Generating videos by traversing image manifolds learned by GANs

João Monteiro, **Isabela Albuquerque**, and Tiago Falk

Institut National de la Recherche Scientifique - Montréal

isabela.albuquerque@emt.inrs.ca

Outline

- Introduce a two-step approach aiming to train a generative model of natural scenes
  - Decouple frames content and time coherence by training one model for each aspect

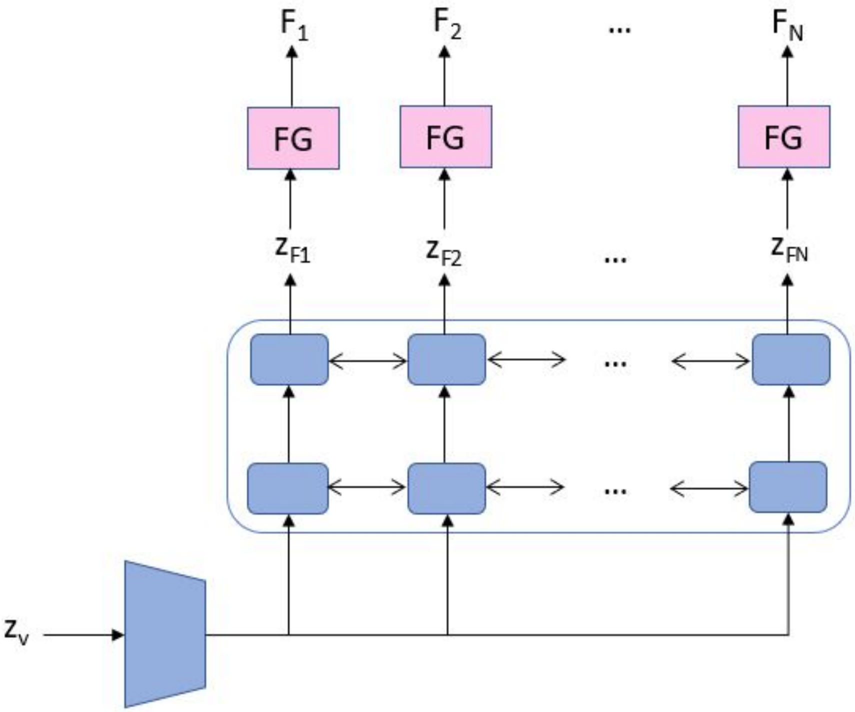
  - Content quality is ensured by a generative model of frames

  - Then a recurrent model is trained to “navigate” in the latent space yielding time-coherent video samples

- Application of the proposed framework to reconstruct fast imaging data
  - Ensure frame quality first

