Comparing different state-of-the-art solutions for image prediction using time-series analysis

Sören Dittrich

University of Hildesheim

Summerterm 2020

Table of contents

Contents

Contents

Image Prediction

- Field inside machine learning / computer vision

Image Prediction

- Field inside machine learning / computer vision
- Predict future image/s, given sequence of images

Image Prediction

- Field inside machine learning / computer vision
- Predict future image/s, given sequence of images
- X the image sequence of length n

Image Prediction

- Field inside machine learning / computer vision
- Predict future image/s, given sequence of images
- X the image sequence of length n
- with $X = (x_0, \ldots, x_{n-1})$

Image Prediction

- Field inside machine learning / computer vision
- Predict future image/s, given sequence of images
- X the image sequence of length n
- with $X = (x_0, \ldots, x_{n-1})$
- Two possible use-cases

Image Prediction

- Field inside machine learning / computer vision
- Predict future image/s, given sequence of images
- X the image sequence of length n
- with $X = (x_0, \ldots, x_{n-1})$
- Two possible use-cases
  - One-frame prediction

Image Prediction

- Field inside machine learning / computer vision
- Predict future image/s, given sequence of images
- X the image sequence of length n
- with $X = (x_0, \ldots, x_{n-1})$
- Two possible use-cases
  - One-frame prediction
    - Predicting x_n

Image Prediction

- Field inside machine learning / computer vision
- Predict future image/s, given sequence of images
- X the image sequence of length n
- with $X = (x_0, \ldots, x_{n-1})$
- Two possible use-cases
  - One-frame prediction
    - Predicting x_n
  - Multi-frame prediction

Image Prediction

- Field inside machine learning / computer vision
- Predict future image/s, given sequence of images
- X the image sequence of length n
- with $X = (x_0, \ldots, x_{n-1})$
- Two possible use-cases
  - One-frame prediction
    - Predicting x_n
  - Multi-frame prediction
    - Predict $t > 1$ frames into the future $(x_n, \ldots, x_{n+t-1})$

Image Prediction

- Field inside machine learning / computer vision
- Predict future image/s, given sequence of images
- X the image sequence of length n
- with $X = (x_0, \ldots, x_{n-1})$
- Two possible use-cases
  - One-frame prediction
    - Predicting x_n
  - Multi-frame prediction
    - Predict $t > 1$ frames into the future $(x_n, \ldots, x_{n+t-1})$
    - Often it is one-frame prediction in a feedback loop

Image Prediction

- Field inside machine learning / computer vision
- Predict future image/s, given sequence of images
- X the image sequence of length n
- with $X = (x_0, \ldots, x_{n-1})$
- Two possible use-cases
  - One-frame prediction
    - Predicting x_n
  - Multi-frame prediction
    - Predict $t > 1$ frames into the future $(x_n, \ldots, x_{n+t-1})$
    - Often it is one-frame prediction in a feedback loop
    - Propagate the error $\rightarrow$ Greater error in later images

Autoencoder

- Two networks chained together

[1] h is the so named **code**. Output layer is named **bottleneck layer**.

Autoencoder

- Two networks chained together
  - Encoder

[1] h is the so named **code**. Output layer is named **bottleneck layer**.

Autoencoder

- Two networks chained together
  - Encoder
    - Input is x

[1] h is the so named **code**. Output layer is named **bottleneck layer**.

Autoencoder

- Two networks chained together
  - Encoder
    - Input is x
    - Output is h[1]

[1] h is the so named **code**. Output layer is named **bottleneck layer**.

Autoencoder

- Two networks chained together
  - Encoder
    - Input is x
    - Output is h^1
    - $E(x) = h$

[1] h is the so named **code**. Output layer is named **bottleneck layer**.

Autoencoder

- Two networks chained together
  - Encoder
    - Input is x
    - Output is h^1
    - $E(x) = h$
  - Decoder

[1] h is the so named **code**. Output layer is named **bottleneck layer**.

Autoencoder

- Two networks chained together
  - Encoder
    - Input is x
    - Output is h[1]
    - $E(x) = h$
  - Decoder
    - Input is h

[1] h is the so named **code**. Output layer is named **bottleneck layer**.

Autoencoder

- Two networks chained together
  - Encoder
    - Input is x
    - Output is h[1]
    - $E(x) = h$
  - Decoder
    - Input is h
    - Output is $x\prime$

[1] h is the so named **code**. Output layer is named **bottleneck layer**.

