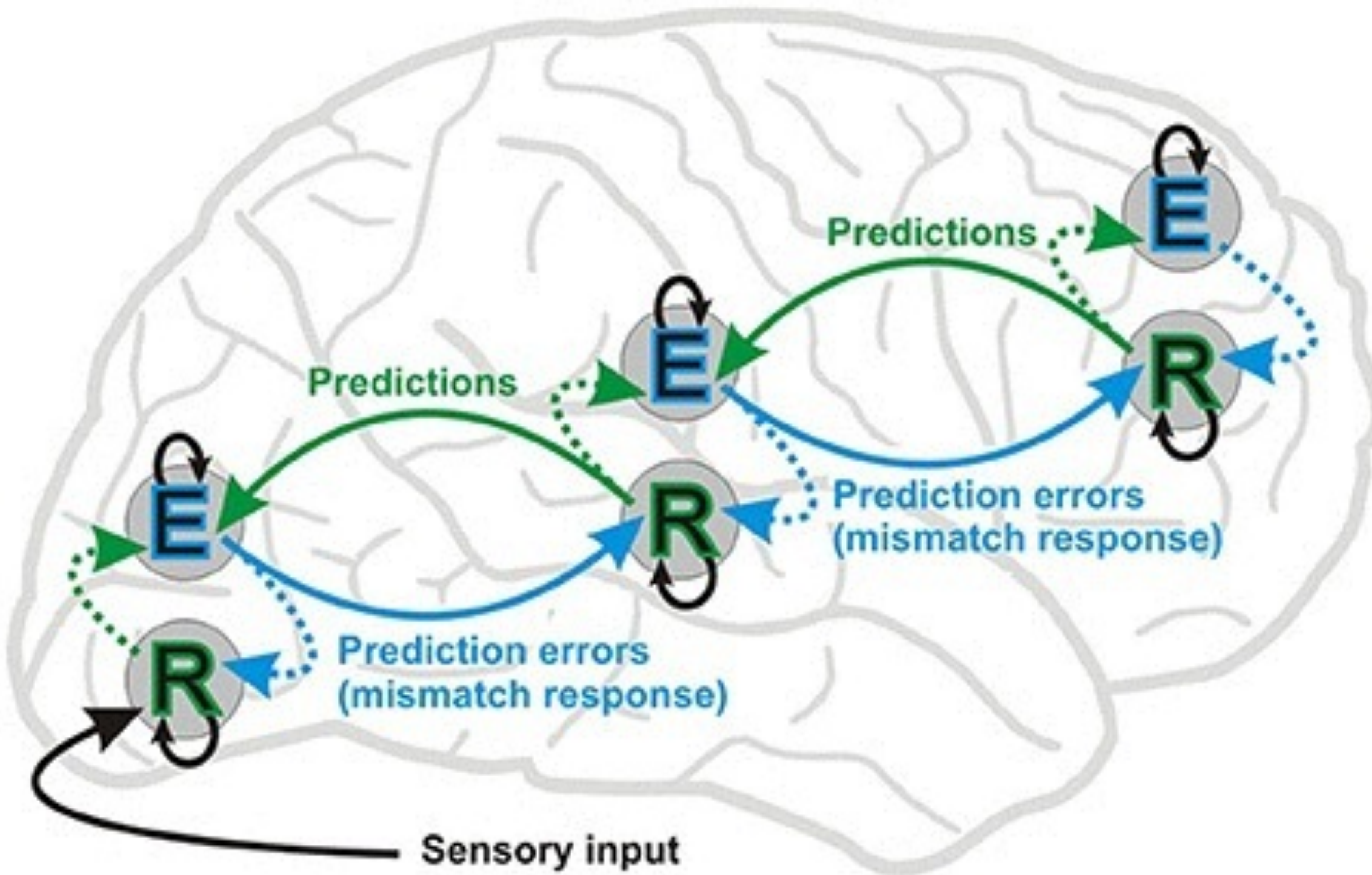
Unsupervised Learning of Visual Structure Using Predictive Generative Networks

