Stochastic Video Generation with a Learned Prior: Supplementary Material

Emily Denton [1] **Rob Fergus** [1 2]

Appendix

A. Variational bound

We first review the variational lower bound on the data likelihood:

$$\begin{aligned}
\log p_\theta(\mathbf{x}) &= \log \int_{\mathbf{z}} p_\theta(\mathbf{x}|\mathbf{z}) p(\mathbf{z}) \\
&= \log \int_{\mathbf{z}} p_\theta(\mathbf{x}|\mathbf{z}) p(\mathbf{z}) \frac{q_\phi(\mathbf{z}|\mathbf{x})}{q_\phi(\mathbf{z}|\mathbf{x})} \\
&= \log \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \frac{p_\theta(\mathbf{x}|\mathbf{z}) p(\mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})} \\
&\geq \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \log \frac{p_\theta(\mathbf{x}|\mathbf{z}) p(\mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})} \\
&= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) - \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p(\mathbf{z})} \\
&= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) - D_{KL}(q_\phi(\mathbf{z}|\mathbf{x}) || p(\mathbf{z}))
\end{aligned}$$

Recall that the SVG frame predictor is parameterized by a recurrent neural network. At each time step the model takes as input $\mathbf{x}_{t-1}$ and $\mathbf{z}_t$ and through the recurrence the model also depends on $\mathbf{x}_{1:t-2}$ and $\mathbf{z}_{1:t-1}$. Then, we can further simplify the bound with:

$$\begin{aligned}
\log p_\theta(\mathbf{x}|\mathbf{z}) &= \log \prod_t p_\theta(\mathbf{x}_t | \mathbf{x}_{1:t-1}, \mathbf{z}_{1:T}) \\
&= \sum_t \log p_\theta(\mathbf{x}_t | \mathbf{x}_{1:t-1}, \mathbf{z}_{1:t}, \cancel{\mathbf{z}_{t+1:T}}) \\
&= \sum_t \log p_\theta(\mathbf{x}_t | \mathbf{x}_{1:t-1}, \mathbf{z}_{1:t})
\end{aligned}$$

Recall, the inference network used by SVG-FP and SVG-LP is parameterized by a recurrent neural network that outputs a different distribution $q_\phi(\mathbf{z}_t|\mathbf{x}_{1:t})$ for every time step t. Let $\mathbf{z} = [\mathbf{z}_1, ..., \mathbf{z}_T]$ denote the collection of latent variables across all time steps and $q_\phi(\mathbf{z}|\mathbf{x})$ denote the distribution over $\mathbf{z}$. Due to the independence across time, we have

$$q_\phi(\mathbf{z}|\mathbf{x}) = \prod_t q_\phi(\mathbf{z}_t|\mathbf{x}_{1:t})$$

The independence of $\mathbf{z}_1, ..., \mathbf{z}_T$ allows the D_{KL} term of the loss to be decomposed into individual time steps:

$$\begin{aligned}
&D_{KL}(q_\phi(\mathbf{z}|\mathbf{x}) || p(\mathbf{z})) \\
&= \int_{\mathbf{z}} q_\phi(\mathbf{z}|\mathbf{x}) \log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p(\mathbf{z})} \\
&= \int_{\mathbf{z}_1} ... \int_{\mathbf{z}_T} q_\phi(\mathbf{z}_1|\mathbf{x}_1) ... q_\phi(\mathbf{z}_T|\mathbf{x}_{1:T}) \log \frac{q_\phi(\mathbf{z}_1|\mathbf{x}_1) ... q_\phi(\mathbf{z}_t|\mathbf{x}_{1:T})}{p(\mathbf{z}_1) ... p(\mathbf{z}_T)} \\
&= \int_{\mathbf{z}_1} ... \int_{\mathbf{z}_T} q_\phi(\mathbf{z}_1|\mathbf{x}_1) ... q_\phi(\mathbf{z}_T|\mathbf{x}_{1:T}) \sum_t \log \frac{q_\phi(\mathbf{z}_t|\mathbf{x}_{1:t})}{p(\mathbf{z}_t)} \\
&= \sum_t \int_{\mathbf{z}_1} ... \int_{\mathbf{z}_T} q_\phi(\mathbf{z}_1|\mathbf{x}_1) ... q_\phi(\mathbf{z}_T|\mathbf{x}_{1:T}) \log \frac{q_\phi(\mathbf{z}_t|\mathbf{x}_{1:t})}{p(\mathbf{z}_t)}
\end{aligned}$$