Autoencoder

- Two networks chained together
  - Encoder
    - Input is x
    - Output is h[1]
    - $E(x) = h$
  - Decoder
    - Input is h
    - Output is $x\prime$
    - $D(h) = x\prime$

[1] h is the so named **code**. Output layer is named **bottleneck layer**.

Autoencoder

- Two networks chained together
  - Encoder
    - Input is x
    - Output is h[1]
    - $E(x) = h$
  - Decoder
    - Input is h
    - Output is $x\prime$
    - $D(h) = x\prime$
- Used for reconstruction $x \approx x\prime$

[1] h is the so named **code**. Output layer is named **bottleneck layer**.

Autoencoder

- Two networks chained together
  - Encoder
    - Input is x
    - Output is h[1]
    - $E(x) = h$
  - Decoder
    - Input is h
    - Output is $x\prime$
    - $D(h) = x\prime$
- Used for reconstruction $x \approx x\prime$
- Important is to prevent the network to simply copy x to $x\prime$ (Interpolation)

[1] h is the so named **code**. Output layer is named **bottleneck layer**.

Autoencoder

- Two networks chained together
  - Encoder
    - Input is x
    - Output is h[1]
    - $E(x) = h$
  - Decoder
    - Input is h
    - Output is $x\prime$
    - $D(h) = x\prime$
- Used for reconstruction $x \approx x\prime$
- Important is to prevent the network to simply copy x to $x\prime$ (Interpolation)
- Simplest architecture is the undercomplete autoencoder

[1] h is the so named **code**. Output layer is named **bottleneck layer**.

Autoencoder

- Two networks chained together
  - Encoder
    - Input is x
    - Output is h[1]
    - $E(x) = h$
  - Decoder
    - Input is h
    - Output is $x\prime$
    - $D(h) = x\prime$
- Used for reconstruction $x \approx x\prime$
- Important is to prevent the network to simply copy x to $x\prime$ (Interpolation)
- Simplest architecture is the undercomplete autoencoder
  - Code smaller then input

[1] h is the so named **code**. Output layer is named **bottleneck layer**.

Autoencoder

- Two networks chained together
  - Encoder
    - Input is x
    - Output is h[1]
    - $E(x) = h$
  - Decoder
    - Input is h
    - Output is $x\prime$
    - $D(h) = x\prime$
- Used for reconstruction $x \approx x\prime$
- Important is to prevent the network to simply copy x to $x\prime$ (Interpolation)
- Simplest architecture is the undercomplete autoencoder
  - Code smaller then input
  - Network needs to distinguish between useful and obsolete

[1] h is the so named **code**. Output layer is named **bottleneck layer**.

Autoencoder

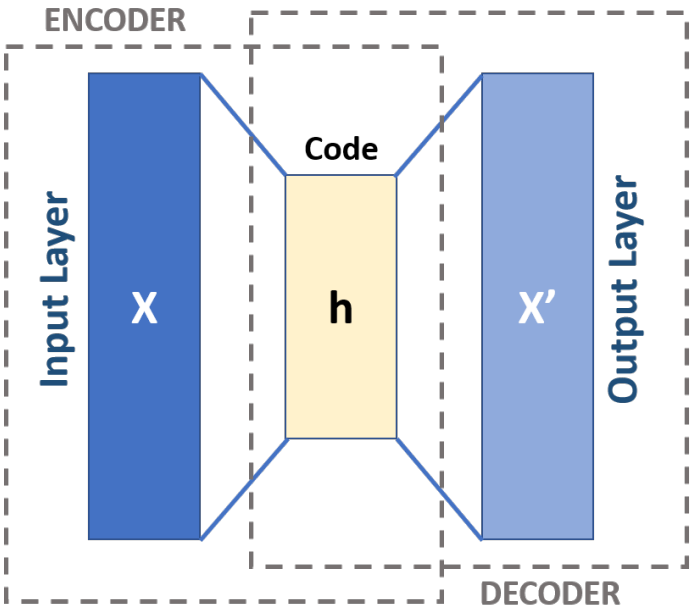

Figure: Autoencoder schema [**?**]

CNN

- Convolutional Neural Network

CNN

- Convolutional Neural Network
- Consists of three stages

CNN

- Convolutional Neural Network
- Consists of three stages
  1. Convolutional layer

CNN

- Convolutional Neural Network
- Consists of three stages
  1. Convolutional layer
  2. Non-linearity (ReLU, sigmoid, ...)