William Lotter, Gabriel Kreiman & David Cox

Harvard University, Cambridge, USA

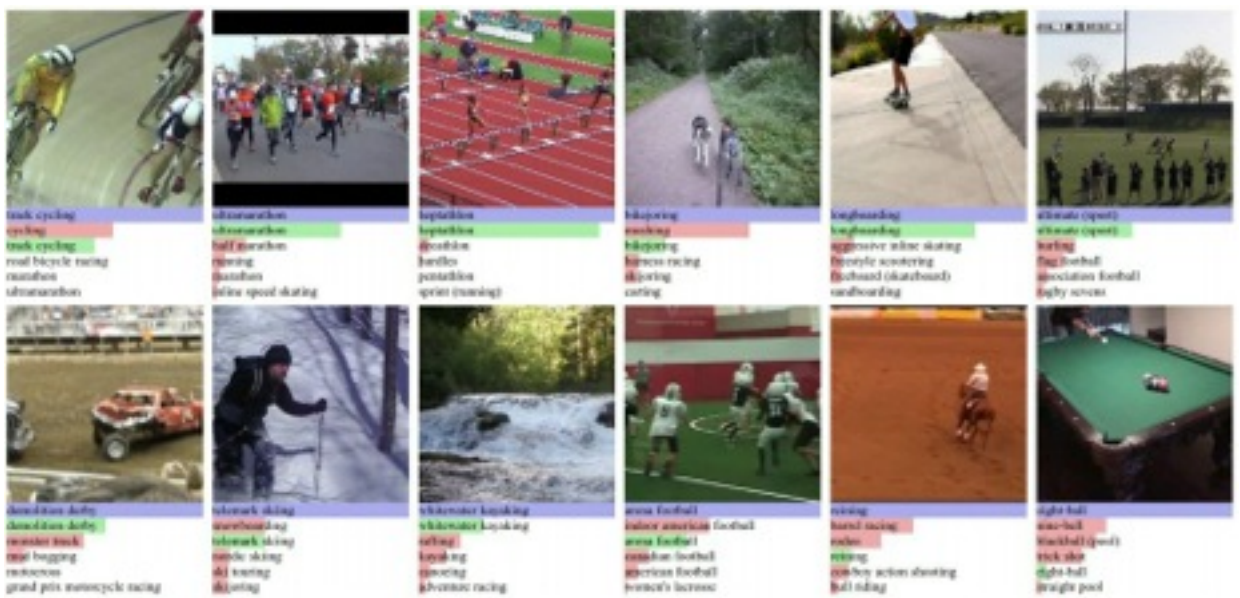
Article overview by Ilya Kuzovkin

Computational Neuroscience Seminar
University of Tartu
2015

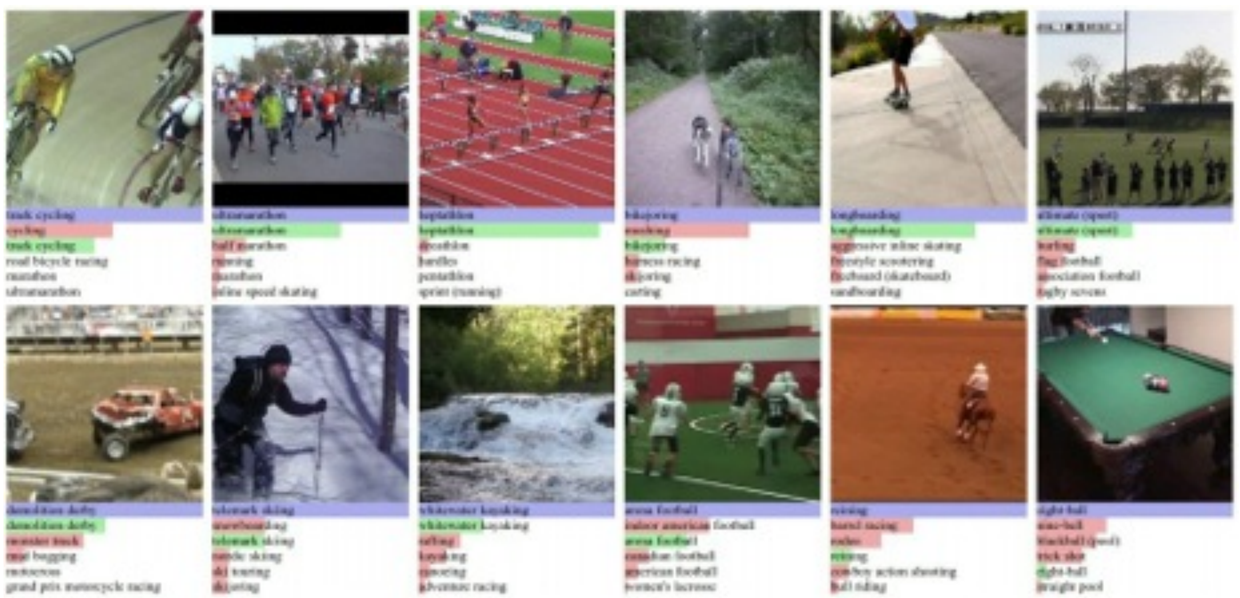


The idea of *predictive coding* in neuroscience

“state-of-the-art deep learning models rely on millions of labeled training examples to learn”

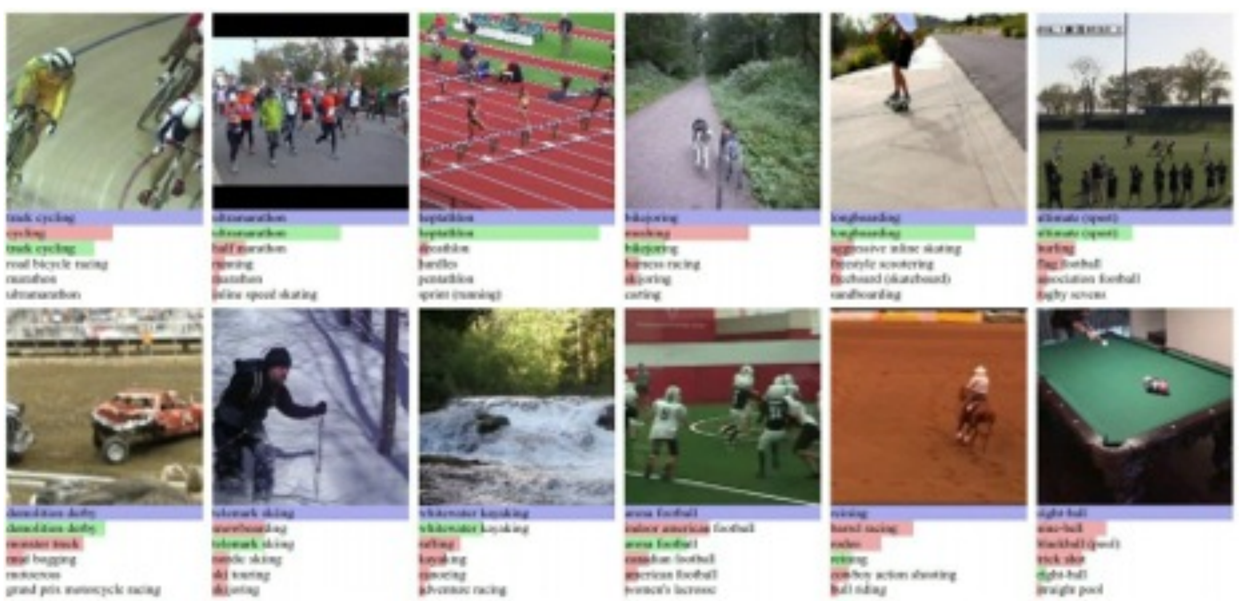


“state-of-the-art deep learning models rely on millions of labeled training examples to learn”



