \tag{2}$$

For example, SSF (Agustsson et al., 2020) is a special case of this class and reduces to

$$\hat{\mathbf{x}}_t \;=\; h_{warp}(\hat{\mathbf{x}}_{t-1}, g_w(\mathbf{w}_t)) + g_v(\mathbf{v}_t, \mathbf{w}_t). \qquad \text{(Agustsson et al., 2020)} \tag{3}$$

In Eq. 3, g_w and g_v are deep neural networks, $g_w(\mathbf{w}_t)$ has the interpretation of an estimated optical flow (motion) field, h_{warp} is the computer vision technique of warping, and the residual $\mathbf{r}_t := g_v(\mathbf{v}_t, \mathbf{w}_t) = \hat{\mathbf{x}}_t - h_{warp}(\hat{\mathbf{x}}_{t-1}, g_w(\mathbf{w}_t))$ represents the prediction error unaccounted for by warping. Lu et al. (2019) only makes use of $\mathbf{v}_t$ in the residual decoder g_v, and performs simple 2D warping by bi-linear interpretation; SSF (Agustsson et al., 2020) augments the optical flow (motion) field with an additional scale field, and applies scale-space-warping to the progressively blurred versions of $\hat{\mathbf{x}}_{t-1}$ to allow for uncertainty in the warping prediction.

Eq. 2 resembles the definition of the Masked Autoregressive Flow (Papamakarios et al., 2017). Above, h_μ and h_σ are functions that transform the previous reconstructed data frame $\hat{\mathbf{x}}_{t-1}$ along with $\mathbf{w}_t$ into a shift and scale parameter, respectively, where h_μ use the same operation as h_{warp}. The function $g_v(\mathbf{v}_t, \mathbf{w}_t)$ converts these latent variables into a noise variable that encodes residuals with respect to the mean next-frame prediction $h_\mu(\hat{\mathbf{x}}_{t-1}, \mathbf{w}_t)$. This provides an added degree of flexibility, in a similar fashion to how RealNVP improves over NICE (Dinh et al., 2014; 2016) in the normalizing flow literature.

2.2 Temporal Prior

In previous works in variational compression, the prior model $p(\mathbf{z}_{1:T})$ typically assumes that the latent variables $\mathbf{z}_{1:T}$ are temporally factorized: $p(\mathbf{z}_{1:T}) = \prod_{t=1}^{T} p(\mathbf{z}_t)$. However, such assumptions may be unrealistic for real world data, such as modeling natural video (Denton & Fergus, 2018; Li & Mandt, 2018; Marino et al., 2020), where temporal dependencies may persist even after removing low-level motion information. To this end, we model these dependencies by introducing a temporal prior: $p(\mathbf{z}_{1:T}) = p(\mathbf{z}_1) \prod_{t=2}^{T} p(\mathbf{z}_t|\mathbf{z}_{<t})$ to more accurately predict the density of the latent variable. Given the framework described in Section 2.1 and Figure 1, we implement the prior of the video compression model with the following two types of temporal conditioning.

Naïve Residual Conditioning $p(\mathbf{v}_t|\mathbf{v}_{t-1})$ (TP). For video compression, we usually make the Markov assumption, as optical flow only shows a one-step vector field of adjacent frames. Moreover, because $-\log p(\mathbf{v}_t) \gg -\log p(\mathbf{w}_t)$, it is not necessary to further reduce the bitrate of the