  - Temporal coherence is learned later

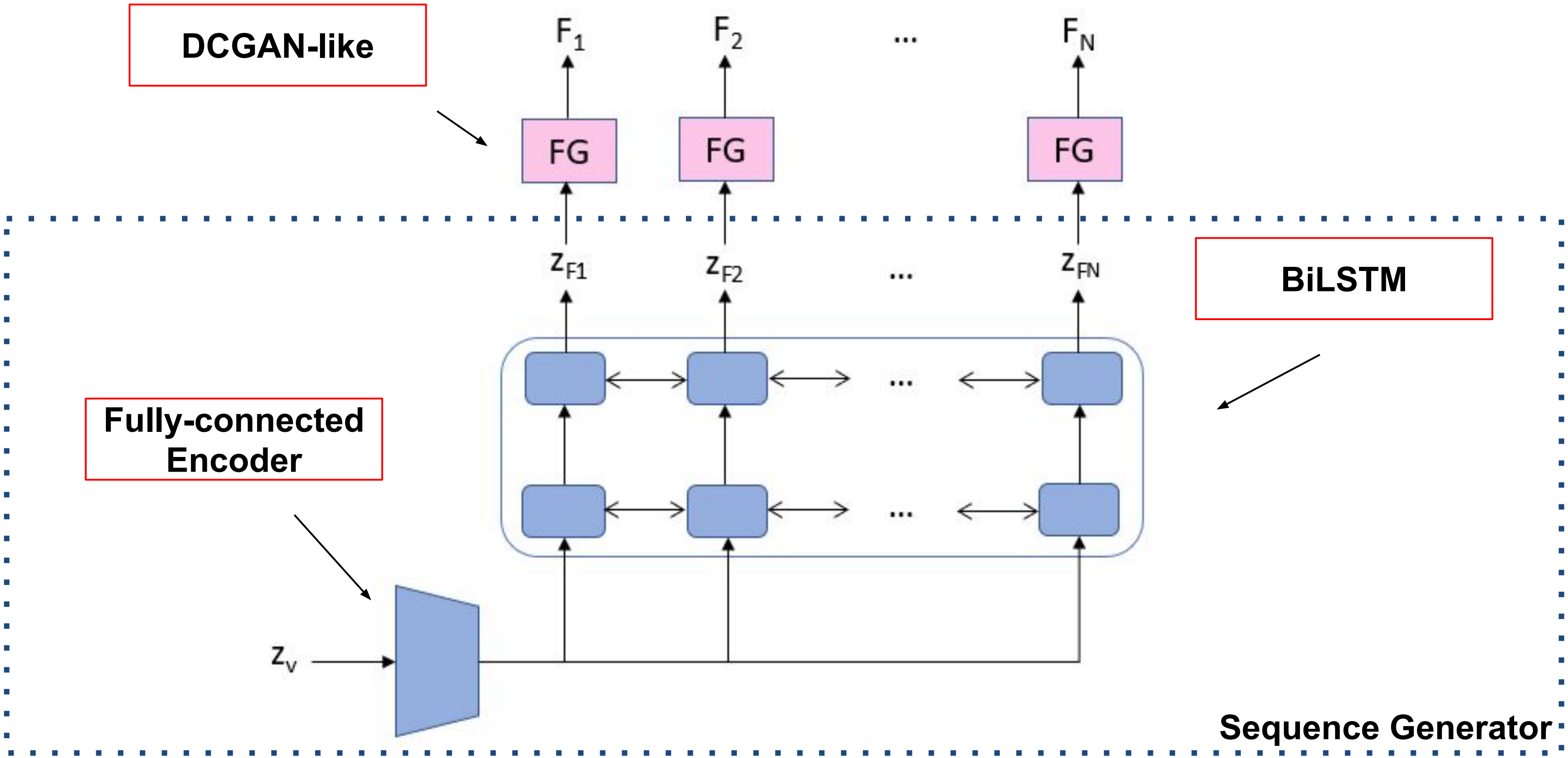
Two-step generation of temporal data



1. A frames generator (FG) is trained in advance to generate individual frames

2. A second model will be trained to navigate the manifold induced by FG

Two-step generation of temporal data



Training details

- Multiple discriminators settings with random projections are employed in both training steps - details on next slide
  - Easier to find a working set of hyperparameters
  - More diverse generated samples

- DCGAN-like discriminator was used for FG training, along with a variation with 3-dimensional convolutions for training the sequence model

- RMSprop in general yielded better results than Adam in both cases

Multiple discriminator training

- Neyshabur et al. (2017) introduced the use of multiple random projections
- Overlap between fake and real samples is larger in a randomly projected lower dimensional space
- The distribution induced by the generator approximates the real data distribution with a sufficiently large number of projections

