Latent Neural Differential Equations for Video Generation

Cade Gordon, Natalie Parde
Department of Computer Science
University of Illinois at Chicago
851 S. Morgan St., Chicago, IL 60607
cadegordonml@gmail.com, parde@uic.edu

Abstract

Generative Adversarial Networks have recently shown promise for video generation, building off of the success of image generation while also addressing a new challenge: time. Although time was analyzed in some early work, the literature has not adequately grown with temporal modeling developments. We propose studying the effects of Neural Differential Equations to model the temporal dynamics of video generation. The paradigm of Neural Differential Equations presents many theoretical strengths including the first continuous representation of time within video generation. In order to address the effects of Neural Differential Equations, we will investigate how changes in temporal models affect generated video quality.

1 Introduction

Generative modeling remains an important problem within computer vision, with new developments providing a better understanding of high-dimensional data modeling and even aiding the supervised learning sphere. Good representations of distributions improve feature space visualisation, clustering, and classification. Many approaches have tackled the problem of representing a distribution, including Generative Adversarial Networks (GANs) [10], which have recently shown immense potential for image generation. As time progresses GANs become more robust, allowing for greater image size [32, 22, 3] and quality [20, 21].

The success of GANs in image generation propelled them towards being the prominent methodology for video generation. However, the application of GANs to video generation has come with new challenges. Adding time to the preexisting color, width, and height dimensions has increased computational costs and complexity by an order of magnitude. Early models generated videos of a meagerly 64 by 64 pixels [42, 34, 39]. The addition of the new temporal component not only restricted video size, it also opened many questions regarding the best way to navigate an entirely new dimension. The first model to use GANs for video generation was VGAN [42], which used 3D convolutional kernels to account for time, framing it as no more than an extra feature channel blended in with color, width and height.

Treating temporal features as a separate dimensional scope allowed for the subsequent TGAN [34] to outperform VGAN in terms of Inception Score (IS) [36]. The authors proposed two separate generational architectures: a 1D convolutional temporal generator and an image generator. Further investigation of the temporal latent space was done by MoCoGAN [39], in which the authors proposed decomposing the image generator's input into a single content vector and an evolving motion vector. Experimentation has also gone into increasing frame size and network depth, with some reflections on computational mitigation [35, 6, 24], but little work has gone into rigorously examining time.

Our work reopens the discussion of the temporal latent space. After the revelation of separate temporal generation, researchers have stopped asking questions about the temporal generator. Works

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

after TGAN employed Long Short-Term Memory (LSTM) [16] or Convolutional LSTM (CLSTM) [43] blocks. To this day, the LSTM remains and has never been fully ablated or examined with control. A similar previously unproven but accepted notion was content motion decomposition. First published in 2018 as part of MoCoGAN [39], content and motion decomposition was not ablated until the publication of MoFlowGAN [24] in 2020. With very limited analysis, much of the temporal space remains an open question within these models.

While able to model temporal dynamics, recurrent models such as the LSTM and its variants only represent discrete samples. We propose to re-explore the temporal space under a continuous paradigm. Neural Ordinary Differential Equations (NODEs) [5] offer the potential for a continuous representation of the temporal dimension. Extending the paradigm of Neural Differential Equations, we propose the first continuous video generation model.

Our work will make the following contributions:

- We will establish the first continuous GAN for video generation.
- We will experiment with multiple novel architectures for video generation.
- We will analyze how changes in the temporal latent space modality affect visual fidelity through an ablation study.