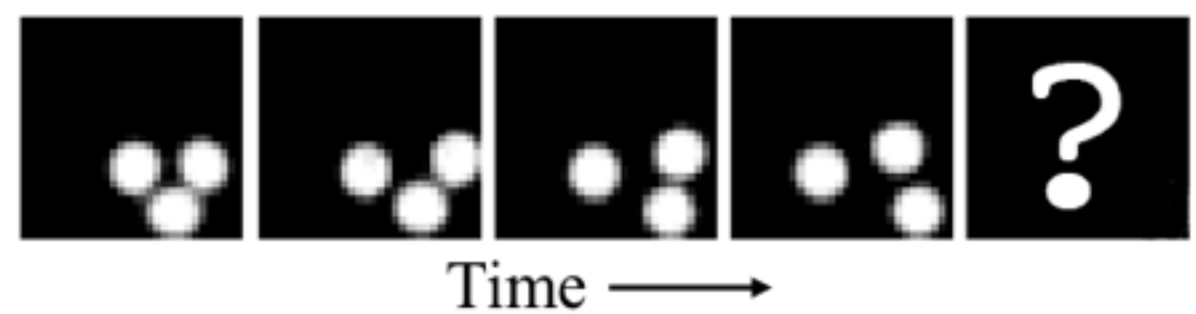
“in contrast to biological systems, where learning is largely unsupervised”

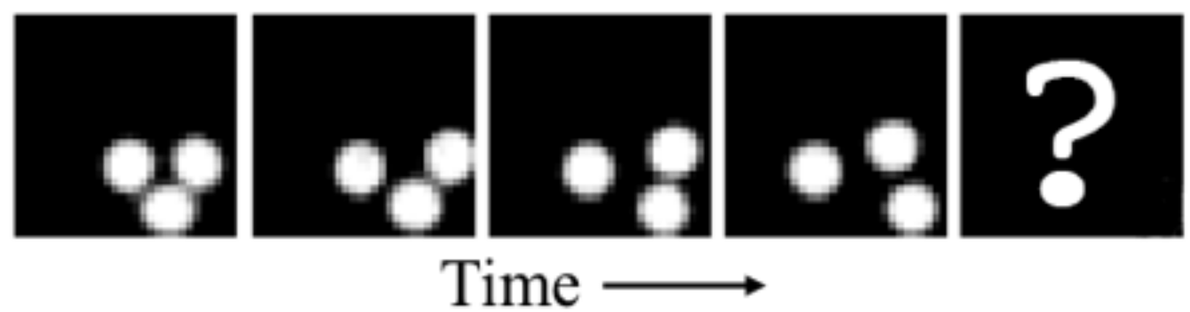
“state-of-the-art deep learning models rely on millions of labeled training examples to learn”



“in contrast to biological systems, where learning is largely unsupervised”

“we explore the idea that prediction is not only a useful end-goal, but may also serve as a powerful unsupervised learning signal”