And because $\int_x p(x) = 1$ this simplifies to:

$$\begin{aligned}
&= \sum_t \int_{\mathbf{z}_t} q_\phi(\mathbf{z}_t|\mathbf{x}_{1:t}) \log \frac{q_\phi(\mathbf{z}_t|\mathbf{x}_{1:t})}{p(\mathbf{z}_t)} \\
&= \sum_t D_{KL}(q_\phi(\mathbf{z}_t|\mathbf{x}_{1:t}) || p(\mathbf{z}_t))
\end{aligned}$$

Putting this all together we have:

$$\begin{aligned}
\log p_\theta(\mathbf{x}) &\geq \mathcal{L}_{\theta,\phi}(\mathbf{x}_{1:T}) \\
&= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) - D_{KL}(q_\phi(\mathbf{z}|\mathbf{x}) || p(\mathbf{z})) \\
&= \sum_t \big[\mathbb{E}_{q_\phi(\mathbf{z}_{1:t}|\mathbf{x}_{1:t})} \log p_\theta(\mathbf{x}_t | \mathbf{x}_{1:t-1}, \mathbf{z}_{1:t}) \\
&\qquad - D_{KL}(q_\phi(\mathbf{z}_t|\mathbf{x}_{1:t}) || p(\mathbf{z}_t)) \big]
\end{aligned}$$

B. Additional results

Stochastic Moving MNIST

In Section 4.2 we introduce the Stochastic Moving MNIST dataset. This dataset contains videos of MNIST digits bouncing around the frame. Digits moves with a constant velocity along a trajectory until they hit at wall at which point they bounce off with a random speed and direction.

[1] New York University [2] Facebook AI Research. Correspondence to: Emily Denton <denton@cs.nyu.edu>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018. Copyright 2018 by the author(s).

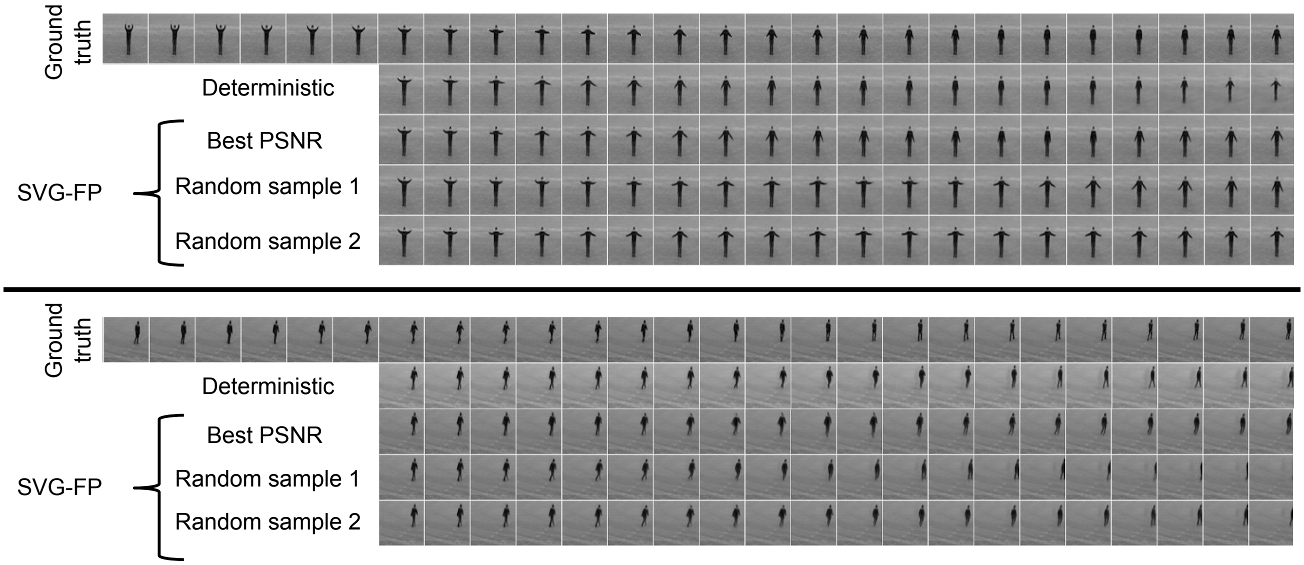

Figure 1. Qualitative comparison between SVG-LP and a purely deterministic baseline. Both models were conditioned on the first 10 frames (the final 5 are shown in the figure) of test sequences.

