Wildfire Forecasting with Satellite Images and Deep Generative Model

Thai Nam Hoang[1, 2], **Sang T. Truong**[3], **Chris Schmidt**[2]

[1] Beloit College
[2] SSEC, University of Wisconsin, Madison
[3] Stanford University
thoang5@wisc.edu, sttruong@cs.stanford.edu, chris.schmidt@ssec.wisc.edu

Abstract

Wildfire prediction has been one of the most critical tasks that humanities want to thrive at. While it plays a vital role in protecting human life, it is also difficult because of its stochastic and chaotic properties. We tackled the problem by interpreting a series of wildfire images into a video and used it to anticipate how the fire would behave in the future. However, creating video prediction models that account for the inherent uncertainty of the future is challenging. The bulk of published attempts are based on stochastic image-autoregressive recurrent networks, which raise various performance and application difficulties such as computational cost and limited efficiency on massive datasets. Another possibility is to use entirely latent temporal models that combine frame synthesis with temporal dynamics. However, due to design and training issues, no such model for stochastic video prediction has yet been proposed in the literature. This paper addresses these issues by introducing a novel stochastic temporal model whose dynamics are driven in a latent space. It naturally predicts video dynamics by allowing our lighter, more interpretable latent model to beat previous state-of-the-art approaches on the GOES-16 dataset. Results are compared using various benchmarking models.

Introduction

Weather forecasting has been an essential task of humankind. Since 1975, the US Government has utilized the Geostationary Operational Environmental Satellite (GOES) to produce data that can enhance weather and climate models, thus allowing for more precise and quicker weather forecasting and a better knowledge of long-term climate conditions (NASA 2015; Dunbar and Garner 2021). Using informative GOES data can help with wildfire prediction and detection, as current data has shown, more and more wildfires are happening. The increase in severity, frequency, and duration of wildfires brought on by anthropogenic climate change and rising global temperatures has resulted in the emission of significant amounts of greenhouse gases, the destruction of forests and their related habitats, as well as damage to infrastructure and property (Marlon et al. 2012; Abatzoglou and Williams 2016; Vilà-Vilardell et al. 2020).

Even though previous projects on fire detection have been done, most of them attempted to tackle segmentation (Khyrashchev and Larionov 2020; Pereira et al. 2021; Zhang et al. 2021). Although these methods may not serve the whole idea of wildfire prediction and prevention, they tend to recognize the fire once it has happened, and the larger it grows, the easier they can segment. Additionally, they have only tried to use traditional rasterized satellite images in which images are patched into traditional RGB 3-channel. Therefore they could not capture very early predictors of small wildfire.

An advantage of GOES compared to other satellites is that GOES uses an Advanced Baseline Imager (ABI) that takes the image of the Earth with 16 spectral bands (two visible channels, four near-infrared channels, and ten infrared channels) with a fast scan time of 12 slices per hour and a higher resolution of 0.5-2km (Schmit et al. 2017; NOAA 2017b). We can utilize the robustness of ABI images to create a temporal-like dataset that serves as baseline data for making predictions.

For the purpose of synthesizing images, generating adversarial networks (GANs) can be taken into account. The network introduces a generator and discriminator for unsupervised adversarial training, which indirectly "learns" the dataset through a minimax game (Goodfellow et al. 2014). The discriminator distinguishes genuine pictures from a training set from synthetic phony ones created by the generator. Starting from this idea, we can evolve to generating video frames instead of a single image. This idea can be classified as stochastic video prediction. However, it is a daunting task as most approaches are usually based on image-autoregressive models (Babaeizadeh et al. 2017; Denton and Fergus 2018; Weissenborn, Täckström, and Uszkoreit 2019), which were pixel-wise tackles built around Recurrent Neural Networks (RNNs) where each generated frame is fed back into the model to produce the next frame. The performance of this approach, however, relies heavily on the capability of its encoders and decoders, as each generated frame has to be re-encoded in a latent space. Such techniques may have negative impact on performance and have limited application, especially when dealing with massive amounts of data (Gregor et al. 2018; Rubanova, Chen, and Duvenaud 2019).

Another technique is to separate the dynamic of the state representations from the produced frames, which are decoded separately from the latent space. This becomes computationally interesting when combined with a low-dimensional latent space and eliminating the relationship mentioned above between frame generation and temporal dynamics. Furthermore, such models are more interpretable than autoregressive models and may be used to create a complete representation of a system's state, for example, in reinforcement learning applications (Gregor et al. 2018). These State-Space Models (SSMs), however, are more challenging to train since they need non-trivial inference systems (Krishnan, Shalit, and Sontag 2016) and careful dynamic model construction (Karl et al. 2016). As a result, most effective SSMs are only assessed on minor or contrived toy tasks.

In this paper, we present a novel stochastic dynamic model for video prediction that successfully harnesses the structural and computational advantages of SSMs operating on low-dimensional latent spaces. Its dynamic component governs the system's temporal evolution through residual updates of the latent state, which are conditioned on learned stochastic variables. This approach enables us to execute an efficient training strategy and analyze complex high-dimensional data such as movies in an interpretable manner. This residual principle is related to recent breakthroughs in the relationship between residual networks and Ordinary Differential Equations (ODEs). As illustrated in our research, this interpretation offers additional possibilities, such as creating videos at varied frame rates. As evidenced by comparisons with competing baselines on relevant benchmarks, the proposed technique outperforms existing state-of-the-art models on the task of stochastic video prediction.