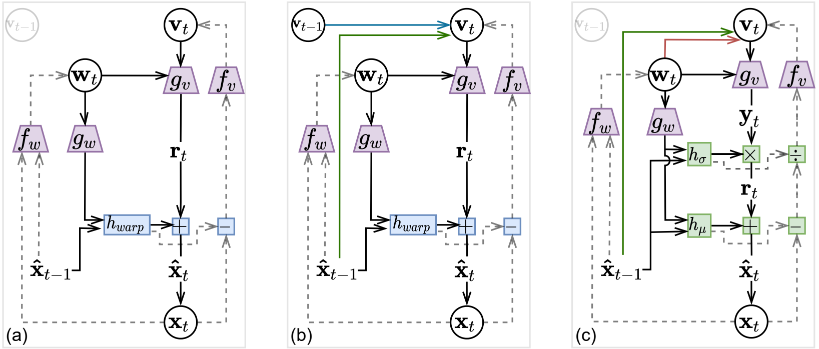


Figure 1: **Model diagrams** for the generative and inference procedures of the current frame $\mathbf{x}_t$, for various neural video compression methods considered (see Table 1). Random variables are shown in circles; all other quantities are deterministically computed; solid and dashed arrows describe computational dependency during generation (decoding) and inference (encoding), respectively. Purple nodes correspond to neural encoders and decoders. From left to right: (a) **SSF** (Agustsson et al., 2020). (b) **SSF-TP(+)** (this paper) with two possible prior choices: conditioning on $\mathbf{v}_{t-1}$ ("TP", blue arrow) and conditioning on $\hat{\mathbf{x}}_{t-1}$ ("TP+", green arrow). (c) **STAT-SSF** (Yang et al., 2021) with optional modified priors. The basic model ("STAT-SSF") uses no temporal prior (no solid arrows into $\mathbf{v}_t$). **STAT-SSF-SP** introduced a structured (but not temporal) prior $p(\mathbf{v}_t|\mathbf{w}_t)$ via the red arrow. **STAT-SSF-SP-TP+** (this paper) uses additional temporal information from $\hat{\mathbf{x}}_{t-1}$ (green arrow).

latent variable $\mathbf{w}$, so we choose to keep $p(\mathbf{w}_t)$ temporally factorized for simplicity. Temporal conditioning in $\mathbf{v}$ is implemented by using an additional neural network that takes $\mathbf{v}_{t-1}$ as input to construct the conditional Gaussian prior for $\mathbf{v}_t$. However, naïve conditioning may require special handling for the initial frames because the individual image compression model for the I-frame (the first frame) does not have a "residual." This indicates that such temporal conditioning can only start from the 2$^{\text{nd}}$ frame, as $p(\mathbf{v}_t|\mathbf{v}_{t-1})$ is only available for $t \geq 3$. Instead, we use a separate factorized prior $p(\mathbf{v}_2)$ for the 2$^{\text{nd}}$ frame.

Conditioning on Previous Frame $p(\mathbf{v}_t|\hat{\mathbf{x}}_{t-1})$ (TP+). Learning a temporal prior directly may be challenging, as $\mathbf{v}$ changes throughout learning without *complete* information from previous time step (only residual), affecting both the inputs and outputs of the prior. Thus, in addition to naïve conditioning, we explore an alternative scheme, **TP+**, in which $\mathbf{v}_t$ is conditioned on the previous reconstruction, $\hat{\mathbf{x}}_{t-1}$, which maintains a fixed representation throughout training. In this scenario, the model no longer requires the extra prior for $p(\mathbf{v}_2)$, as in **TP**. However, **TP+** may require additional convolution layers to transform $\hat{\mathbf{x}}_{t-1}$ to match the dimensionality of $\mathbf{v}_t$, which leads to the increase of memory consumption and the time of training and evaluation.

Structured Prior $p(\mathbf{v}_t|\mathbf{w}_t)$ (SP). Yang et al. (2021) explores the use of a structured prior within time steps, conditioning $\mathbf{v}_t$ on $\mathbf{w}_t$, observing that motion information encoded in $\mathbf{w}_t$ can often be informative of the residual error encoded in $\mathbf{v}_t$. While this is not a temporal prior, Yang et al. (2021), we employ the same structured prior in our setup for supplemental evaluation.