2 Related Work

2.1 Generative Adversarial Networks

Two neural networks compose GANs: a Discriminator D and Generator G. The generator transforms a sampled noise vector z from a distribution p_z and maps it to an image (or in our case a video). The Discriminator functions by taking an input image or video x and mapping it to a value representing whether it believes x is sampled from the real distribution p_x or the distribution produced by the generator p_g. The two compete to minimize or maximize a loss function that may be represented generically as shown below, where ϕ is a function of the Discriminator's prediction and the truth label represented as 1 (real) or 0 (fake):

$$\max_G \min_D \mathbb{E}_{x\sim p_x}[\phi(D(x),1)] + \mathbb{E}_{z\sim p_z}[\phi(D(G(z)),0)]$$

ϕ is typically the identity function, cross entropy function, or hinge loss function. Loss choice has been shown to be less consequential so long as a Lipschitz constraint is met [31]. GANs are often difficult to train, and two approaches to increasing their stability during training time are through applying a form of Lipschitz constraint or multi-scale generation. WGAN [1] and WGAN-GP [12] showed the effectiveness of the Lipschitz regularization. SNGAN [30] subsequently showed a refined way to enforce the constraint through spectral normalization. Progressive GAN [22] stabilized training by increasing the generated resolution over time.

2.2 Video Prediction

Video prediction conditions a model on a sample of frames, and models the subsequent frames. A common approach is the use of a recurrent architecture such as an LSTM [33, 37, 8, 17, 4, 29]. Another common methodology is using optical flow [27, 28, 13, 26]. Prior work has also explored the stochastic nature of videos [7, 2, 23, 9, 41].

2.3 Video Generation

To the best of our knowledge, the first work to use a GAN to generate videos was VGAN [42]. VGAN generated videos using spatio-temporal convolutions with 3D kernels and fractional strides, separately generating the motion and background. In order to combine the two it used a learned mask to produce the final output.

Its successor, TGAN [34], separately generated temporal and frame features. TGAN transformed a single noise vector into multiple vectors accounting for time with a temporal generator G_t, a series of 1D convolutions. The generated vectors concatenated with the starting single noise vector were

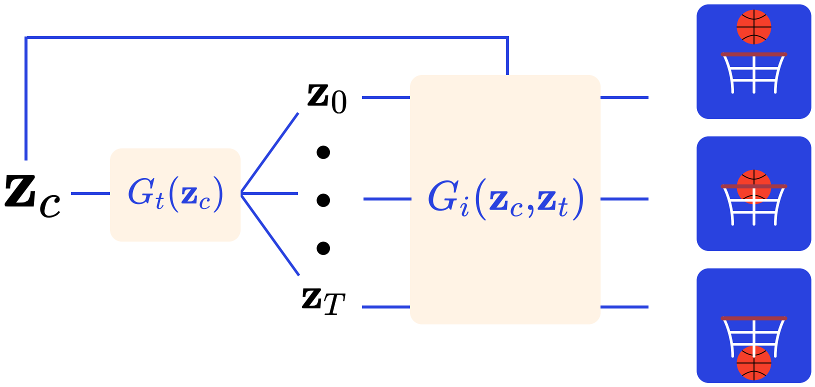

Figure 1: A latent variable $\mathbf{z}_c$ is transformed by G_t into a series of temporal vectors $\mathbf{z}_0, \mathbf{z}_1, ..., \mathbf{z}_T$. Each temporal vector $\mathbf{z}_t$ is concatenated with $\mathbf{z}_c$ and transformed into an image. Said images are joined to compose a video.

then fed into an image generator G_i. By separating temporal generation into its own process, TGAN outperformed VGAN [36]. The general form of G may be seen in Figure 1.

MoCoGAN [39] continued in the line of temporal manipulation by using an LSTM to generate temporal features. The authors assumed that the temporal space was composed of a motion and content subspace. Though their work outperformed TGAN, it was not until MoFlowGAN [24], two years later, that the content and motion decomposition was fairly ablated, with results showing that it led to a positive increase in IS. It is hard to know for many of these models which features actually allowed for their success, since the discriminator and image generation architectures change and increase in complexity from one paper to the next. In light of this, a properly controlled analysis of our proposed model will fill in many of the gaps in the current literature.