Related Works

Video synthesis encompasses a wide range of tasks, from super-resolution (Caballero et al. 2016), interpolation between distant frames (Jiang et al. 2017), generation (Tulyakov et al. 2017), video-to-video translation (Wang et al. 2018), and conditioning video prediction, which is the subject of this study.

Deterministic models

Beginning with RNN-based sequence generating models (Graves 2013), a variety of video prediction algorithms based on LSTMs (Long Short-Term Memory networks (Hochreiter and Schmidhuber 1997)) and its convoluted variation of ConvLSTMs (Shi et al. 2015) were developed (Srivastava, Mansimov, and Salakhutdinov 2015; De Brabandere et al. 2016; Wichers et al. 2018; Jin et al. 2020). Computer vision algorithms are indeed frequently aimed at high-dimensional video sequences and employ domain-specific approaches such as pixel-level transformations, and optical flow (Walker, Gupta, and Hebert 2015; Walker et al. 2016; Vondrick and Torralba 2017; Lu, Hirsch, and Scholkopf 2017; Fan, Zhu, and Yang 2019) to assist the generation of high-quality predicting outputs. Such algorithms are deterministic, limiting their efficacy by failing to produce high-quality long-term video frames (Babaeizadeh et al. 2017; Denton and Fergus 2018). Another approach is to apply adversarial losses (Goodfellow et al. 2014) in sharpening the resulting frames (Vondrick and Torralba 2017; Lu, Hirsch, and Scholkopf 2017; Wu et al. 2020). Adversarial losses, conversely, are famously challenging to train. As a result, mode collapse develops, restricting generational diversity.

Stochastic and image-autoregressive models

Other methods manipulate pixel-level autoregressive generation and concentrate on precise probability maximization (Oord et al. 2016; Kalchbrenner et al. 2016; Weissenborn, Täckström, and Uszkoreit 2020). Flow normalization has also been studied using invertible transformations between the observation and latent spaces (Kingma and Dhariwal 2018; Kumar et al. 2019). However, they necessitate the careful construction of sophisticated temporal production systems that manage high-dimensional data, resulting in exorbitant temporal generation costs. For the inference of low-dimensional latent state variables, Variational Auto-encoders are utilized in more efficient continuous models (VAEs (Kingma and Welling 2013)). Stochastic variables were integrated into ConvLSTM in (Babaeizadeh et al. 2017). In order to sample random variables that are supplied to a predictor LSTM, both (He et al. 2018) and (Denton and Fergus 2018) utilized a prior LSTM conditioned on previously produced frames. Finally, (Lee et al. 2018) merged the ConvLSTM with learned prior, sharpening the resulting videos with an adversarial loss. However, all of these approaches are image-autoregressive in that they feed their predictions back into the latent space, connecting the frame synthesis and temporal models together and increasing their computing cost. (Minderer et al. 2019) offered an autoregressive VRNN model based on learned image key points rather than raw frames, which is similar to our approach. It is unknown to what degree this adjustment will alleviate the issues mentioned above, instead, we address these concerns by focusing on video dynamics and proposing a state-space model that operates on a limited latent space.

State-space model

Numerous latent state-space models, often trained using deep variational inference, have been suggested for sequence modelization (Bayer and Osendorfer 2014; Fraccaro et al. 2016; Hafner et al. 2018). These initiatives, which employ locally linear or RNN-based dynamics, are intended for low-dimensional data since learning such models on complex data is either difficult or it concentrates on control or planning tasks. On the other hand, the utterly latent technique is the first to have been successfully applied to complex high-dimensional data such as videos due to a temporal model based on residual updates of its latent state. It is part of a recent development that connects differential equations with neural networks (Lu et al. 2017; Long et al. 2017), leading to the integration of ODEs, which are seen as continuous residual networks (He et al. 2016). On

the other hand, follow-ups and similar research are confined to low-dimensional data, prone to overfitting, and unable to manage stochasticity within a sequence. Another line of research examines stochastic differential equations using neural networks (Ryder et al. 2018; De Brouwer et al. 2019), but is confined to continuous Brownian noise, whereas video generation also involves the modeling of punctual stochastic events.

Methods

We are concerned with the challenge of stochastic video prediction, attempting to forecast future frames of a video given the initial conditioning frames.

Latent Residual Dynamic Model

Let $x_{0:T} = \{x_0, x_1, \ldots, x_{T-1}\}$ be a sequence of T video frames, where each state $x_t \in \mathbb{R}^{b\times m\times n}$ is a satellite image, where $b = 10$ is the number of band ("channel"), and $m = 1500$ and $n = 2500$ are the maximum image sizes. We want to generate $x_{T:T+h}$, where h is the forecasting horizon. One way to achieve this goal is to use a parameterized autoregressive model f_θ that maps one state to another: $x_{t+1} = f_\theta(x_t)$. We introduce latent variables y driven by a dynamic temporal model to achieve this. Each frame x_t is then solely generated from the corresponding latent state y_t, making the dynamics independent from the previously generated frames.

Based on (Franceschi et al. 2020), we suggest using a stochastic residual network to describe the transition function of the latent dynamic of y. State y_{t+1} is selected to be deterministically dependent on the preceding state y_t and conditionally dependent on an auxiliary random variable z_{t+1}. These auxiliary variables represent the video dynamics' unpredictability. They have a learned factorized Gaussian prior that is solely affected by the initial state. The model is depicted in Figure (1) and defined as follows:

$$\begin{cases} y_1 \sim \mathcal{N}(0, I), \\ z_{t+1} \sim \mathcal{N}(\mu_\theta(y_t), \sigma_\theta(y_t)I), \\ y_{t+1} = y_t + f_\theta(y_t, z_{t+1}), \\ x_t \sim \mathcal{G}(g_\theta(y_t)) \end{cases} \tag{1}$$

where $\mu_\theta, \sigma_\theta, f_\theta, g_\theta$ are neural nets, and $\mathcal{G}(g_\theta(y_t))$ is a probability distribution parameterized by $g_\theta(y_t)$. Note that y_1 is assumed to have a standard Gaussian prior and, in our VAE setting, will be inferred from conditioning frames for the prediction.