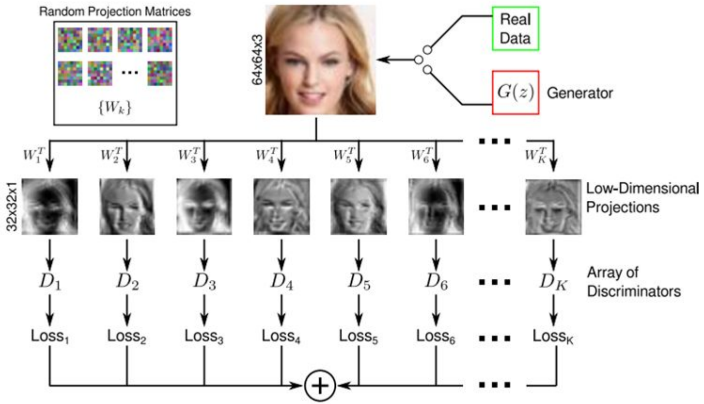


Figure from Neyshabur et al., 2017 - https://arxiv.org/abs/1705.07831

$$\min_G \max_{\{D_k\}} \sum_{k=1}^{K} V(\{D_k\}, G) = \sum_{k=1}^{K} \mathbb{E}_{x\sim p_x} \log D_k(W_k^T x) + \sum_{k=1}^{K} \mathbb{E}_{z\sim p_z} \log(1 - D_k(W_k^T G(z)))$$

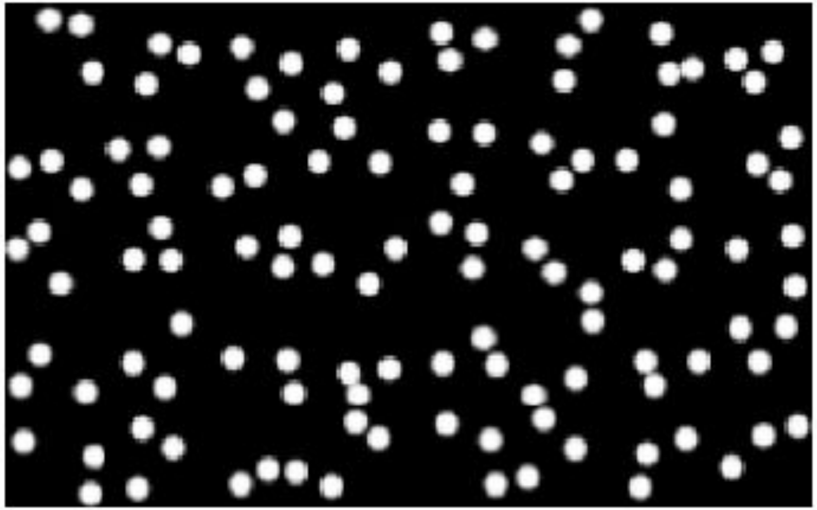
$$\mathcal{L}_G = -\sum_{k=1}^{K} \mathbb{E}_{z\sim p_z} \log D_k(G(z)) \qquad \mathcal{L}_{D_k} = -\mathbb{E}_{x\sim p_{\text{data}}} \log D_k(x) - \mathbb{E}_{z\sim p_z} \log(1 - D_k(G(z)))$$

Experiments

- Bouncing balls dataset with 3 balls
- 50000 x 40 training scenes
- Frames generator is trained against 48 discriminators for 50 epochs
  - Random frames are selected on the fly
- The sequence generator is then trained against 16 discriminators

Experiments - Frames generator samples

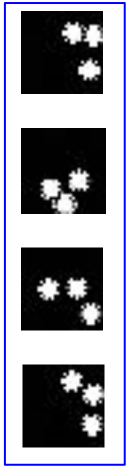
- Bouncing balls dataset with 3 balls
- 50000 x 40 training samples
- Trained against 48 discriminators for 50 epochs



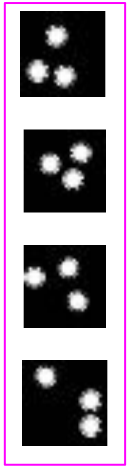
Experiments - Sequence generator samples

- Generated samples with 30 frames
- 16 discriminators with spatial random projections
- 3D convolutions DCGAN-like discriminator