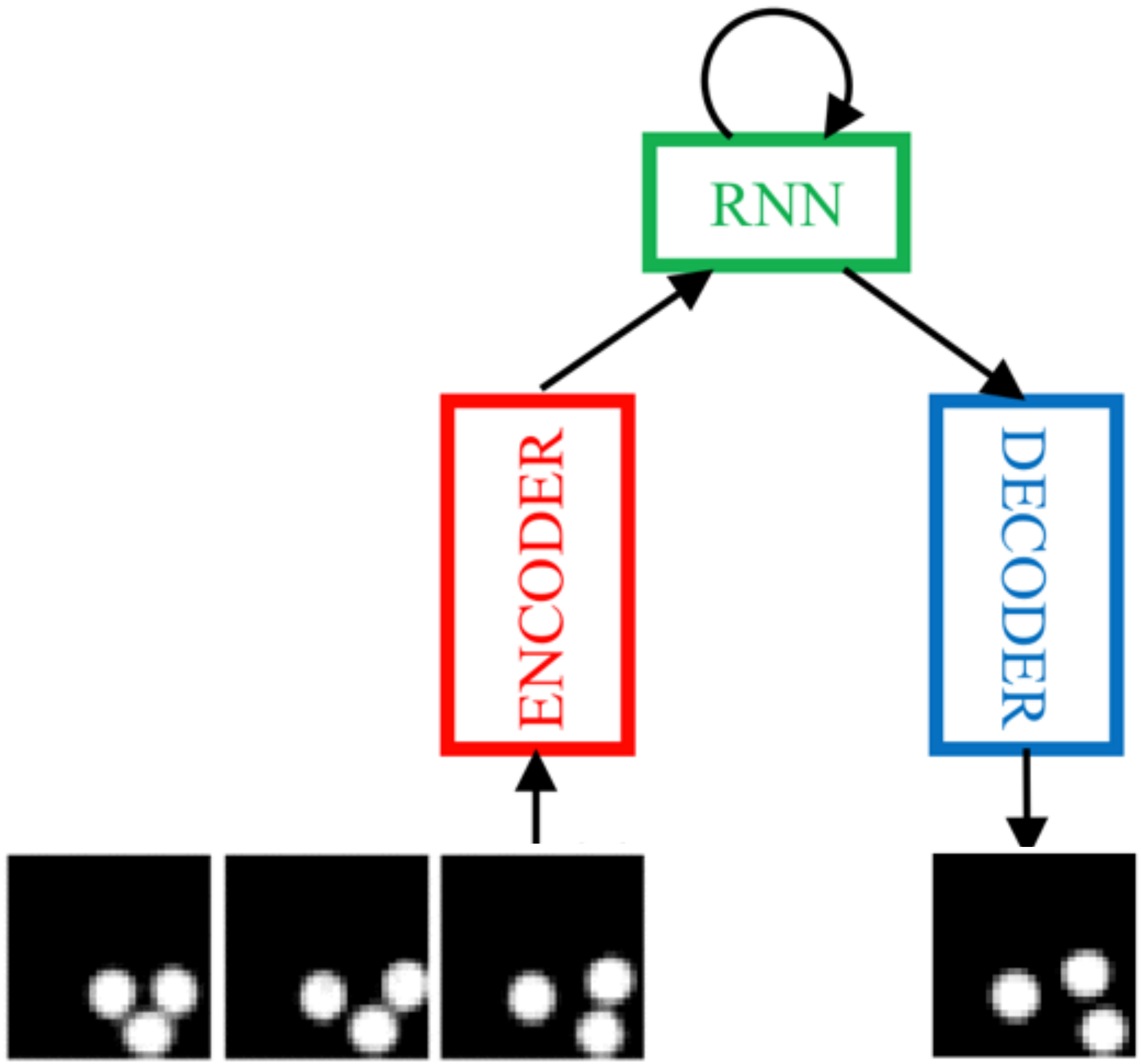




PART I
THE IDEA OF PREDICTIVE ENCODER

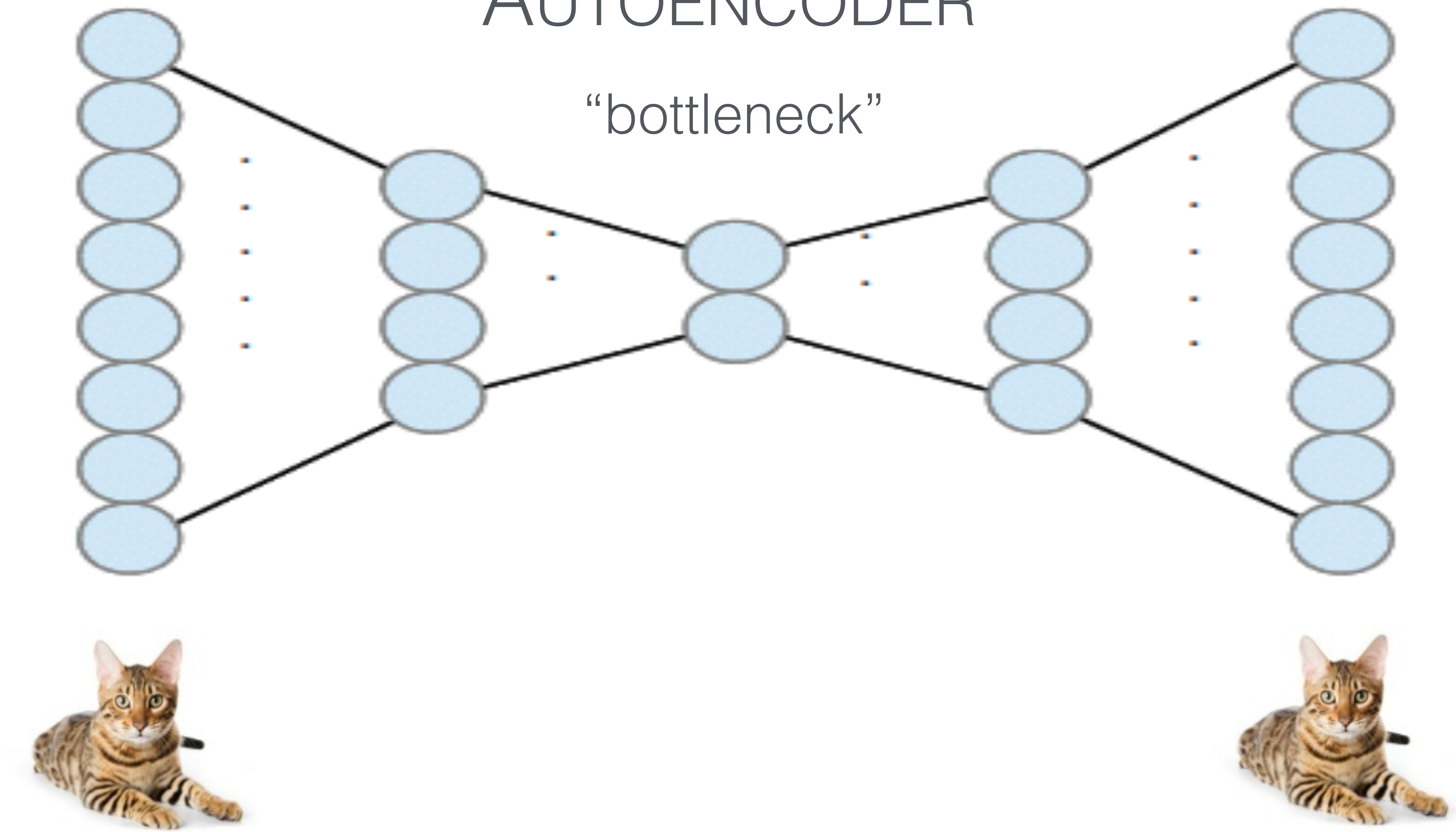
"prediction may also serve as a powerful unsupervised learning signal"