Other papers focus on increasing the dimension of the video output. DVD-GAN and MoFlowGAN [6, 24] produce 128x128 pixel videos. The current state-of-the-art TGANv2 [35] even boasts 192x192 pixel videos. Recently TGAN-F [18] further improved performance by simplifying the discriminator of TGAN.

2.4 Neural Differential Equations

NODEs [5] transformed the vision of ResNets [14] by giving them a continuous definition. Instead of the singular discrete additions of a neural network function $f(x)$, they proposed integrating using ordinary differential equation (ODE) solvers. The new interpretation allows for an approximate continuous temporal representation, where $\mathbf{h}$ represents the hidden state of a layer, and t represents the ordering of layers:

$$\mathbf{h}_{t+1} = \mathbf{h}_t + f(\mathbf{h}_t) = \mathbf{h}_t + \int_t^{t+1} g(\mathbf{h}_t, t)t$$
$$\frac{d\mathbf{h}_t}{dt} = g(\mathbf{h}_t, t)$$

We use t to represent time. Works like ODE^{2}VAE [44] extended this to second order ODEs. Much of the work surrounding NODEs revolves around Variational Autoencoders [5, 11, 44]. Recently, even more differential equation families have been explored; Neural Stochastic Differential Equations (NSDEs) are a successful example [25, 40].

3 Neural Differential Equation Video GANs

There is a significant gap in the literature explaining the choice for temporal generators. To remedy this, we propose to explore it under a paradigm common to general physics: using differential

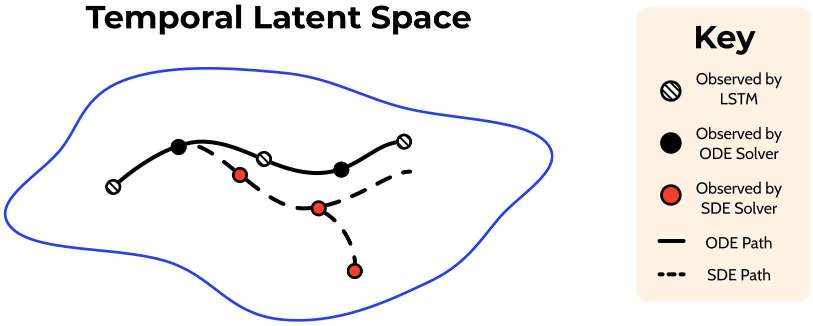

Figure 2: When compared with typical LSTMs, neural differential equations have more frequent observations, and SDEs have greater potentiality for solutions.

equations to represent temporal dynamics. While using historical image generator functions, we will observe changes in performance metrics with different temporal generator functions. Comparisons between the families of generative functions may be visualized by Figure 2.

3.1 Ordinary Differential Equations (ODEs)

Instead of an auto-regressive LSTM or 1D Kernel, a differential equation may be used to model the evolution of a latent variable $\mathbf{z}_t$. By a learned function $f(\mathbf{z}_t)$, future $\mathbf{z}_T$ may be found by integrating:

$$\mathbf{z}_T = \mathbf{z}_0 + \int_0^T f(\mathbf{z}_t, t)dt$$

$$\dot{\mathbf{z}}_t = \frac{d\mathbf{z}_t}{dt} = f(\mathbf{z}_t, t)$$

The image generator $G_i(\mathbf{z})$ may then produce an image from $\mathbf{z}_t$. Using a differential equation the model may account for the finer nuances of traversing the latent space accounting for motion in $\mathbf{z}_{t<t+\epsilon<t+1}$. LSTMs only view sparse time steps; the model moves from $\mathbf{z}_t$ to $\mathbf{z}_{t+1}$ never accounting for a $\mathbf{z}_{t+0.5}$. NODEs allow for the intermediate $\mathbf{z}_t$ values to be traversed, which may potentially lead to better performance as this can more closely approximate a latent trajectory.