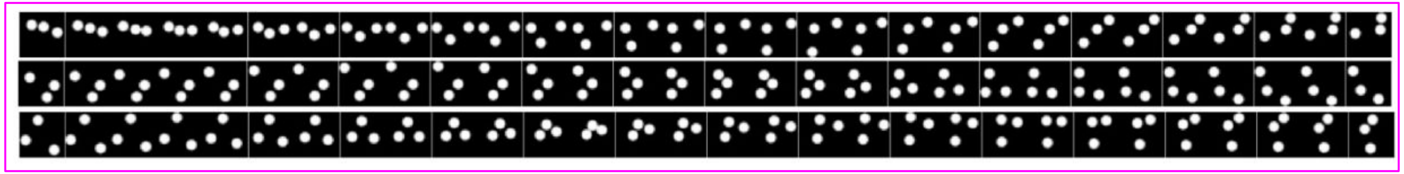
Generated



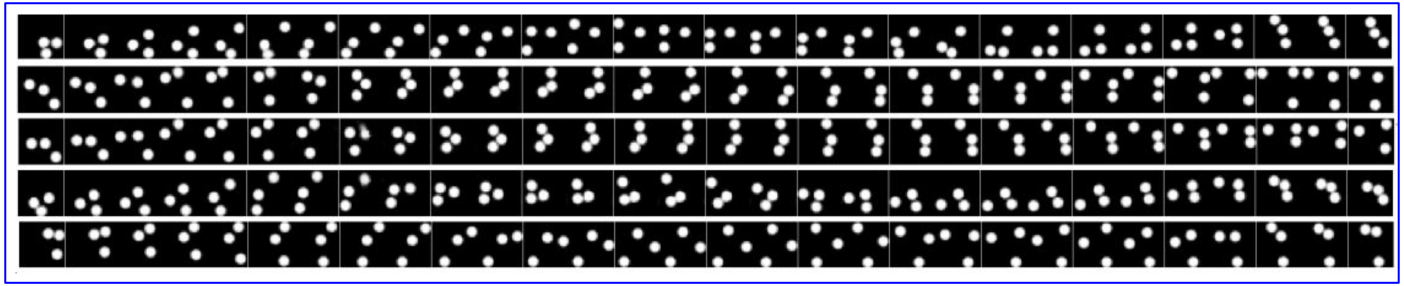
Real



Real

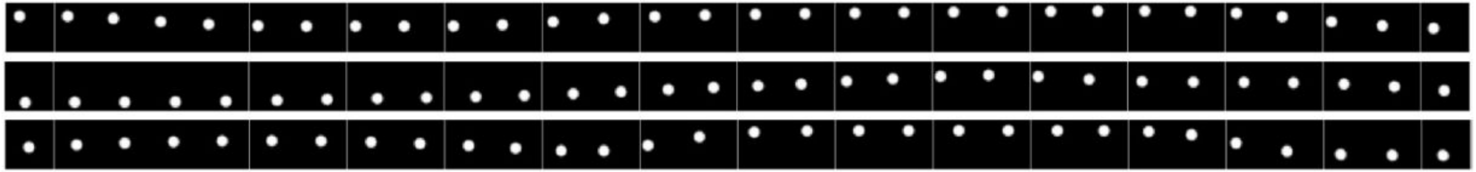


Generated

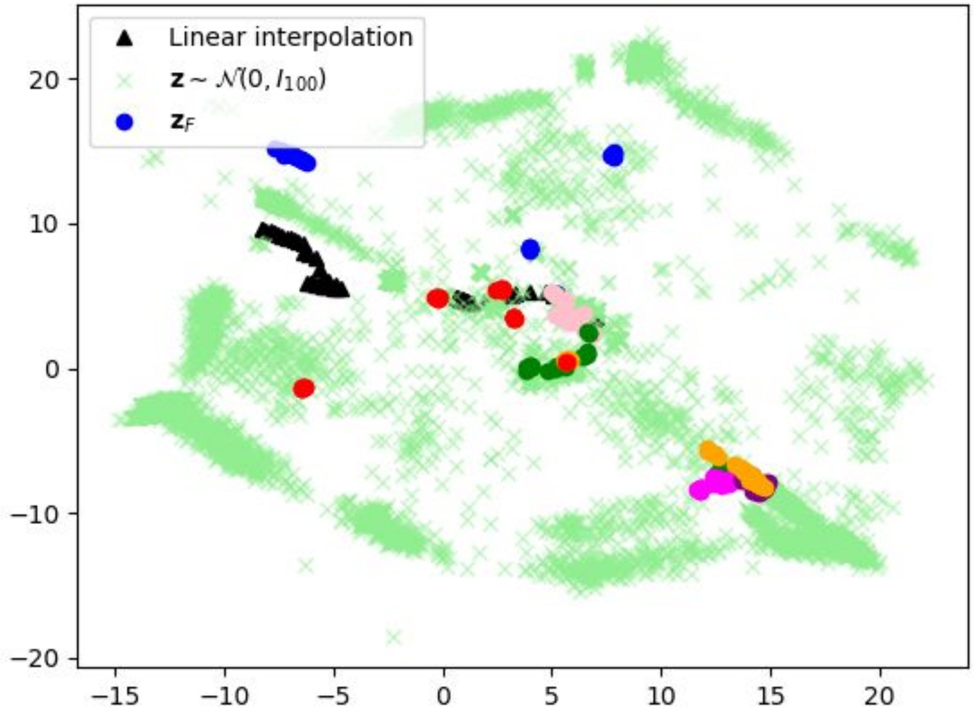


Replacing the frames generator

- We replaced the frames generator by one trained with 1 ball
- Some of the physics still holds, and frame transitions are smooth

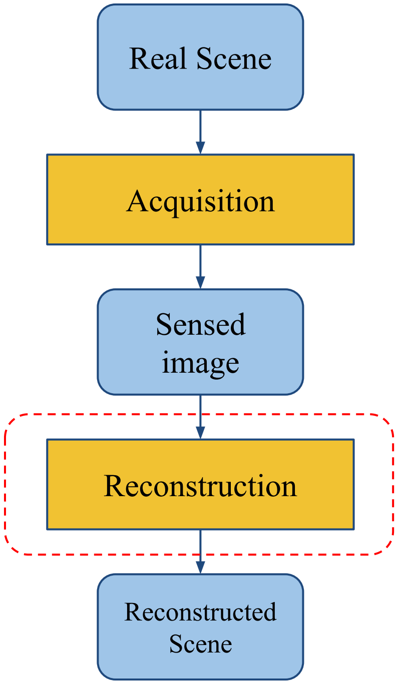


What is the video generator learning?



- 2D Isomap of generated sequences of latent variables
- The video generator learns to “jump” across the latent space rather than simply linearly interpolating

Learning to reconstruct* - simulating samples from fast imaging



- Fast imaging systems record video at ultra-high frame rate - see *Gao, Liang, et al. "Single-shot compressed ultrafast photography at one hundred billion frames per second." Nature 516.7529 (2014): 74.*