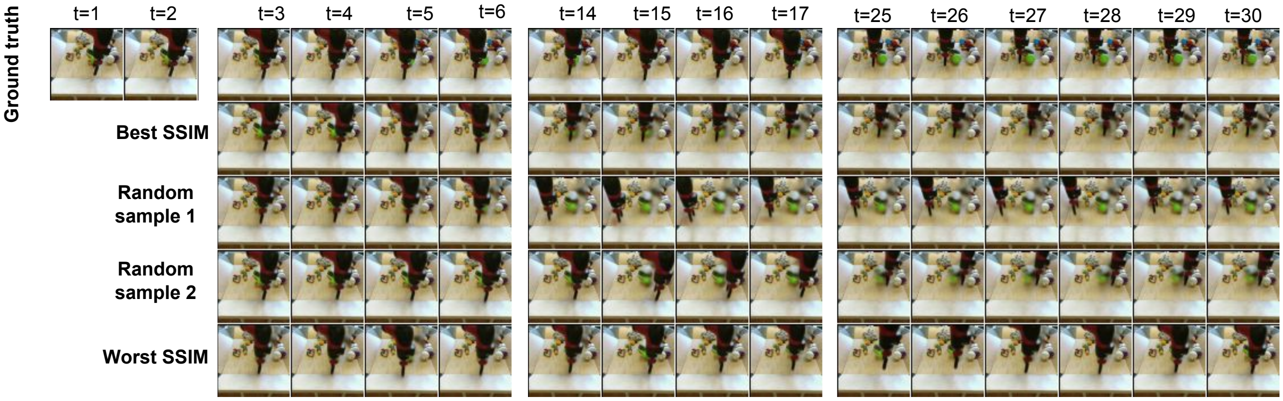

Figure 2. Additional examples of generations from SVG-LP showing crisp and varied predictions. A large segment of the background is occluded in conditioning frames, preventing SVG-LP from directly copying these background pixels into generated frames. In addition to crisp robot arm movement, SVG-LP generates plausible background objects in the space occluded by the robot arm in initial frames.

The experiments in Section 4.2 assumed a uniform distribution of digit motion and speed. Here, we evaluate SVG-LP on a more challenging, non-uniform, distribution of trajectories. Fig. 3 plots the distribution of Δx and Δy from which velocity vectors are initially sampled at the start of a video sequence. All subsequent velocity vectors are sampled from a modified variant of this distribution where invalid directions are given zero probability and the remaining probabilities are re-normalized. Note that depending which wall the digit hits, a different subset of velocity vectors will be valid (e.g. if the digit hits the right wall, $\Delta x > 0$ would be invalid) and so the distribution is dependent on the precise location the digits hits the wall.

We trained SVG-LP on this non-uniform SM-MNIST dataset and assessed the model's ability to capture the digit trajectory using the same technique described in Section 4.2. Fig. 4 shows SVG-LP accurately capturing the distribution of MNIST digit trajectories for many time steps. The digit trajectory is deterministic before a collision. This is accurately reflected by the highly peaked distribution of velocity vectors from SVG-LP in the time steps leading up to a collision. Following a collision, the distribution broadens and effectively captures the complex trajectory distribution for many time steps.

KTH

Fig. 1 shows additional generations from the SVG-FP model

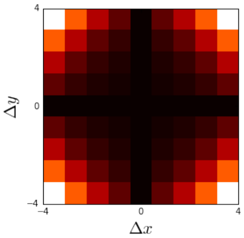

Figure 3. Initial distribution of Δx / Δy.

and a deterministic baseline. The deterministic model produces plausible predictions for the future frames but frequently mispredicts precise limb locations. In contrast, different samples from SVG-FP reflect the variability of the persons pose in future frames. By picking the sample with the best PSNR, SVG-FP closely matches the ground truth sequence.

BAIR robot pushing dataset

Fig. 2 shows sample generations from the SVG-LP model up to 30 timesteps alongside ground truth video frames.