The residual update rule is based on the Euler Differentiation Technique for differential equations. The state of the system y_t is updated by its first-order movement, i.e, the residual $f_\theta(y_t, z_{t+1})$. This fundamental idea makes our temporal model lighter and more interpretable than a normal RNN. Equation (1), on the other hand, differs from a discretized ODE because of the introduction of the stochastic discrete-time variable z. Nonetheless, we recommend that the Euler

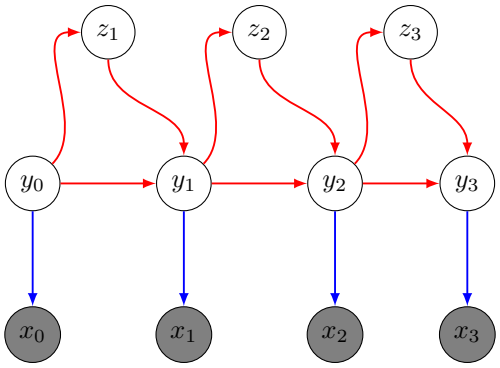

Figure 1: Generative model p

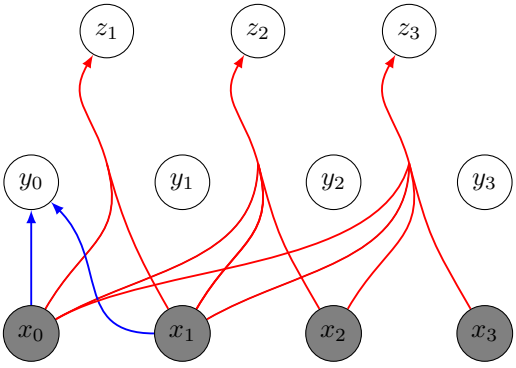

Figure 2: Inference model q

step size Δt always be less than 1 to get the temporal model closer to continuous dynamics. With $\frac{1}{\Delta t} \in \mathbb{N}$ to synchronize the step size with the video frame rate, the updated dynamics are as follows:

$$y_{t+\Delta t} = y_t + \Delta t \cdot f_\theta(y_t, z_{\lfloor t \rfloor + 1}) \tag{2}$$

The auxiliary variable z_t is held constant between two integer time steps in the formulation. It should be noted that during training or testing, a different Δt might be utilized. Due to the chance that each intermediate latent state may be decoded in the observation space, our model can generate videos at any frame rate. This capacity allows us to assess the learned dynamic's quality while challenging its ODE inspiration by evaluating its generalization to the continuous limit. For the rest of this section, we consider that $\Delta t = 1$ generalization to a smaller Δt is straightforward as Figure (1) remains unchanged.

Content Variable

Some video sequence components, such as the terrain, can be static or include very slightly variations, such as moving clouds. Since they may not affect the dynamics thus we model them separately, as (Denton and Birodkar 2017; Yingzhen and Mandt 2018) have done. We calculate a content variable w that remains constant throughout the generation process and is passed into the frame generator along

with y_t. It allows the dynamical element of the model to solely concentrate on movement, making it lighter and increasing stability. Furthermore, it enables us to use architectural developments in neural networks, such as skip connections (Ronneberger, Fischer, and Brox 2015), to generate more realistic frames.

This content variable is a deterministic function c_ψ of a fixed number $k < T$ of frames $x_c^{(k)} = (x_{i_0}, x_{i_2}, \dots, x_{i_k})$:

$$\begin{cases} w = c_\psi(x_c^{(k)}) = c_\psi(x_{i_0}, x_{i_2}, \dots, x_{i_k}) \\ x_t \sim \mathcal{G}(g_\theta(y_t)) \end{cases} \tag{3}$$

During testing, $x_c^{(k)}$ are the last k conditioning frames.

In contrast to the dynamic variables y and z, this content variable has no probabilistic prior. As a result, the information it carries is only confined in the structure rather than in the loss function as well. To avoid leaking temporal information in w, we propose sampling these k frames evenly inside $x_{0:T}$ during training. We also built c_ψ as a permutation invariant function (Zaheer et al. 2017) composed of an MLP fed with the sum of each frame representation, as shown in (Santoro et al. 2017).

Due to the absence of prior and architectural limitations, w can contain as much non-temporal information as feasible while still excluding dynamic information. On the contrary, y and z should only include temporal information that w cannot capture owing to their high standard Gaussian priors.

This content variable may be deleted from our model, resulting in a more traditional deep state-space model.