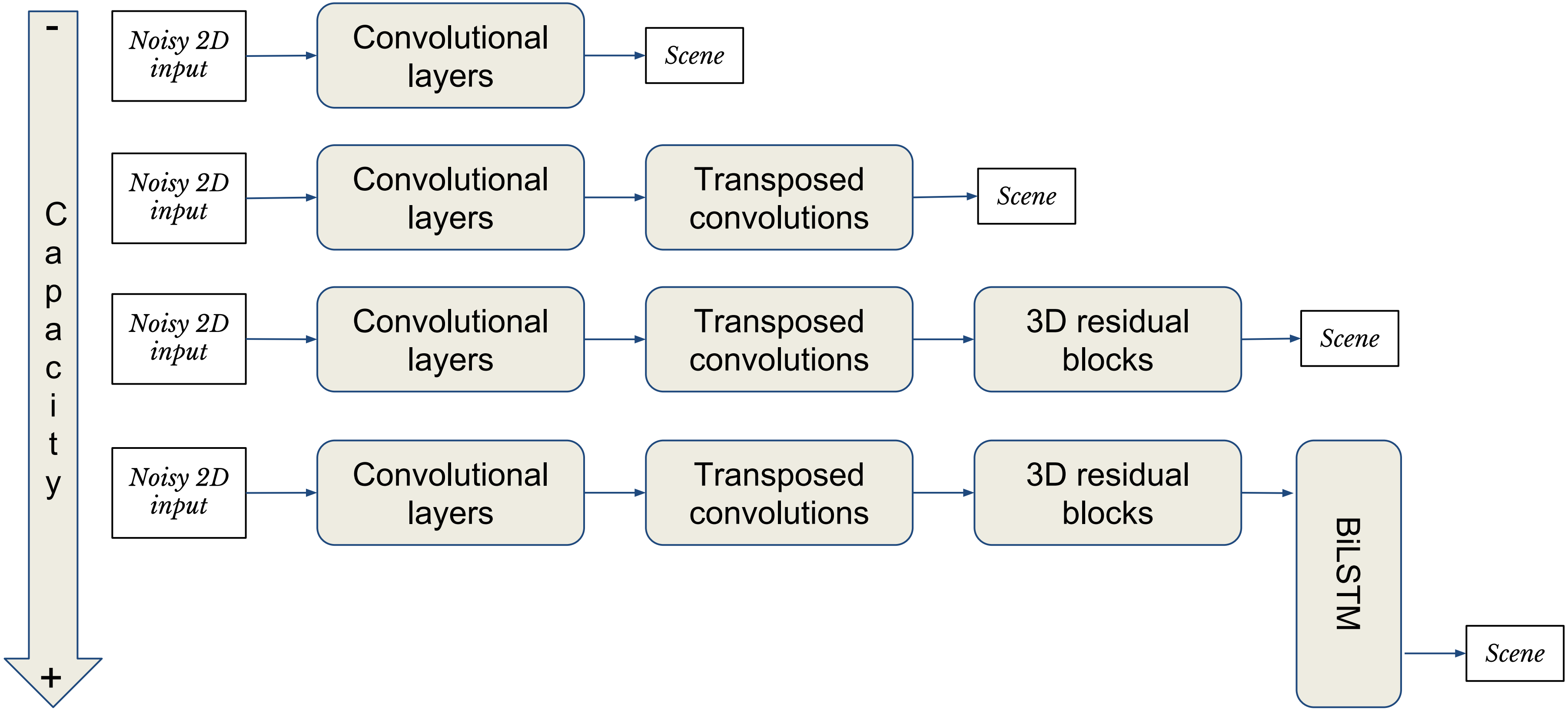
- Sensed images are noisy and sparse low dimensional versions of actual scenes