CNN

- Convolutional Neural Network
- Consists of three stages
  1. Convolutional layer
  2. Non-linearity (ReLU, sigmoid, ...)
  3. Pooling layer

CNN (First stage)

- Convolutional operation is discrete

CNN (First stage)

- Convolutional operation is discrete
- $(I * K)(i, j) = \sum_m \sum_n I(m, n)K(i - m, j - n)$

CNN (First stage)

- Convolutional operation is discrete
- $(I * K)(i,j) = \sum_m \sum_n I(m,n)K(i-m, j-n)$

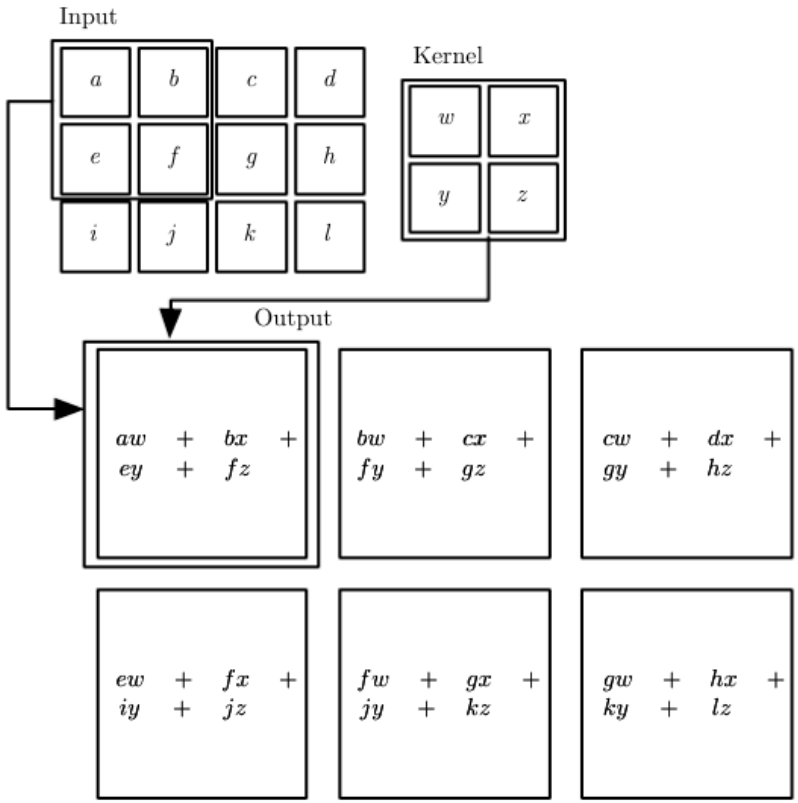

Figure: Two dimensional convolutional operation [**?**]

RNN

LSTM

ConvLSTM

Backpropagation

BPTT

Contents

LSTM Autoencoder

- „Unsupervised Learning of Video Representations using LSTMs“by Srivastava et. al. [**?**]

LSTM Autoencoder

- „Unsupervised Learning of Video Representations using LSTMs“by Srivastava et. al. [**?**]
- Using the standard LSTM from Hochreiter & Schmidhuber [**?**]

LSTM Autoencoder

- „Unsupervised Learning of Video Representations using LSTMs“by Srivastava et. al. [**?**]
- Using the standard LSTM from Hochreiter & Schmidhuber [**?**]
- Autoencoder architecture

LSTM Autoencoder

- „Unsupervised Learning of Video Representations using LSTMs“by Srivastava et. al. [**?**]
- Using the standard LSTM from Hochreiter & Schmidhuber [**?**]
- Autoencoder architecture
- Useful for future image prediction & image reconstruction

LSTM Autoencoder

- „Unsupervised Learning of Video Representations using LSTMs“by Srivastava et. al. [**?**]
- Using the standard LSTM from Hochreiter & Schmidhuber [**?**]
- Autoencoder architecture
- Useful for future image prediction & image reconstruction
- Typical baseline for newer, more advanced algorithms