Predictive Generative Network
(a.k.a "Predictive Encoder" Palm 2012)

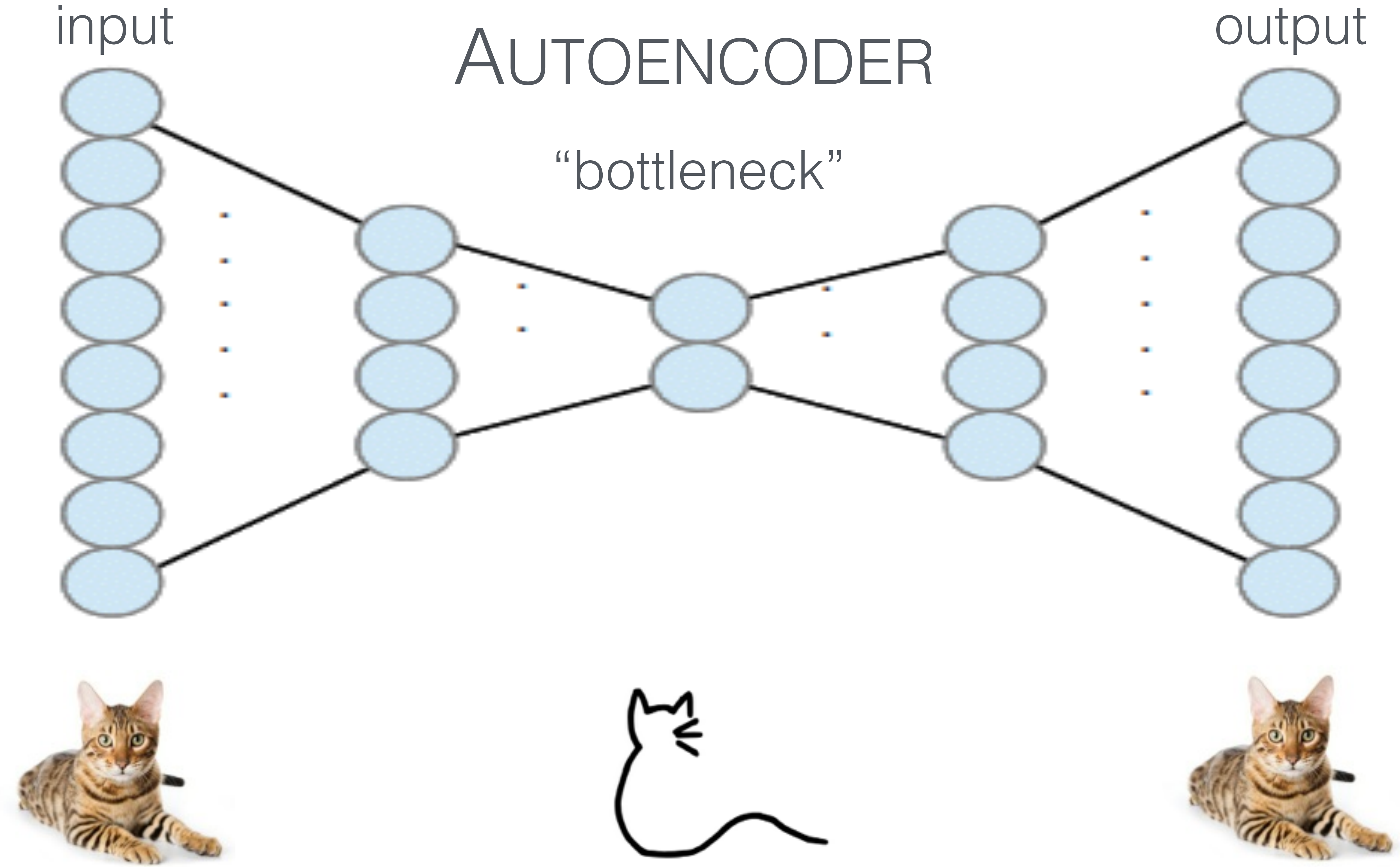


$x^i_{1:t}$ $G(x^i_{1:t})$ vs. x^i_{t+1}