- Reconstruction is computationally expensive

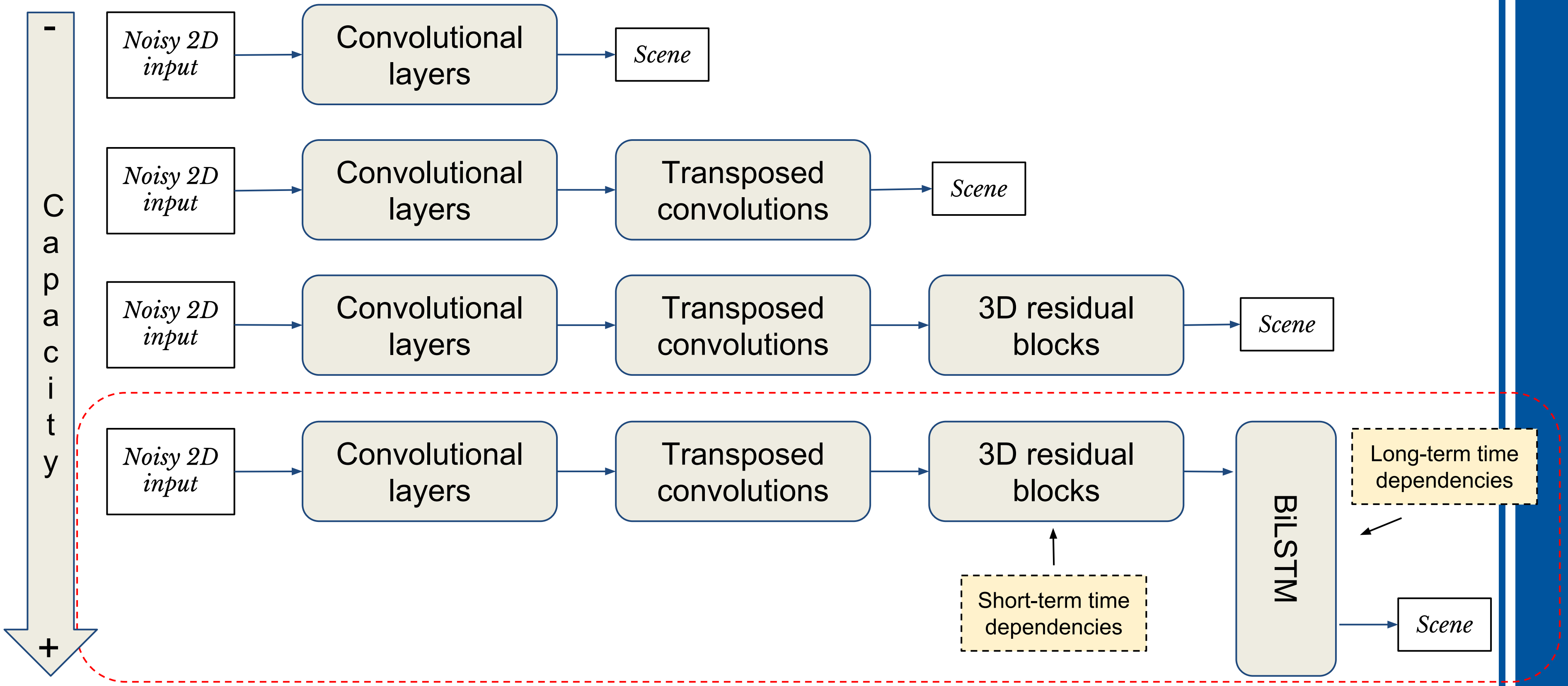
- Can the reconstruction phase be learned by a Neural Network?
  - Expensive offline training. Fast at test time
  - Can be done in batch mode, with GPU support.

***Collaboration with Prof. Jinyang Liang - INRS-EMT**

First trial - Direct reconstruction



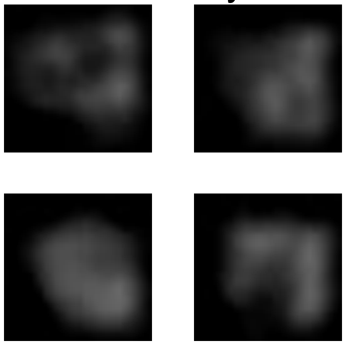
First trial - Direct reconstruction



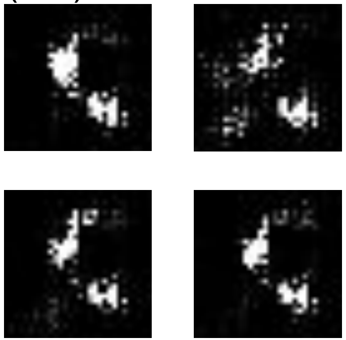
Reconstructed samples

- Training scheme:
  - Offline transformation of real scenes to look like sensed images
  - Neural net is trained with transformed/real pairs (400k pairs)
- Experiments with synthetic data:
  - Conventional MSE minimization leads to blurry samples
  - Adversarial loss adds artifacts (Fully convolutional DCGAN-style discriminator) and has low diversity