LSTM Autoencoder

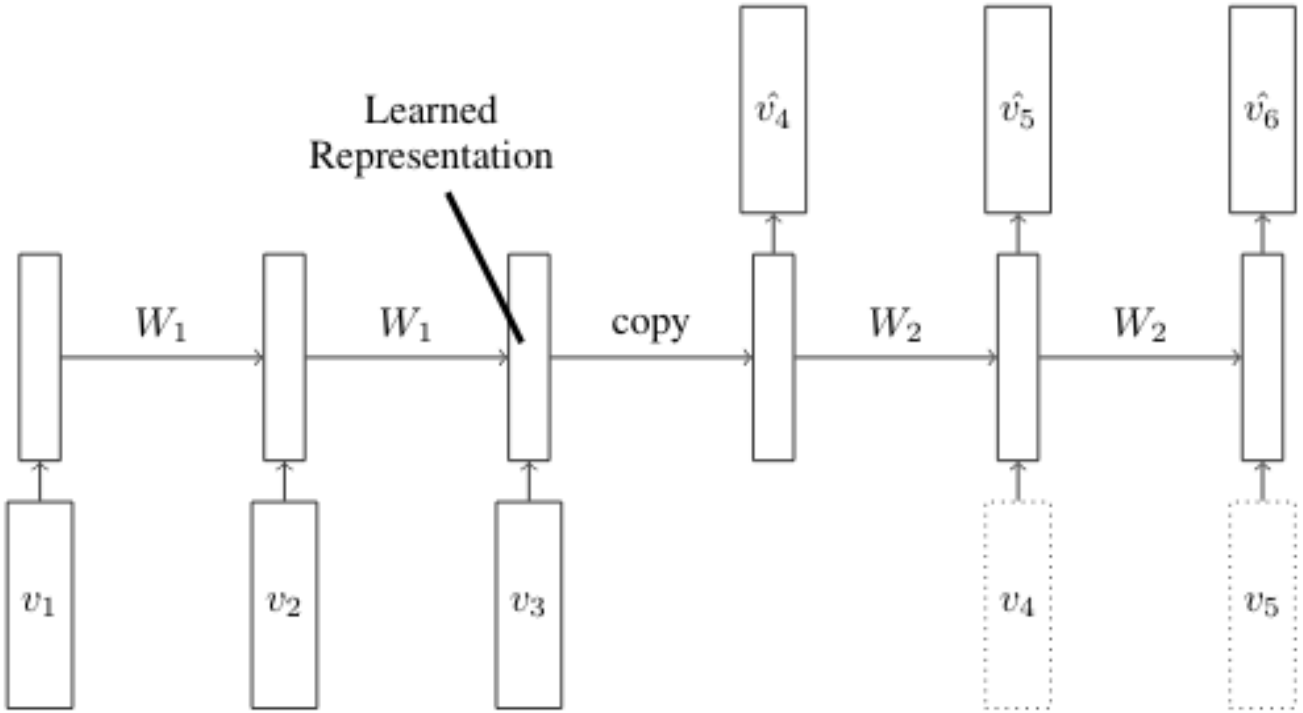

Figure: Future image prediction model [**?**]