The family of NODEs also allows for higher order interpretations of the model. Our $f(\mathbf{z}_t)$ may represent higher orders than simple $\dot{\mathbf{z}}_t$, such as $\ddot{\mathbf{z}}_t$ or higher. A first-order ODE parameterizes more immediate changes during integration, whereas higher orders represent much more long-term shifts, such as concavity, in the latent variable.

3.2 Stochastic Differential Equations (SDEs)

NODEs allow for path approximations in determinate systems. Every $\mathbf{z}_t$ will produce a single $\mathbf{z}_{t+1}$, but this isn't reflective of how videos truly function. There is an inherent stochastic nature to how videos progress—actors have a branching tree of decisions and so do particles for their motion. NSDEs may be a good way to represent the random nature present in videos, offering all of the benefits of NODEs while allowing for randomness with their added noise. Under this form, and letting $\mu(\mathbf{z}_t)$ and $\sigma(\mathbf{z}_t)$ represent drift and diffusion respectively, we find $\mathbf{z}_T$ with:

$$\mathbf{z}_T = \mathbf{z}_0 + \int_0^T \mu(\mathbf{z}_t, t)dt + \int_0^T \sigma(\mathbf{z}_t, t)dW_t$$

Each of $\mu(\mathbf{z}_t)$ and $\sigma(\mathbf{z}_t)$ are parameterized by a neural network. W_t is a Wiener process, a continuous series of values with Gaussian increments.

The validity of this formulation may be exemplified by thinking about a video of a face changing expressions. If the actor starts out with a neutral face they may then produce a sad one after. However, a smile would be equally likely. By injecting randomness either path may be explored by the model.

3.3 Benefits of Differential Equations

Differential equations allow for increased control over how paths are traversed because of their continuous properties. Because $\mathbf{z}_t$ is found by integration, there are two unique characteristics that other modalities do not possess. First, $\mathbf{z}_t$ can be integrated backwards in time allowing the discovery of $\mathbf{z}_{t-n}$. This can be thought of as what happens before the first frame. Second, if increased frame rates are desired, they can easily be accounted for. The differential equation solver will necessitate evaluations of $\mathbf{z}_{t<t+\epsilon<t+1}$. To achieve a higher frame rate, the image generator simply needs to sample some of the intermediate $\mathbf{z}_t$ evaluations. Control like this is impossible in recurrent models.

4 Experimental Protocol

The most widely used and comparable metrics for video generation are IS and Fréchet Inception Distance (FID) [15]. These are calculated by a C3D model [38] pretrained on the UCF101 dataset [19]. Their values quantify visual fidelity of the generated videos. We will observe changes to these metrics as we alter G_t.

We will train the following models on UCF101:

- TGAN
- MoCoGAN
- TGANv2 (used for Effects of Family and Order only)

Each model will run for 100,000 epochs using the model's originally proposed hyperparameters. IS will be calculated on samples of 2,048 videos every 2,000 epochs. The epoch with the highest IS will be used to calculate the model's final statistics. Using the best performing epoch, five batches of 2,048 videos will be created. FID and IS will be calculated on each batch, and we will report the mean value and standard deviation for each metric.

4.1 Finding $f(x)$

In order to generate videos under this paradigm an appropriate neural network architecture for $f(x)$ needs to be studied. To find $f(x)$, TGAN and MoCoGAN will be trained under different G_ts. For each, $\mathbf{z}_t$ will be found by integrating $f(x)$ as a first order NODE. Ablation will occur with the following $f(x)$s: $f : \mathbb{R}^d \to \mathbb{R}^d$ using a single learned layer with a nonlinearity; $f, g, h : \mathbb{R}^d \to \mathbb{R}^d$ where $f(x) = (g \circ h)(x)$, with g and h being also single learned layers with a nonlinearity; and the same functions as the previous setup but with g and h equalizing parameters of each model's original G_t. Testing these choices of $f(x)$ across both TGAN and MoCoGAN allows for greater evidence for or against how well each $f(x)$ generalizes to the task and architecture.