Variational Inference and Architecture

Following the generative model depicted in Figure (1), the conditional joint probability of the full model, given a content variable w, can be written as:

$$\begin{aligned} & p(x_{0:T}, z_{1:T}, y_{0:T}|w) \\ = \; & p(y_0) \prod_{t=1}^{T} p(z_t, y_t|y_{t-1}) \prod_{t=0}^{T} p(x_t|y_t, w) \end{aligned} \tag{4}$$

with

$$p(z_t, y_t|y_{t-1}) = p(z_t|y_{t-1}) p(y_t|y_{t-1}, z_t) \tag{5}$$

According to Equation (1), $p(y_t|y_{t-1}, z_t) = \delta(y_t - y_{t-1} - f_\theta(y_{t-1}, z_t))$, where δ is the Dirac delta function centered at 0. Hence in order to optimize the likelihood of the observed videos $p(x_{0:T}|w)$, we need to infer latent variables y_0 and $z_{1:T}$. This can be done with deep variational inference using the inference model parameterized by ϕ as shown in Figure (2), which comes down to considering a variational distribution $q_{Z,Y}$ defined and factorized as follows:

$$\begin{aligned} & q_{Z,Y} \triangleq q(z_{1:T}, y_{0:T}|x_{0:T}, w) \\ = \; & q(y_0|x_{0:k}) \prod_{t=1}^{T} q(z_t|x_{0:t}) q(y_t|y_{t-1}, z_t) \\ = \; & q(y_0|x_{0:k}) \prod_{t=1}^{T} q(z_t|x_{0:t}) p(y_t|y_{t-1}, z_t) \end{aligned} \tag{6}$$

where $q(y_t|y_{t-1}, z_t) = p(y_t|y_{t-1}, z_t)$ begin the aforementioned Dirac delta function. This yields the following evidence lower bound (ELBO):

$$\begin{aligned} & \log p(x_{0:T}|w) \geq \mathcal{L}(x_{0:T}; w, \theta, \phi) \\ \triangleq \; & - D_{\mathrm{KL}}[q(y_0|x_{0:k}) \parallel p(y_0)] \\ & + \mathbb{E}_{(\tilde{z}_{1:T}, \tilde{y}_{0:T}) \sim q_{Z,Y}} \Bigg[\sum_{t=0}^{T} \log p(x_t|\tilde{y}_t, w) \\ & - \sum_{t=1}^{T} D_{\mathrm{KL}}[q(z_t|x_{0:t}) \parallel p(z_t|\tilde{y}_{t-1}] \Bigg] \end{aligned} \tag{7}$$

where D_{KL} denotes the Kullback-Leibler (KL) divergence.

The sum of KL divergence expectations implies considering the full past sequence of inferred states for each time step due to the dependence on conditionally deterministic variable $y_{1:T}$. Optimizing $\mathcal{L}(x_{0:T}; w, \theta, \psi)$ with respect to model parameter θ and variational parameters ϕ can be done efficiently by sampling a single full sequence of states from $q_{Z,Y}$ per example, and computing gradients by backpropagation (Rumelhart, Hinton, and Williams 1986), troughing all inferred variables, using reparameterization trick (Kingma and Welling 2013). We classically choose $q(y_0|x_{0:k})$ and $q(z_t|x_{0:t})$ to be factorized Gaussian so that all KL divergences can be computed analytically.

We include an L_2 regularization term on residual f_θ applied to y, which stabilizes the temporal dynamics of the residual network, as noted by (Behrmann et al. 2019; de Bézenac, Ayed, and Gallinari 2019; Rousseau, Drumetz, and Fablet 2019). Given a set of videos $\mathcal{X}$, the complete optimization problem, where $\mathcal{L}$ is defined as in Equation (7), is then given as:

$$\begin{aligned} \arg\max_{\theta, \phi, \psi} \sum_{x \in \mathcal{X}} \Bigg[& \mathbb{E}_{x_c^{(k)}} \mathcal{L}(x_{0:T}; c_\psi(x_c^{(k)}), \theta, \phi) \\ & - \lambda \mathbb{E}_{(\tilde{z}_{1:T}, \tilde{y}_{0:T}) \sim q_{Z,Y}} \sum_{t=1}^{T} \parallel f_\theta(y_{t-1, z_t}) \parallel_2 \Bigg] \end{aligned} \tag{8}$$

The first latent variables are inferred with the conditioning frames and are then predicted with the dynamic model. In contrast, each frame of the input sequence is considered for

inference during training, which is done as follows. Firstly, each frame x_t is independently encoded into a vector-valued representation $\tilde{x}_t$, with $\tilde{x}_t = h_\psi(x_t)$. y_0 is then inferred using an MLP on the first k encoded frames $\tilde{x}_{0:k}$. Each z_t is inferred in a feed-forward fashion with an LSTM on the encoded frames. Inferring z this way experimentally performs better than, e.g., inferring them from the whole sequence $x_{0:T}$; we hypothesize that this follows from the fact that this filtering scheme is closer to the prediction setting, where the future is not available.

Experiments

Training

In this section, we qualitatively investigate the dynamics and latent space learned by our model.

The stochastic nature and originality of the video prediction task make it challenging to evaluate ordinarily (Lee et al. 2018): because the task is stochastic, comparing the ground truth and a predicted video is insufficient. We, therefore, follow the standard strategy (Denton and Fergus 2018; Lee et al. 2018), which consists of sampling a specific number (here, 100 samples) of probable futures from the tested model and reporting the highest performing sample against the genuine video for each test sequence. We show this disparity for two generally used metrics that are computed frame-by-frame and averaged over time: Peak Signal-to-Noise Ratio (PSNR, *higher is better*) and Structured Similarity (SSIM, *higher is better*) (Horé and Ziou 2010). PSNR penalizes inaccuracies in projected dynamics since it is a pixel-wise measurement derived from $L2$ distance, but it may also favor fuzzy predictions. To avoid this problem, SSIM compares local frame patches, although this results in some dynamics information being lost. We considered these two measures to be complementary since they capture distinct sizes and modalities.

We present experimental results on the GOES data that is briefly presented in the following section. We also compare our model against SVG (Denton and Fergus 2018) and StructVRNN (Minderer et al. 2019). SVG has the most similar training and architecture among the models. To make fair comparisons using this technique, we employed the same neural architecture as SVG for our encoders and decoders. Unless otherwise indicated, our model is evaluated with the same Δt as in training, as shown in Equation (2).