LSTM Autoencoder

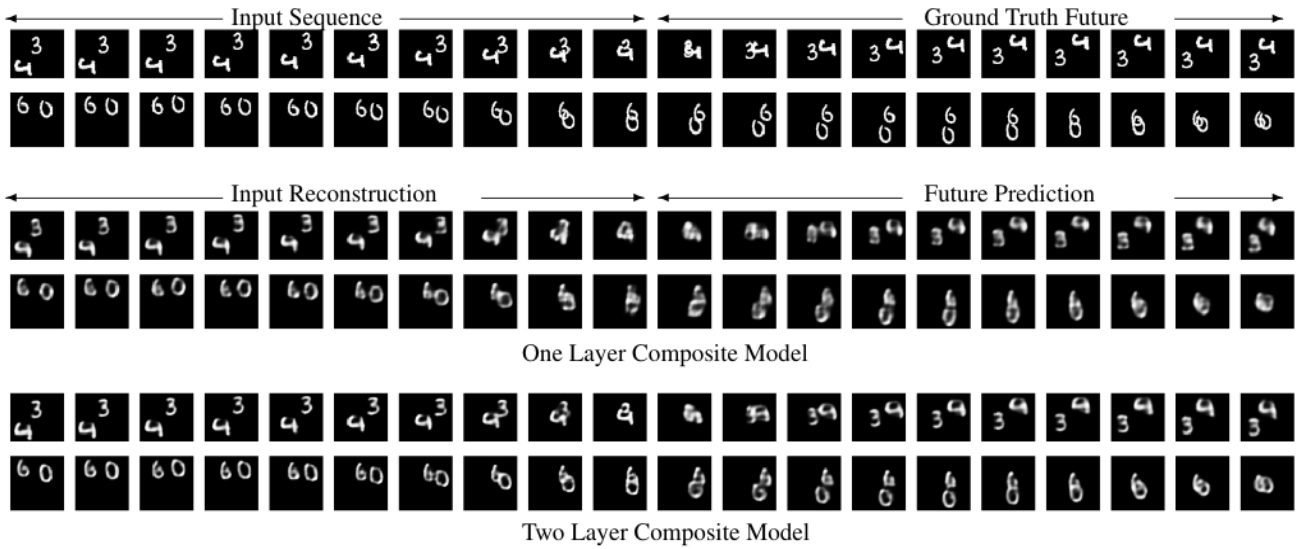

Figure: Results of MovingMNIST experiment [**?**]

ConvLSTM Autoencoder

- „Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting“by Shi et. al. [**?**]

ConvLSTM Autoencoder

- „Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting“by Shi et. al. [**?**]
- Similar to LSTM Autoencoder, but uses ConvLSTM instead

ConvLSTM Autoencoder

- „Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting“by Shi et. al. [**?**]
- Similar to LSTM Autoencoder, but uses ConvLSTM instead
- Outperforms the LSTM Autoencoder

ConvLSTM Autoencoder

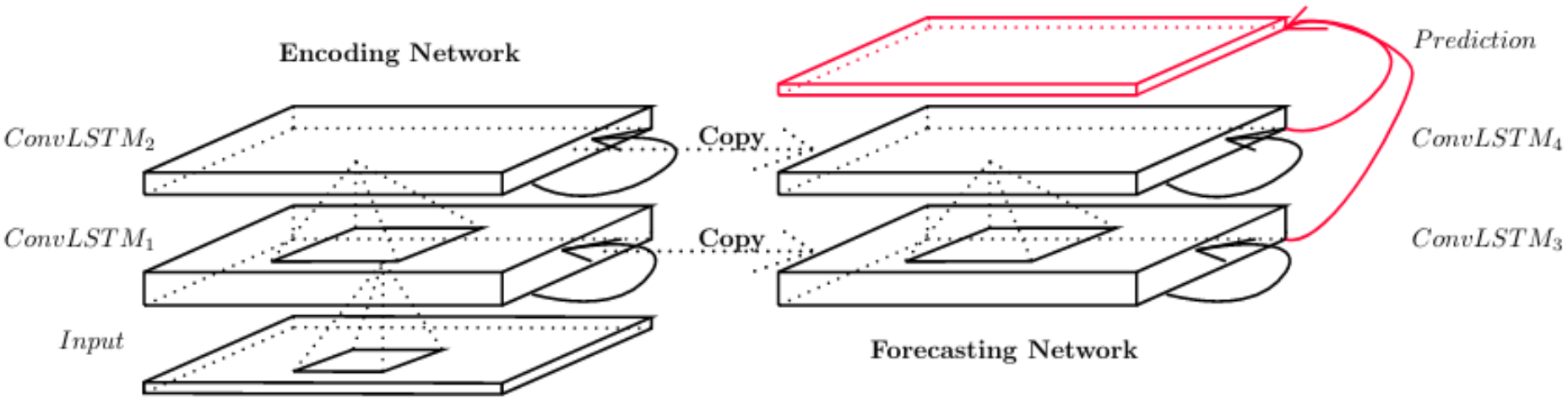

Figure: Future image prediction model [**?**]