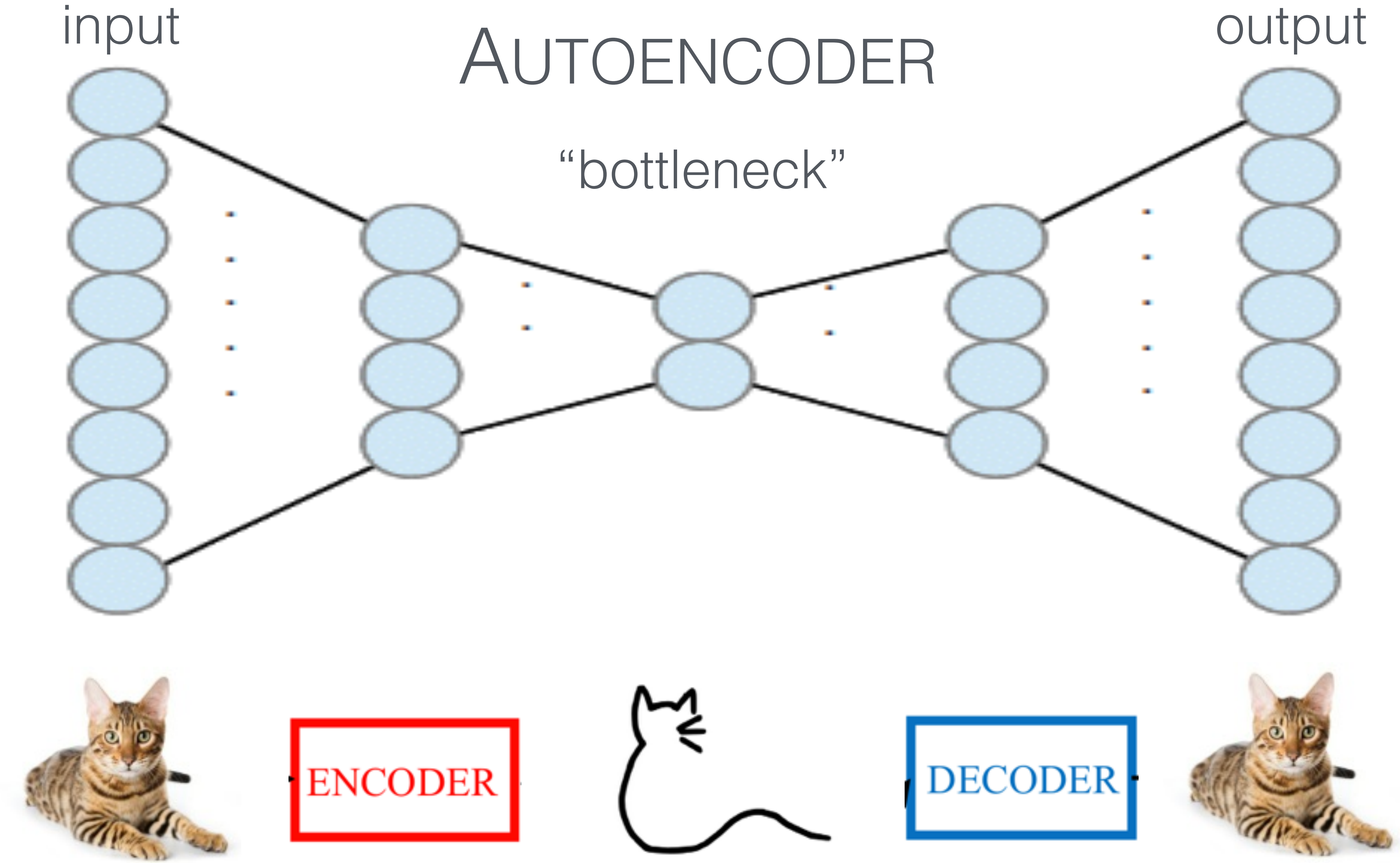
AUTOENCODER



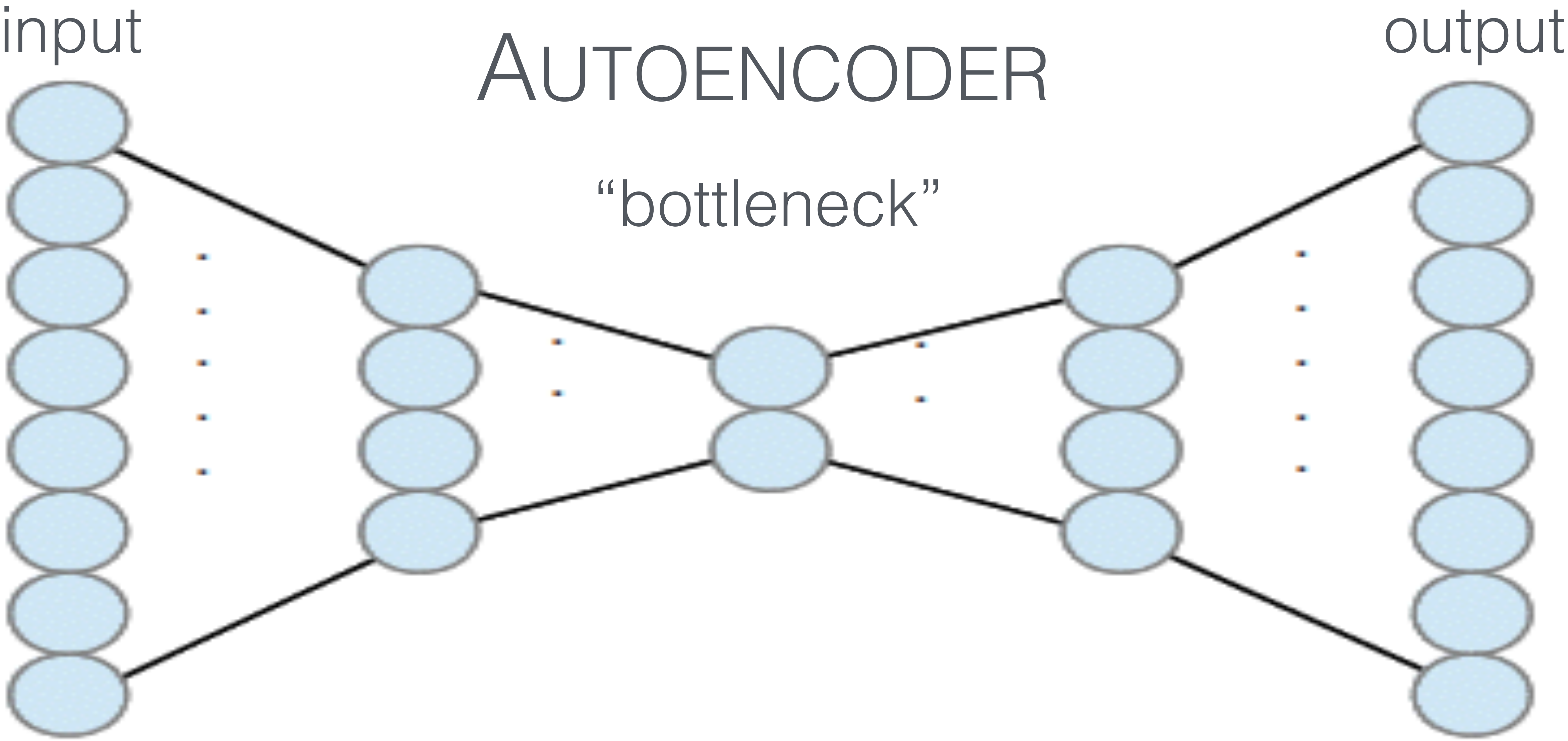
AUTOENCODER