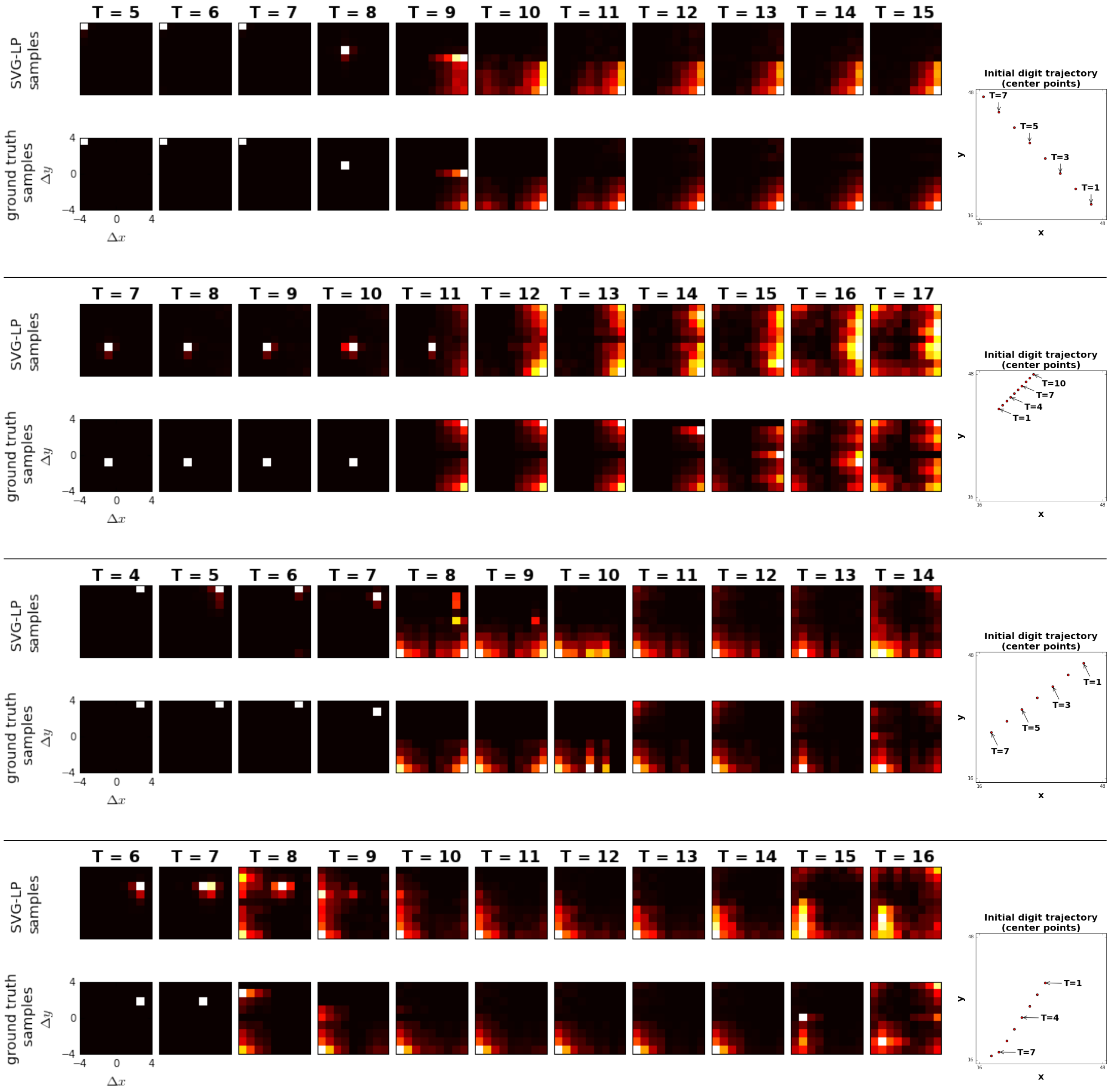

Figure 4. Four examples of our SVG-LP model accurately capturing the distribution of MNIST digit trajectories following collision with a wall. Digit trajectory velocity vectors are sampled from a *non-uniform* distribution with higher probability given to greater speeds. On the right we show the trajectory of a digit prior to the collision. Each of the sub-plots shows the *distribution* of $\Delta x, \Delta y$ at each time step. In the lower ground truth sequence, the trajectory is deterministic before the collision (occurring between $t = 8$ and $t = 9$ in the first example), corresponding to a delta-function. Following the collision, the distribution broadens out and is eventually reshaped by subsequent collisions. The upper row shows the distribution estimated by our SVG-LP model (after conditioning on ground-truth frames from $t = 1 \ldots 5$). Note how our model accurately captures the correct distribution many time steps into the future, despite its complex shape. The distribution was computed by drawing many samples from the model, as well as averaging over different digits sharing the same trajectory. The remaining examples show different trajectories with correspondingly different impact times