ConvLSTM Autoencoder

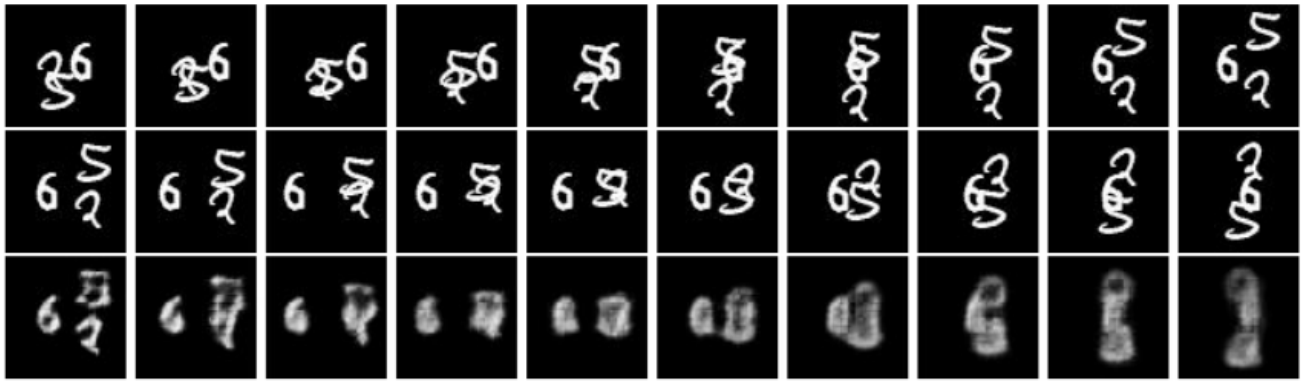

Figure: Results of MovingMNIST experiment [**?**]

Spatio-temporal Video Autoencoder

- „Spatio-Temporal Video Autoencoder With Differentiable Memory “by Patraucean et. al. [**?**]

Spatio-temporal Video Autoencoder

- „Spatio-Temporal Video Autoencoder With Differentiable Memory “by Patraucean et. al. [**?**]
-

Spatio-temporal Video Autoencoder

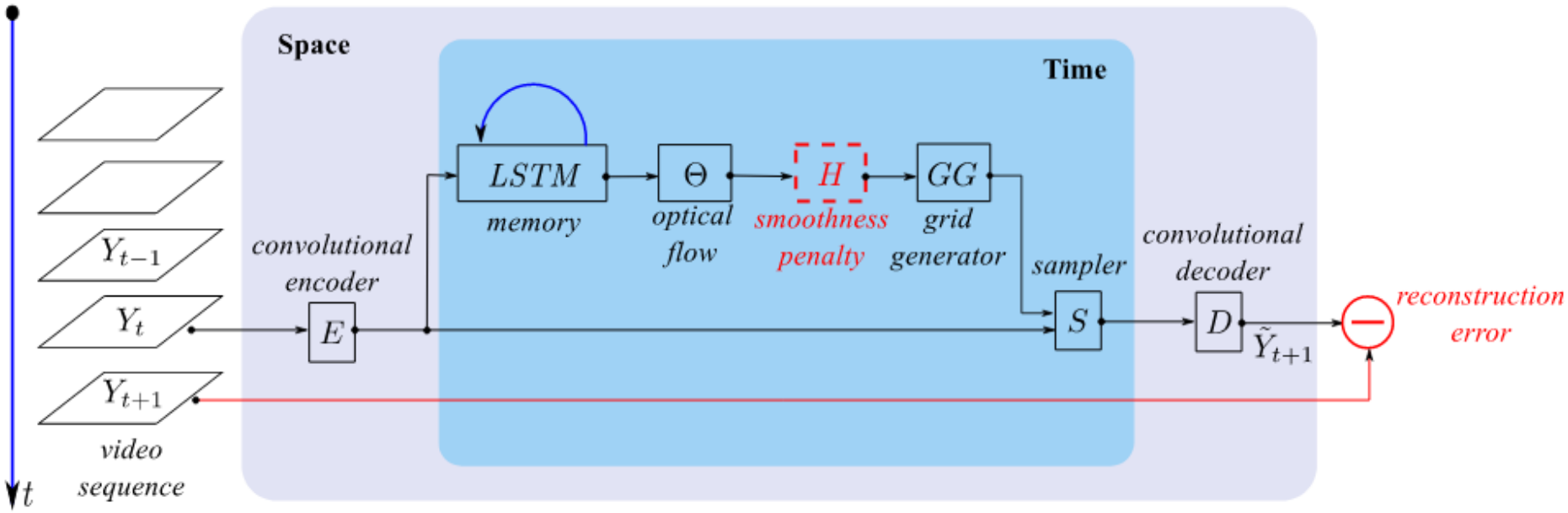

Figure: Spatio-temporal Video Autoencoder Architecture [**?**]

Spatio-temporal Video Autoencoder

Figure: Results of MovingMNIST experiment [**?**]

PredNet

PredRNN