Dataset

We used GOES-16 CONUS (ABI-L1b-RadC) infrared ABI spectral bands (band 7 - 16) (NASA 2017), specifically at night from 22:00:00 to 5:00:00 CST (UTC-5) or 5:00:00 to 12:00:00 UTC. The dataset can be pulled directly from AWS s3 (AWS 2022). We set the data from day 80 to day 135 of the year 2022, which is 56 days or eight weeks. For each band, there will be a total of 4704 slices.

Preprocessing

Initially, each slice will be 1500×2500. Since a single slice sizes around 3-4Mb, we decided to crop to 256×256 to,

Listing 1 Normalization

```
def normalization(crop: da.Array) -> da.Array:
    stack_len, _, _ = crop.shape

    dif_max = da.nanmax(crop, axis=(1, 2))
    dif_min = da.nanmin(crop, axis=(1, 2))

    new_crop_stack = []
    for i in range(stack_len):
        curr = crop[i]
        new_crop_stack.append((curr - dif_min[i]) / (dif_max[i] - dif_min[i]))

    new_crop_stack = da.stack(new_crop_stack, axis=0)
    return new_crop_stack
```

first of all, cut down the size and, as it follows, speed up the calculation. Doing so will center our focus on the Mid and Southern regions of the US where wildfires typically erupt during spring and summer. Each pixel represents the value of radiance of brightness in each band. Firstly, we have to convert the other band's radiance temperature into band 7's radiance temperature. We then calculate its brightness temperature by applying the Planck function, and the spectral bandpass correction into radiance temperature (NOAA 2017a):

$$BT = \left[\frac{fk_2}{\log\left(\frac{fk_1}{L_v} + 1\right)} - bc_1\right] \times \frac{1}{bc_2} \tag{9}$$

where L_v is the radiance, fk_1 and fk_2 are coefficients of the Planck function derived from physical constants (i.e., the speed of light, the Boltzmann constant, and the Planck constant) and the bandpass central wavenumber, and bc_1 and bc_2 are the spectral response function offset and scale correction terms. These four coefficients are included in the product metadata as variables: `planck_fk1`, `planck_fk2`, `planck_bc1`, and `planck_bc2` (NOAA 2019).

Next, we will calculate the radiance of band b in band 7 (NOAA 2017a):

$$Rad_{b\_in\_b7} = \frac{fk_1}{\exp\left(\frac{fk_2}{BT \times bc_2 + bc_1}\right) - 1} \tag{10}$$

After that, we normalized every band into the domain of $[0, 1]$ to evenly cut down the size. Notice here we used `dask.array` to store chunks instead of `numpy`, which can cause bottleneck while calculating the data (1):

We treated each band slice as a single channel for our image. Therefore a single "image" can have 10 "channels." To explore the data's stochastic characteristics, we

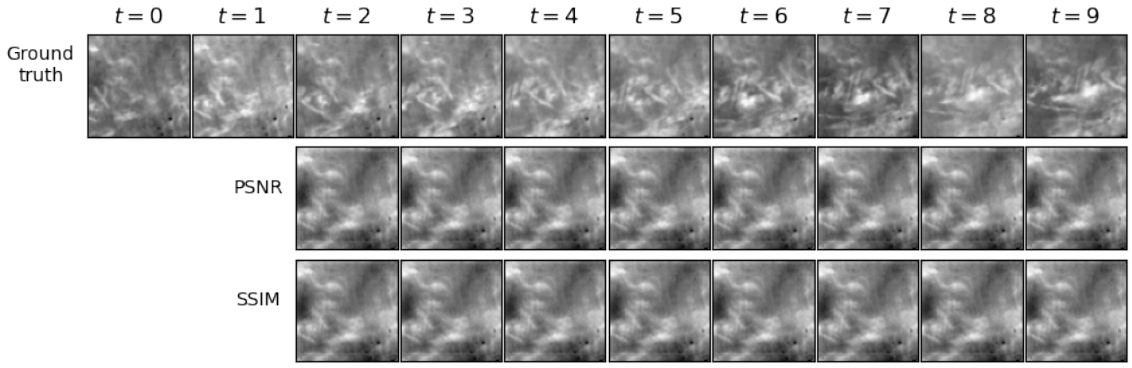

Figure 3: Ground truth (conditioning frames) and generated frames from our model.

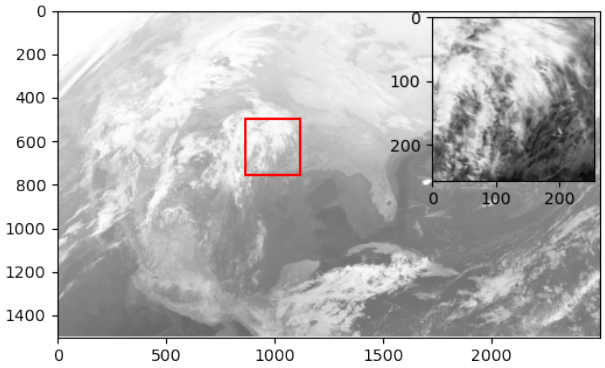

Figure 4: Band 7 and crop region

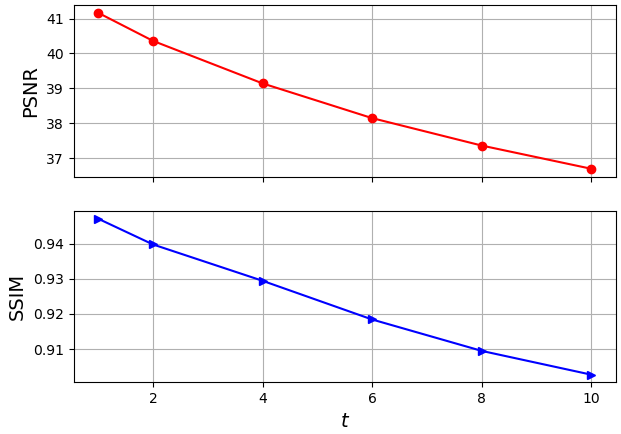

Figure 5: PSNR and SSIM scores with respect to t the dataset.