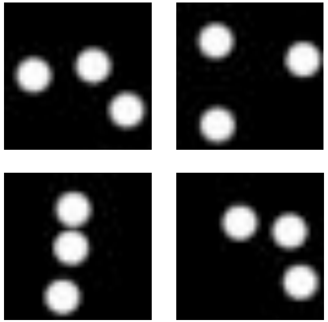
MSE only



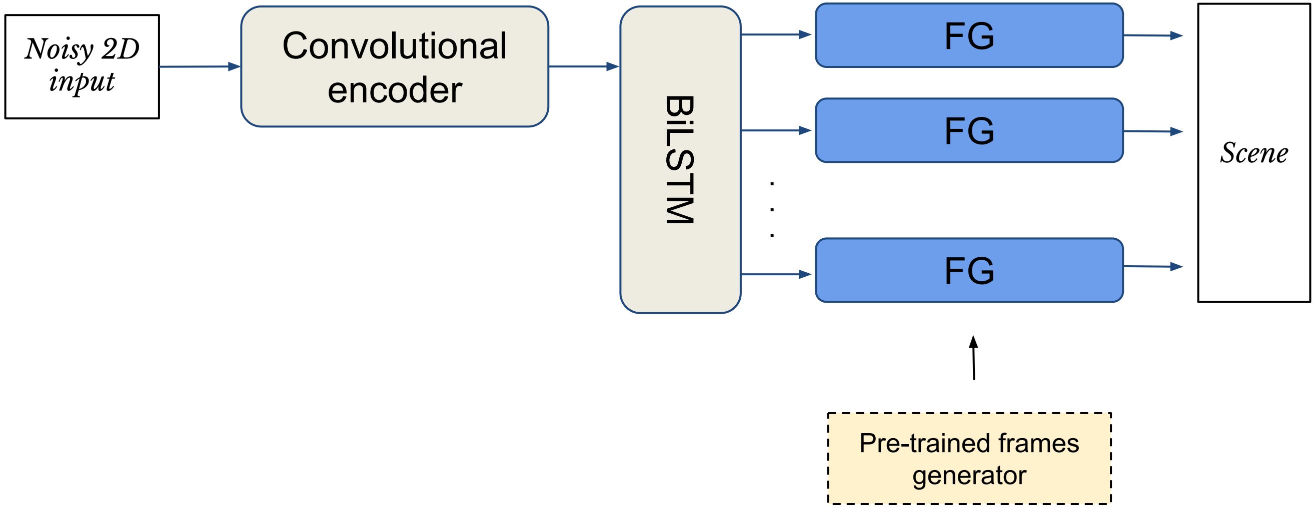
(1-a)*MSE + a*ADV



Real samples

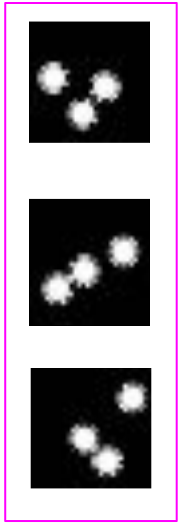


Two-step framework



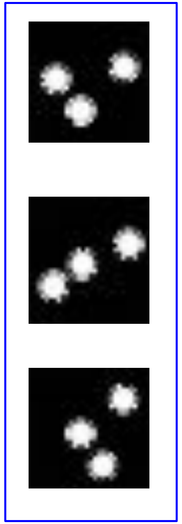
Reconstructed samples

Original scenes



Sensed Images

Reconstruction



Future work

- Other training strategies:
  - Let the frame generator continue training while the video generator is trained
  - Try different regularization strategies to enforce smooth frame transitions

- Scale to realistic data

- Evaluate objective quality metrics

Thank you!