3 Experiments

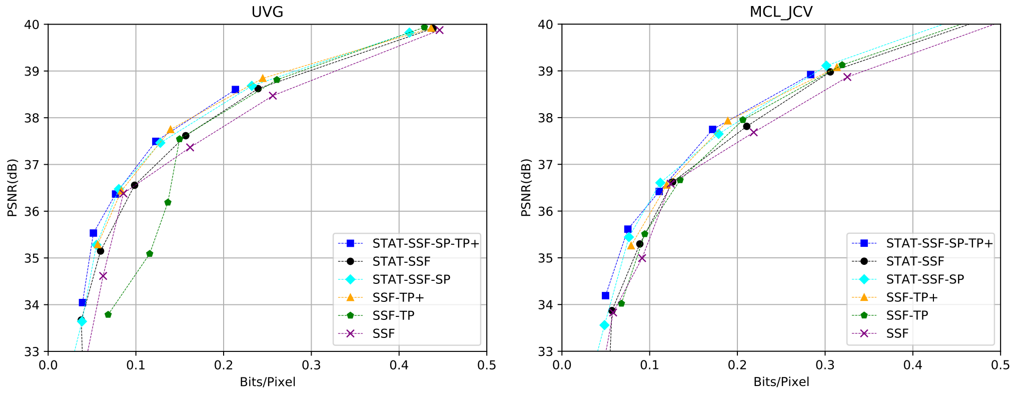
We conducted experiments to illustrate the rate-distortion performance of SSF (Agustsson et al., 2020) and its recent extension based on stochastic autoregressive transforms (STAT-SSF) (Yang et al., 2021) using different temporal priors.

Data. We use Vimeo-90k (Xue et al., 2019) and UVG (Mercat et al., 2020) to train and evaluate the model, using the conventional train-test splits and schedules outlined in (Yang et al., 2021). We use sequences of 3 consecutive frames for training and 100 frames for evaluation.

Ablations and Baselines. We compare against a variety of models and ablations from Agustsson et al. (2020); Yang et al. (2021). For naming conventions, we refer to Table 1. The first three rows are existing versions; the last three ones are proposed ablations.

Table 1: Naming Convention and Overview of Ablation Models

Model Name	**Reference / Definition**
SSF	Base model of Agustsson et al. (2020)
STAT-SSF	Yang et al. (2021); version with factorized prior $p(\mathbf{v}_t)$
STAT-SSF-SP	Yang et al. (2021); version with structured prior (SP), $p(\mathbf{v}_t\|\mathbf{w}_t)$
SSF-TP	New: Agustsson et al. (2020) with naive temporal prior (TP), $p(\mathbf{v}_t\|\mathbf{v}_{t-1})$
SSF-TP+	New: Agustsson et al. (2020) with temporal prior conditioned on $\hat{\mathbf{x}}_{t-1}$, $p(\mathbf{v}_t\|\hat{\mathbf{x}}_{t-1})$
STAT-SSF-SP-TP+	New: structured prior (SP) of Yang et al. (2021) plus improved temporal prior, $p(\mathbf{v}_t\|\hat{\mathbf{x}}_{t-1}, \mathbf{w}_t)$



(a)

(b)

Figure 2: **Rate-Distortion Performance** of various models and ablations (see Table 1). Results are evaluated on **(a)** UVG and **(b)** MCL_JCV datasets. Best results are obtained by deterministic temporal conditioning (dark blue), while conditioning on previous latents (green) degrades performance.

Figure 2 shows the RD curve of each ablation model. We see that performing temporal conditioning through the prior does indeed improve performance, both for **STAT** and **SSF**. However, we also observe that, as compared with the **SSF** baseline model, **SSF-TP** degrades in performance at the low bitrate regime on the UVG dataset. We hypothesize that this may be attributable either to optimization challenges, which are known to affect latent variable models (Bowman et al., 2015) or video length inconsistency between training and evaluation data. As far as we know, most proposed generative or compression models with temporal prior are trained with more than three consecutive frames (Marino et al., 2020; Li & Mandt, 2018; Denton & Fergus, 2018; Liu et al., 2019; Yang et al., 2020a;b), as three frames can only cover one step of observation $p(\mathbf{v}_3|\mathbf{v}_2)$ during training in our case. Indeed, we see that instead temporally conditioning on the previous reconstruction (**SSF-TP+**) does not suffer from the same issue, outperforming naïve conditioning in almost all bitrate regimes across both datasets. Finally, **SSF-TP+** performs comparably with the previous state-of-the-art, **STAT-SSF-SP**, and the hybrid model **STAT-SSF-SP-TP+** offers slight improvement.