can stack them to create a "video" with 12 frames in total `video.shape = (12, 10, 256, 256)`. Finally, videos are compressed as `.npz` files to reduce the size and increase the flexibility in storing.

Results and Discussion

We trained the model on our GOES-16 dataset. The dataset is highly stochastic as the cloud can change its direction anytime. We set the size of both the state-space variable y and auxiliary variable z to 20. For the Euler step shown in Equation (2), we set it to 2. We used 2 conditioning frames to generate the next h frames, as shown in Figure (5). Our method averages 40.43 on PSNR and 0.934 on SSIM. As discussed by (Villegas et al. 2017), expanding network capacity can enhance the performance of such variational models. However, this is outside the scope of our work.

Here, we challenge the ODE inspiration of our model. Equation (2) amounts to learning a residual function $f_{z_{\lfloor t \rfloor + 1}}$ over $t \in [\lfloor t \rfloor, \lfloor t \rfloor + 1]$. We aim to test whether this dynamic is close to its continuous generalization:

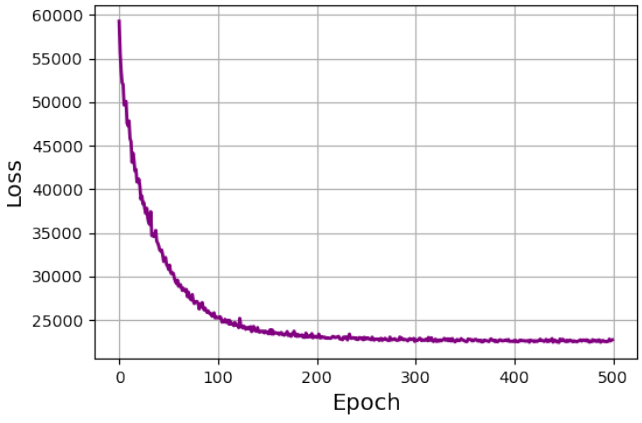

Figure 6: Loss vs. epochs over time

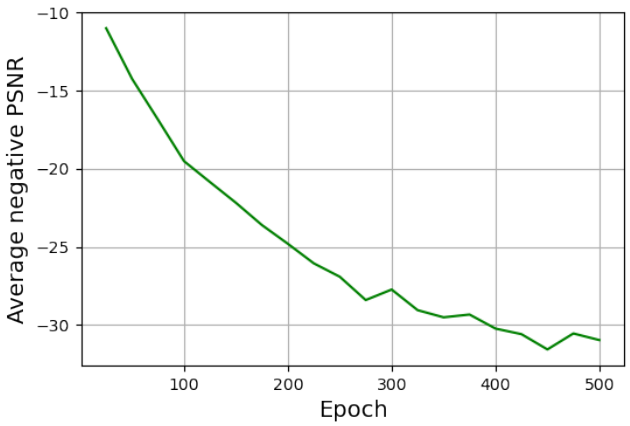

Figure 7: Average negative PSNR vs. epochs over time

$$\frac{dy}{dx} = f_{z_{\lfloor t \rfloor + 1}}(y) \tag{11}$$

which is a piecewise ODE. To this end, we refine this Euler approximation during testing by using $\frac{\Delta t}{2}$; if this maintains the performance of our model, then the dynamic rule of the latter is close to the piecewise ODE, as shown in Figure (5).

Conclusion and Future Works

We provide a unique dynamic latent model for stochastic video prediction that decouples frame synthesis and dynamics, unlike previous image-autoregressive models. This temporal model is based on residual updates of a tiny latent state and has outperformed RNN-based models. This confers numerous desired qualities for our strategy, including temporal economy and latent space interpretability. We empirically illustrate the proposed model's performance and benefits, which beats previous state-of-the-art approaches for stochastic video prediction. To the best of our knowledge, this is the first paper to offer a latent dynamic model that scales for video prediction. The suggested model is particularly innovative compared to current work on neural networks and ODEs for temporal modeling; it is the first residual model to scale to complex stochastic data such as videos.

This study also verifies the model's effectiveness on wildfire prediction where high sensitivity is required and early prediction could be obtained. Experiments showed that the proposed architecture achieved relatively competitive reconstructed accuracy and reliable recognition. However, there might still be limitations in terms of comprehensive wildfire prediction even though fire regions could be synthesized in precise detail. Besides, various weather conditions such as fog and snow may also hinder data capability.

The major principles of our method (state-space, residual dynamic, static content variable) may be applied to different models. We will supply a large amount of data for future studies, from GOES-16 (East) and GOES-17 (West), to increase variety in wildfire circumstances. Furthermore, instead of employing ten bands, we may enhance our general characteristic of picture slices by mixing all visible and near-infrared bands with infrared to produce a full 16-band image. This gives the output a more distinct and realistic appearance, therefore the produced frames will have more informative values.

