(Re)Live Photos: Generating Videos with Neural Networks

Michael Chen & Sang Goo Kang & Sam Kim

Computer Science Department

Objective

Given the abundance of unlabeled video data in the world, we develop a system to learn from these videos in an unsupervised manner. We present several methods to generate videos using CNNs, RNNs, and GANs. We also evolve these methods to 'predict' futures for photos by conditioning the inputs to these generators with the first frame of the generated video.

Setup

We constrain our problem to generating videos with 32 frames of 64x64 images (for GPU memory)
We train on the Aslan and UCF-101 video dataset, which contain 3697 and 13317 videos respectively.
We also train on a subset of data from Vondrick et al.

Wasserstein GAN

GANs are models that try to learn the distribution of real data by minimizing f-divergences. The Wasserstein GAN has been shown empirically to converge in a more stable manner than traditional GANs and can produce more realistic outputs. Therefore, we use WGANs for optimizing the loss in our architecture.

Discriminator maximizes:

$$\max_{w \in W} \left[\mathbb{E}_{x \sim \mathbb{P}}[D_w(x)] - \mathbb{E}_{z \sim p(z)}[D_w(g_\theta(z))] \right]$$

Generator minimizes:

$$\min_{\theta} -\mathbb{E}_{z \sim p(z)}[D_w(g_\theta(z))]$$

Architectures

Wesserstein GAN

- Latent code size: 2x4x4x512
- Generator: Five 4x4x4 3D convolutions, with stride 2. Each layer is followed by a batch normalization and ReLU activation.
- Discriminator: Five 4x4x4 3D deconvolutions with stride 2. Each layer is followed by batch normalization and leaky ReLU activation.

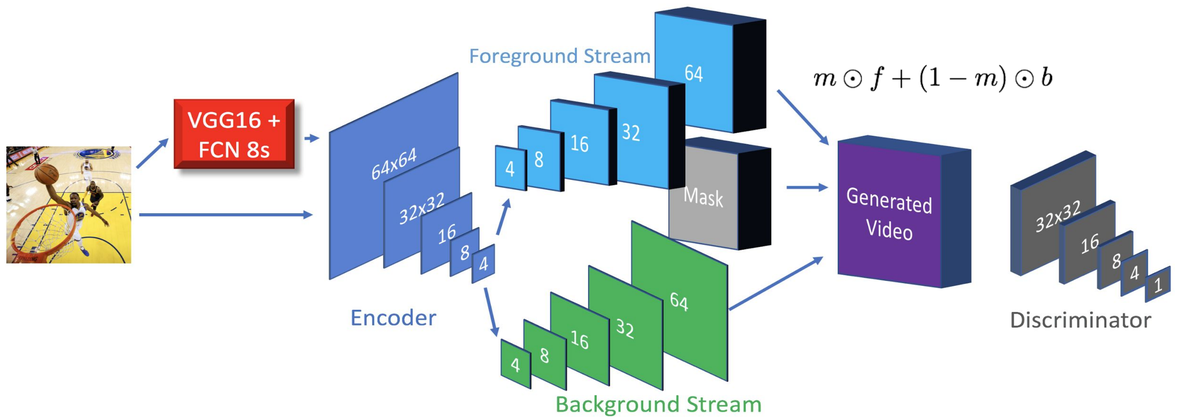
Conditional WGAN

- Conditional Generator: Contains additional five layer encoder network that convolves middle frame down to latent code size.
- Conditional Discriminator: Concatenates middle frame along with generated video as input.

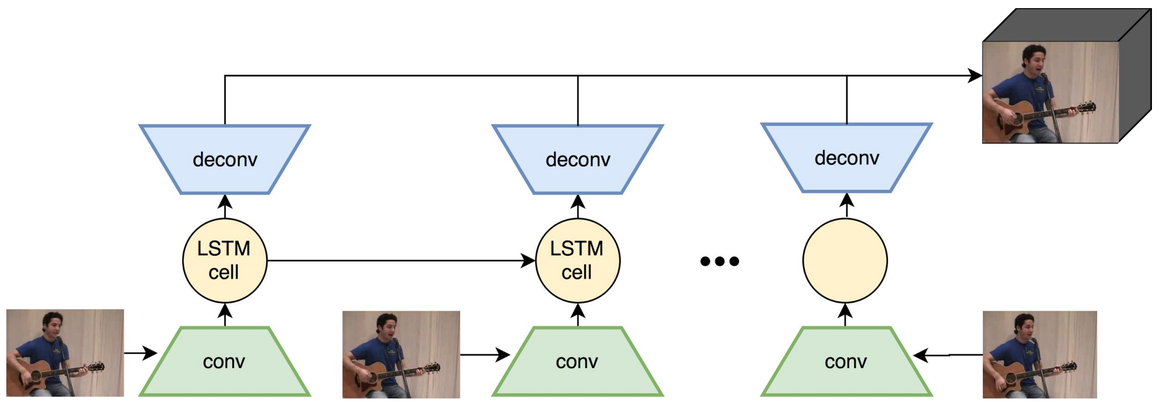
LSTM network

- Input vector size: 1024
- L2 loss function.

Conditional Generative Adversarial Network

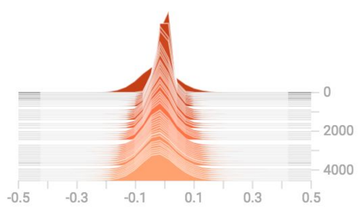


LSTM Network

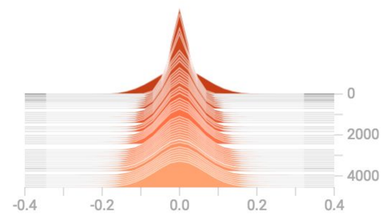


Kernel Histograms

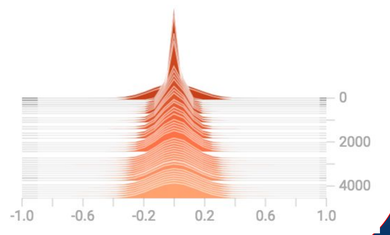
summary/generator/foreground_stream/ger kernel_0



summary/generator/background_stream/ noise_up/kernel_0



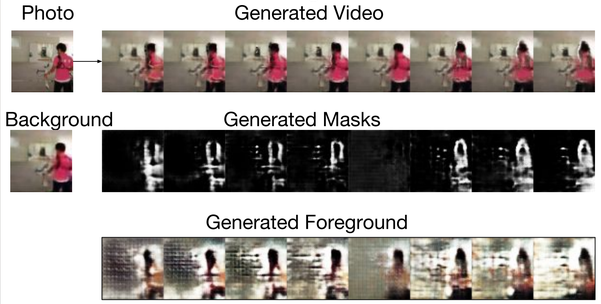
summary/discriminator/disc3/kernel_0



Results

Conditional GAN Examples

Given the middle frame, the GAN is used to predict the surrounding frames.

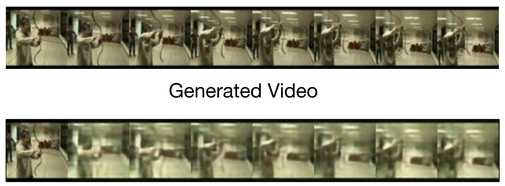


Hallucinating Videos with GANs

After training the GAN, we can map random normal distributions to the space of possible videos by running the generator network forward.

Next Frame Prediction with LSTMs

Each frame is convolved and flattened down to an input vector of size 1024. Each output vector is deconvolved to produce the next frame of the generated video.



Generated Videos: relive-photos.firebaseapp.com