4.2 Effects of Family and Order

With an effective $f(x)$, we can ablate the multiple families and orders. TGAN, MoCoGAN, and now TGANv2 will be tested under the following motion generators: the model's original G_t, the first order ODE, the second order ODE, the third order ODE, and the SDE. For each configuration we will report IS and FID using the process specified earlier.

5 Conclusion

Differential equations have theoretical promise for video generation. We would like to experimentally confirm or reject the validity of using Neural Differential Equations as the temporal generator in video GANs. Our work will help fill the gap in the literature pertaining to the choice of G_t.

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. "Wasserstein generative adversarial networks". In: *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. 2017, pp. 214–223.

[2] Mohammad Babaeizadeh et al. "Stochastic Variational Video Prediction". In: *International Conference on Learning Representations*. 2018.

[3] Andrew Brock, Jeff Donahue, and Karen Simonyan. "Large Scale GAN Training for High Fidelity Natural Image Synthesis". In: *International Conference on Learning Representations*. 2018.

[4] Wonmin Byeon et al. "Contextvp: Fully context-aware video prediction". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 753–769.

[5] Ricky T. Q. Chen et al. "Neural Ordinary Differential Equations". In: *Advances in Neural Information Processing Systems 31*. Ed. by S. Bengio et al. Curran Associates, Inc., 2018, pp. 6571–6583. URL: `http://papers.nips.cc/paper/7892-neural-ordinary-differential-equations.pdf`.

[6] Aidan Clark, Jeff Donahue, and Karen Simonyan. "Adversarial video generation on complex datasets". In: *arXiv* (2019), arXiv–1907.

[7] Emily Denton and Rob Fergus. "Stochastic Video Generation with a Learned Prior". In: *International Conference on Machine Learning*. 2018, pp. 1174–1183.

[8] Chelsea Finn, Ian Goodfellow, and Sergey Levine. "Unsupervised learning for physical interaction through video prediction". In: *Advances in neural information processing systems*. 2016, pp. 64–72.

[9] Jean-Yves Franceschi et al. *Stochastic Latent Residual Video Prediction*. 2020. arXiv: `2002.09219 [cs.CV]`.

[10] Ian Goodfellow et al. "Generative Adversarial Nets". In: *Advances in Neural Information Processing Systems 27*. Ed. by Z. Ghahramani et al. Curran Associates, Inc., 2014, pp. 2672–2680. URL: `http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf`.

[11] Will Grathwohl et al. "FFJORD: Free-Form Continuous Dynamics for Scalable Reversible Generative Models". In: *International Conference on Learning Representations*. 2018.

[12] Ishaan Gulrajani et al. "Improved training of wasserstein gans". In: *Advances in neural information processing systems*. 2017, pp. 5767–5777.

[13] Z. Hao, X. Huang, and S. Belongie. "Controllable Video Generation with Sparse Trajectories". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 7854–7863.

[14] Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[15] Martin Heusel et al. "Gans trained by a two time-scale update rule converge to a local nash equilibrium". In: *Advances in neural information processing systems*. 2017, pp. 6626–6637.

[16] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory". In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[17] Jun-Ting Hsieh et al. "Learning to Decompose and Disentangle Representations for Video Prediction". In: *Advances in Neural Information Processing Systems 31*. Ed. by S. Bengio et al. Curran Associates, Inc., 2018, pp. 517–526. URL: `http://papers.nips.cc/paper/7333-learning-to-decompose-and-disentangle-representations-for-video-prediction.pdf`.

[18] Emmanuel Kahembwe and Subramanian Ramamoorthy. "Lower Dimensional Kernels for Video Discriminators". In: *arXiv preprint arXiv:1912.08860* (2019).

[19] Andrej Karpathy et al. "Large-scale video classification with convolutional neural networks". In: *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 2014, pp. 1725–1732.