Acknowledgements

We would like to thank all members, including students and staffs of 2022 Undergraduate Student Programmer at SSEC for helful discussions and comments, as well as William Roberts for his help in processing the GOES-16 dataset.

References

Abatzoglou, J. T.; and Williams, A. P. 2016. Impact of anthropogenic climate change on wildfire across western US forests. *Proceedings of the National Academy of Sciences*.

AWS. 2022. AWS S3 Explorer NOAA GOES-16. "Accessed: 2022-18-07".

Babaeizadeh, M.; Finn, C.; Erhan, D.; Campbell, R. H.; and Levine, S. 2017. Stochastic Variational Video Prediction.

Bayer, J.; and Osendorfer, C. 2014. Learning Stochastic Recurrent Networks.

Behrmann, J.; Grathwohl, W.; Chen, R. T. Q.; Duvenaud, D.; and Jacobsen, J.-H. 2019. Invertible Residual Networks. In Chaudhuri, K.; and Salakhutdinov, R., eds., *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, 573–582. PMLR.

Caballero, J.; Ledig, C.; Aitken, A.; Acosta, A.; Totz, J.; Wang, Z.; and Shi, W. 2016. Real-Time Video Super-Resolution with Spatio-Temporal Networks and Motion Compensation.

De Brabandere, B.; Jia, X.; Tuytelaars, T.; and Van Gool, L. 2016. Dynamic Filter Networks.

De Brouwer, E.; Simm, J.; Arany, A.; and Moreau, Y. 2019. GRU-ODE-Bayes: Continuous modeling of sporadically-observed time series.

de Bézenac, E.; Ayed, I.; and Gallinari, P. 2019. Optimal Unsupervised Domain Translation.

Denton, E.; and Fergus, R. 2018. Stochastic Video Generation with a Learned Prior.

Denton, E. L.; and Birodkar, v. 2017. Unsupervised Learning of Disentangled Representations from Video. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Dunbar, B.; and Garner, R. 2021. GOES Overview and History.

Fan, H.; Zhu, L.; and Yang, Y. 2019. Cubic LSTMs for Video Prediction.

Fraccaro, M.; Sønderby, S. K.; Paquet, U.; and Winther, O. 2016. Sequential Neural Models with Stochastic Layers.

Franceschi, J.-Y.; Delasalles, E.; Chen, M.; Lamprier, S.; and Gallinari, P. 2020. Stochastic Latent Residual Video Prediction. In *Proceedings of the 37th International Conference on Machine Learning*. arXiv.

Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative Adversarial Networks.

Graves, A. 2013. Generating Sequences With Recurrent Neural Networks.

Gregor, K.; Papamakarios, G.; Besse, F.; Buesing, L.; and Weber, T. 2018. Temporal Difference Variational Auto-Encoder.

Hafner, D.; Lillicrap, T.; Fischer, I.; Villegas, R.; Ha, D.; Lee, H.; and Davidson, J. 2018. Learning Latent Dynamics for Planning from Pixels.

He, J.; Lehrmann, A.; Marino, J.; Mori, G.; and Sigal, L. 2018. Probabilistic Video Generation using Holistic Attribute Control.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778.

Hochreiter, S.; and Schmidhuber, J. 1997. Long Short-Term Memory. *Neural Computation*, 9.

Horé, A.; and Ziou, D. 2010. Image Quality Metrics: PSNR vs. SSIM. In *2010 20th International Conference on Pattern Recognition*, 2366–2369.

Jiang, H.; Sun, D.; Jampani, V.; Yang, M.-H.; Learned-Miller, E.; and Kautz, J. 2017. Super SloMo: High Quality Estimation of Multiple Intermediate Frames for Video Interpolation. *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Jin, B.; Hu, Y.; Tang, Q.; Niu, J.; Shi, Z.; Han, Y.; and Li, X. 2020. Exploring Spatial-Temporal Multi-Frequency Analysis for High-Fidelity and Temporal-Consistency Video Prediction.

Kalchbrenner, N.; Oord, A. v. d.; Simonyan, K.; Danihelka, I.; Vinyals, O.; Graves, A.; and Kavukcuoglu, K. 2016. Video Pixel Networks.

Karl, M.; Soelch, M.; Bayer, J.; and van der Smagt, P. 2016. Deep Variational Bayes Filters: Unsupervised Learning of State Space Models from Raw Data.

Khyrashchev, V.; and Larionov, R. 2020. Wildfire segmentation on satellite images using Deep Learning. *2020 Moscow Workshop on Electronic and Networking Technologies (MWENT)*.

Kingma, D. P.; and Dhariwal, P. 2018. Glow: Generative Flow with Invertible 1x1 Convolutions. In Bengio, S.; Wallach, H.; Larochelle, H.; Grauman, K.; Cesa-Bianchi, N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Kingma, D. P.; and Welling, M. 2013. Auto-Encoding Variational Bayes.

Krishnan, R. G.; Shalit, U.; and Sontag, D. 2016. Structured Inference Networks for Nonlinear State Space Models.

Kumar, M.; Babaeizadeh, M.; Erhan, D.; Finn, C.; Levine, S.; Dinh, L.; and Kingma, D. 2019. VideoFlow: A Conditional Flow-Based Model for Stochastic Video Generation.

Lee, A. X.; Zhang, R.; Ebert, F.; Abbeel, P.; Finn, C.; and Levine, S. 2018. Stochastic Adversarial Video Prediction.

Long, Z.; Lu, Y.; Ma, X.; and Dong, B. 2017. PDE-Net: Learning PDEs from Data.