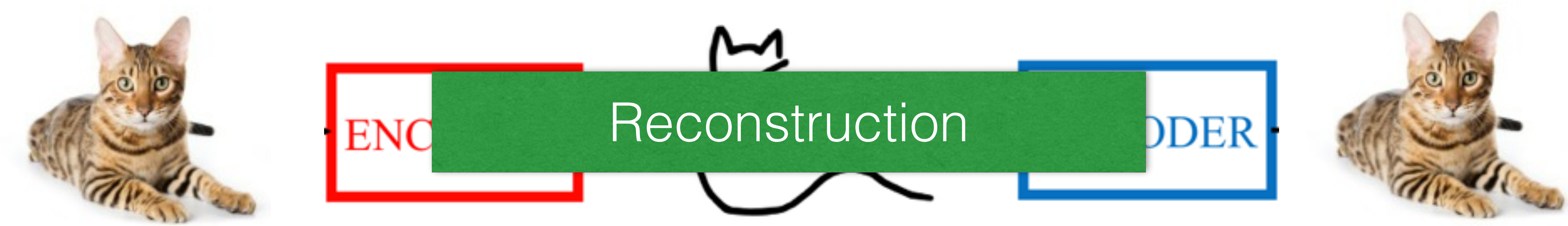


AUTOENCODER

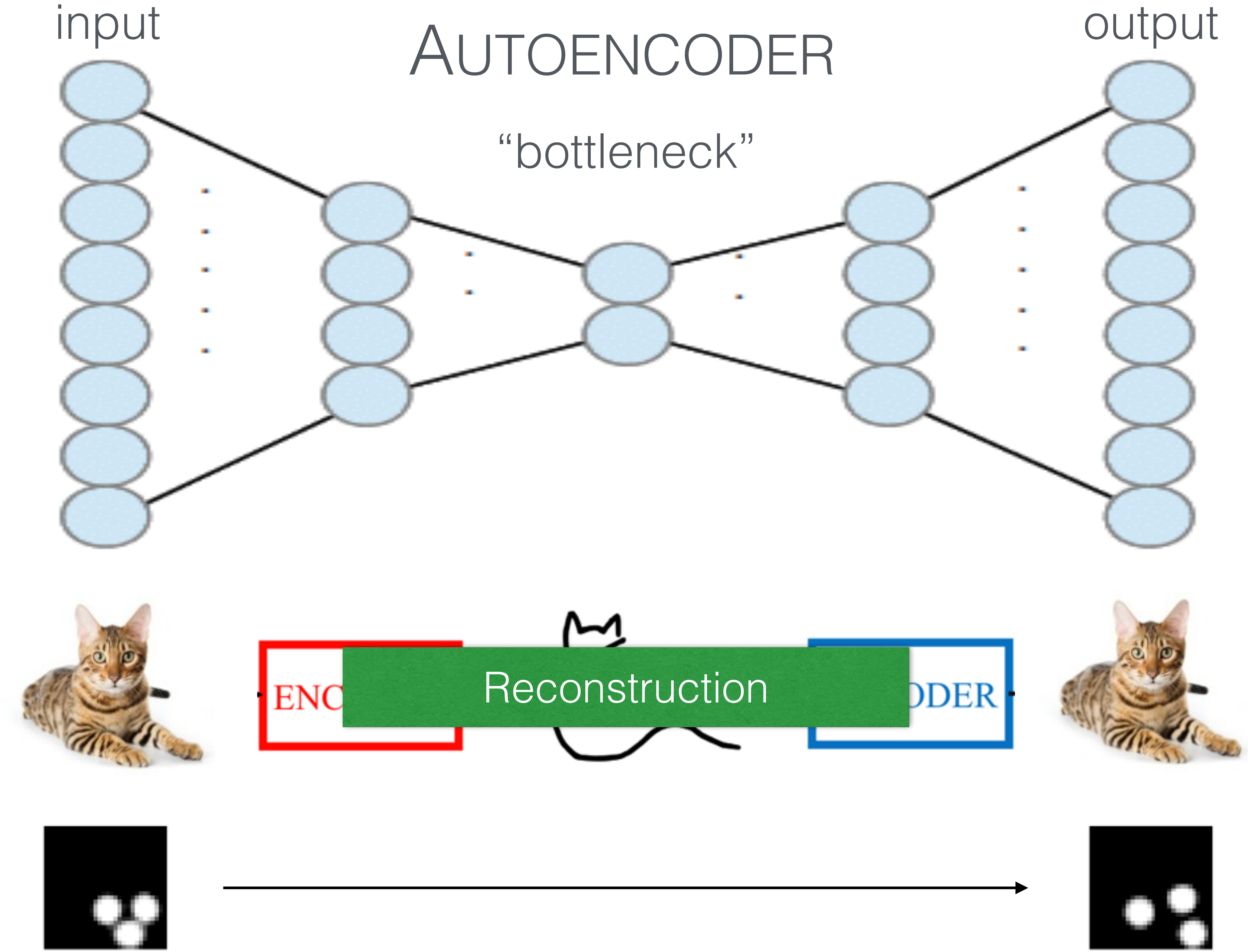


AUTOENCODER





AUTOENCODER

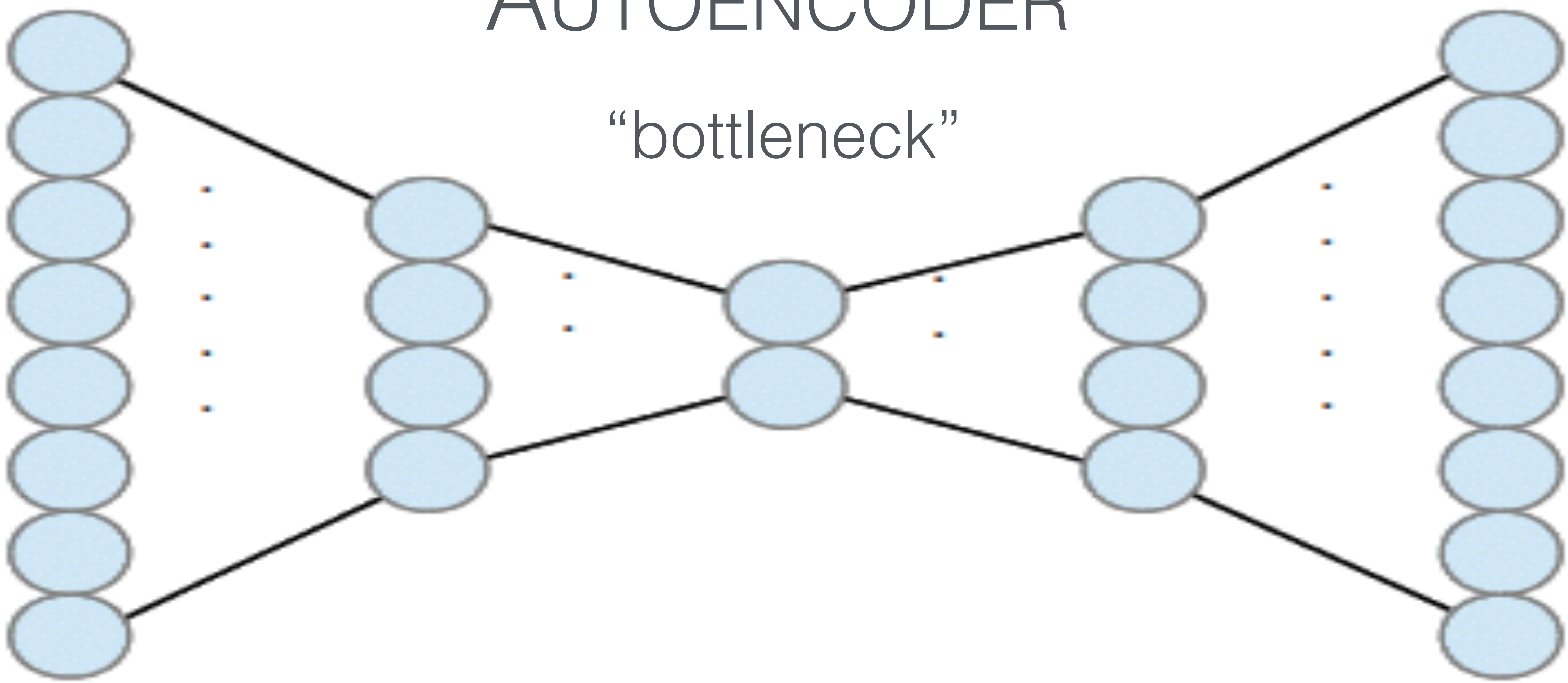