[20] Tero Karras, Samuli Laine, and Timo Aila. "A Style-Based Generator Architecture for Generative Adversarial Networks". In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 4396–4405.

[21] Tero Karras et al. "Analyzing and improving the image quality of stylegan". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 8110–8119.

[22] Tero Karras et al. "Progressive Growing of GANs for Improved Quality, Stability, and Variation". In: *International Conference on Learning Representations*. 2018.

[23] Alex X. Lee et al. *Stochastic Adversarial Video Prediction*. 2018. arXiv: `1804.01523 [cs.CV]`.

[24] Wei Li et al. "Moflowgan: Video Generation With Flow Guidance". In: *2020 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE. 2020, pp. 1–6.

[25] Xuechen Li et al. "Scalable gradients for stochastic differential equations". In: *arXiv preprint arXiv:2001.01328* (2020).

[26] Yijun Li et al. "Flow-Grounded Spatial-Temporal Video Prediction from Still Images". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Sept. 2018.

[27] Xiaodan Liang et al. "Dual Motion GAN for Future-Flow Embedded Video Prediction". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Oct. 2017.

[28] Ziwei Liu et al. "Video frame synthesis using deep voxel flow". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 4463–4471.

[29] Pauline Luc et al. "Transformation-based adversarial video prediction on large-scale data". In: *arXiv preprint arXiv:2003.04035* (2020).

[30] Takeru Miyato et al. "Spectral Normalization for Generative Adversarial Networks". In: *International Conference on Learning Representations*. 2018.

[31] Yipeng Qin, Niloy Mitra, and Peter Wonka. *How does Lipschitz Regularization Influence GAN Training?* 2018. arXiv: `1811.09567 [cs.CV]`.

[32] Alec Radford, Luke Metz, and Soumith Chintala. "Unsupervised representation learning with deep convolutional generative adversarial networks". In: *arXiv preprint arXiv:1511.06434* (2015).

[33] MarcAurelio Ranzato et al. "Video (language) modeling: a baseline for generative models of natural videos". In: *arXiv preprint arXiv:1412.6604* (2014).

[34] Masaki Saito, Eiichi Matsumoto, and Shunta Saito. "Temporal generative adversarial nets with singular value clipping". In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2830–2839.

[35] Masaki Saito et al. "Train Sparsely, Generate Densely: Memory-Efficient Unsupervised Training of High-Resolution Temporal GAN". In: ().

[36] Tim Salimans et al. "Improved techniques for training gans". In: *Advances in neural information processing systems*. 2016, pp. 2234–2242.

[37] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. "Unsupervised learning of video representations using lstms". In: *International conference on machine learning*. 2015, pp. 843–852.

[38] Du Tran et al. "Learning spatiotemporal features with 3d convolutional networks". In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 4489–4497.

[39] Sergey Tulyakov et al. "Mocogan: Decomposing motion and content for video generation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 1526–1535.

[40] Belinda Tzen and Maxim Raginsky. "Neural stochastic differential equations: Deep latent gaussian models in the diffusion limit". In: *arXiv preprint arXiv:1905.09883* (2019).

[41] Ruben Villegas et al. "High fidelity video prediction with large stochastic recurrent neural networks". In: *Advances in Neural Information Processing Systems*. 2019, pp. 81–91.

[42] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. "Generating videos with scene dynamics". In: *Advances in neural information processing systems*. 2016, pp. 613–621.

[43] SHI Xingjian et al. "Convolutional LSTM network: A machine learning approach for precipitation nowcasting". In: *Advances in neural information processing systems*. 2015, pp. 802–810.

[44] Cagatay Yildiz, Markus Heinonen, and Harri Lahdesmaki. "ODE2VAE: Deep generative second order ODEs with Bayesian neural networks". In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach et al. Curran Associates, Inc., 2019, pp. 13412–13421. URL: `http://papers.nips.cc/paper/9497-ode2vae-deep-generative-second-order-odes-with-bayesian-neural-networks.pdf`.