Lu, C.; Hirsch, M.; and Scholkopf, B. 2017. Flexible spatio-temporal networks for video prediction. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Lu, Y.; Zhong, A.; Li, Q.; and Dong, B. 2017. Beyond Finite Layer Neural Networks: Bridging Deep Architectures and Numerical Differential Equations.

Marlon, J. R.; Bartlein, P. J.; Gavin, D. G.; Long, C. J.; Anderson, R. S.; Briles, C. E.; Brown, K. J.; Colombaroli, D.; Hallett, D. J.; Power, M. J.; Scharf, E. A.; and Walsh, M. K. 2012. Long-term perspective on wildfires in the Western USA. *Proceedings of the National Academy of Sciences*.

Minderer, M.; Sun, C.; Villegas, R.; Cole, F.; Murphy, K.; and Lee, H. 2019. Unsupervised Learning of Object Structure and Dynamics from Videos.

NASA. 2015. GOES Environmental Satellites.

NASA. 2017. ABI Bands Quick Information Guides. "Accessed: 2022-18-07".

NOAA. 2017a. GOES-R Calibration Working Group and GOES-R Series Program, (2017): NOAA GOES-R Series Advanced Baseline Imager (ABI) Level 1b Radiances.

NOAA. 2017b. Instruments: Advanced Baseline Imager (ABI).

NOAA. 2019. GOES R Series Product Definition and Users' Guide.

Oord, A. v. d.; Kalchbrenner, N.; Vinyals, O.; Espeholt, L.; Graves, A.; and Kavukcuoglu, K. 2016. Conditional Image Generation with PixelCNN Decoders.

Pereira, G. H. d. A.; Fusioka, A. M.; Nassu, B. T.; and Minetto, R. 2021. Active fire detection in Landsat-8 imagery: A large-scale dataset and a deep-learning study. *ISPRS Journal of Photogrammetry and Remote Sensing*, 178: 171–186.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation.

Rousseau, F.; Drumetz, L.; and Fablet, R. 2019. Residual Networks as Flows of Diffeomorphisms.

Rubanova, Y.; Chen, R. T. Q.; and Duvenaud, D. 2019. Latent ODEs for Irregularly-Sampled Time Series.

Rumelhart, D. E.; Hinton, G. E.; and Williams, R. J. 1986. Learning representations by back-propagating errors.

Ryder, T.; Golightly, A.; McGough, A. S.; and Prangle, D. 2018. Black-box Variational Inference for Stochastic Differential Equations.

Santoro, A.; Raposo, D.; Barrett, D. G. T.; Malinowski, M.; Pascanu, R.; Battaglia, P.; and Lillicrap, T. 2017. A simple neural network module for relational reasoning.

Schmit, T. J.; Griffith, P.; Gunshor, M. M.; Daniels, J. M.; Goodman, S. J.; and Leblair, W. J. 2017. A Closer Look at the ABI on the GOES-R Series. *Bulletin of the American Meteorological Society*, 98.

Shi, X.; Chen, Z.; Wang, H.; Yeung, D.-Y.; Wong, W.-k.; and Woo, W.-c. 2015. Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting.

Srivastava, N.; Mansimov, E.; and Salakhutdinov, R. 2015. Unsupervised Learning of Video Representations using LSTMs.

Tulyakov, S.; Liu, M.-Y.; Yang, X.; and Kautz, J. 2017. MoCoGAN: Decomposing Motion and Content for Video Generation.

Villegas, R.; Yang, J.; Hong, S.; Lin, X.; and Lee, H. 2017. Decomposing Motion and Content for Natural Video Sequence Prediction.

Vilà-Vilardell, L.; Keeton, W. S.; Thom, D.; Gyeltshen, C.; Tshering, K.; and Gratzer, G. 2020. Climate change effects on wildfire hazards in the wildland-urban-interface – blue pine forests of Bhutan. *Forest Ecology and Management*, 461.

Vondrick, C.; and Torralba, A. 2017. Generating the future with Adversarial Transformers. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Walker, J.; Doersch, C.; Gupta, A.; and Hebert, M. 2016. An Uncertain Future: Forecasting from Static Images using Variational Autoencoders.

Walker, J.; Gupta, A.; and Hebert, M. 2015. Dense Optical Flow Prediction from a Static Image.

Wang, T.-C.; Liu, M.-Y.; Zhu, J.-Y.; Liu, G.; Tao, A.; Kautz, J.; and Catanzaro, B. 2018. Video-to-Video Synthesis. In Bengio, S.; Wallach, H.; Larochelle, H.; Grauman, K.; Cesa-Bianchi, N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Weissenborn, D.; Täckström, O.; and Uszkoreit, J. 2019. Scaling Autoregressive Video Models.

Weissenborn, D.; Täckström, O.; and Uszkoreit, J. 2020. Scaling Autoregressive Video Models. In *International Conference on Learning Representations*.

Wichers, N.; Villegas, R.; Erhan, D.; and Lee, H. 2018. Hierarchical Long-term Video Prediction without Supervision.

Wu, Y.; Gao, R.; Park, J.; and Chen, Q. 2020. Future Video Synthesis with Object Motion Prediction.

Yingzhen, L.; and Mandt, S. 2018. Disentangled Sequential Autoencoder. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 5670–5679. PMLR.

Zaheer, M.; Kottur, S.; Ravanbakhsh, S.; Poczos, B.; Salakhutdinov, R.; and Smola, A. 2017. Deep Sets.

Zhang, J.; Zhu, H.; Wang, P.; and Ling, X. 2021. Att Squeeze U-Net: A lightweight network for forest fire detection and recognition. *IEEE Access*, 9.