4 Discussion

We provide an ablation study on the use of temporally-conditioned priors for neural video compression. Our results demonstrate that temporal conditioning can improve performance, however, this can also raise additional challenges on optimization and model complicity. Conditioning on previous reconstructions, rather than latent variables, appears to provide a simple method for learning temporal priors. Our temporally-conditioned models outperform current state-of-the-art video compression methods without temporal priors, scale-space-flow (Agustsson et al., 2020) and its hierarchical generalized version (Yang et al., 2021). Future work can further investigate the challenges that arise in learning temporal priors between latent variables, particularly with regard to local optima and other optimization issues.

5 Acknowledgements

We gratefully acknowledge extensive contributions from Yang Yang (Qualcomm), which were indispensable to this work. This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR001120C0021. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Defense Advanced Research Projects Agency (DARPA). Yibo Yang acknowledges funding from the Hasso Plattner Foundation. Furthermore, this work was supported by the National Science Foundation under Grants 1928718, 2003237 and 2007719, as well as Intel and Qualcomm.

References

Eirikur Agustsson, David Minnen, Nick Johnston, Johannes Balle, Sung Jin Hwang, and George Toderici. Scale-space flow for end-to-end optimized video compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8503–8512, 2020.

Johannes Ballé, Valero Laparra, and Eero P Simoncelli. End-to-end optimized image compression. *International Conference on Learning Representations*, 2017.

Samuel R Bowman, Luke Vilnis, Oriol Vinyals, Andrew M Dai, Rafal Jozefowicz, and Samy Bengio. Generating sentences from a continuous space. *arXiv preprint arXiv:1511.06349*, 2015.

Emily Denton and Rob Fergus. Stochastic video generation with a learned prior. *arXiv preprint arXiv:1802.07687*, 2018.

Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

Amirhossein Habibian, Ties van Rozendaal, Jakub M Tomczak, and Taco S Cohen. Video compression with rate-distortion autoencoders. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 7033–7042, 2019.

Jun Han, Salvator Lombardo, Christopher Schroers, and Stephan Mandt. Deep generative video compression. 2019.

Yingzhen Li and Stephan Mandt. Disentangled sequential autoencoder. *arXiv preprint arXiv:1803.02991*, 2018.

Haojie Liu, Lichao Huang, Ming Lu, Tong Chen, Zhan Ma, et al. Learned video compression via joint spatial-temporal correlation exploration. *arXiv preprint arXiv:1912.06348*, 2019.

Guo Lu, Wanli Ouyang, Dong Xu, Xiaoyun Zhang, Chunlei Cai, and Zhiyong Gao. Dvc: An end-to-end deep video compression framework. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 11006–11015, 2019.

Joseph Marino, Lei Chen, Jiawei He, and Stephan Mandt. Improving sequential latent variable models with autoregressive flows. In *Symposium on Advances in Approximate Bayesian Inference*, pp. 1–16, 2020.

Alexandre Mercat, Marko Viitanen, and Jarno Vanne. Uvg dataset: 50/120fps 4k sequences for video codec analysis and development. In *Proceedings of the 11th ACM Multimedia Systems Conference*, pp. 297–302, 2020.

George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density estimation. In *Advances in Neural Information Processing Systems*, pp. 2338–2347, 2017.

Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. *International Journal of Computer Vision (IJCV)*, 127(8):1106–1125, 2019.

Ren Yang, Fabian Mentzer, Luc Van Gool, and Radu Timofte. Learning for video compression with hierarchical quality and recurrent enhancement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020a.

Ren Yang, Fabian Mentzer, Luc Van Gool, and Radu Timofte. Learning for video compression with recurrent auto-encoder and recurrent probability model. *IEEE Journal of Selected Topics in Signal Processing*, 2020b.

Ruihan Yang, Yibo Yang, Joseph Marino, Yang Yang, and Stephan Mandt. Deep generative video compression with temporal autoregressive transforms. *ICML 2020 Workshop on Invertible Neural Networks, Normalizing Flows, and Explicit Likelihood Models*, 2020c.

Ruihan Yang, Yibo Yang, Joseph Marino, and Stephan Mandt. Hierarchical autoregressive modeling for neural video compression. In *International Conference on Learning Representations*, 2021.

Yang Yang, Guillaume Sautière, J Jon Ryu, and Taco S Cohen. Feedback recurrent autoencoder. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3347–3351. IEEE, 2020d.