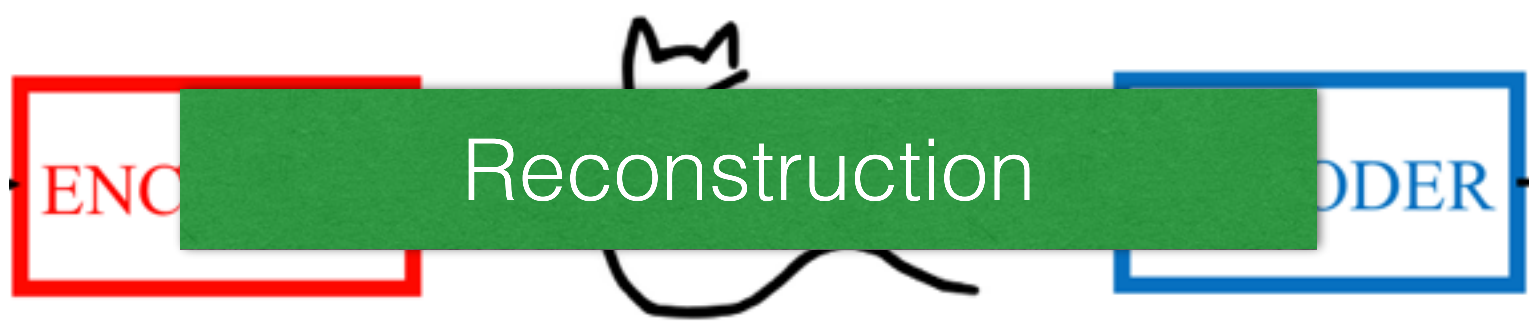
AUTOENCODER

input

output

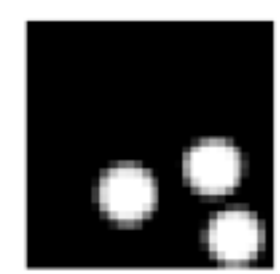
“bottleneck”



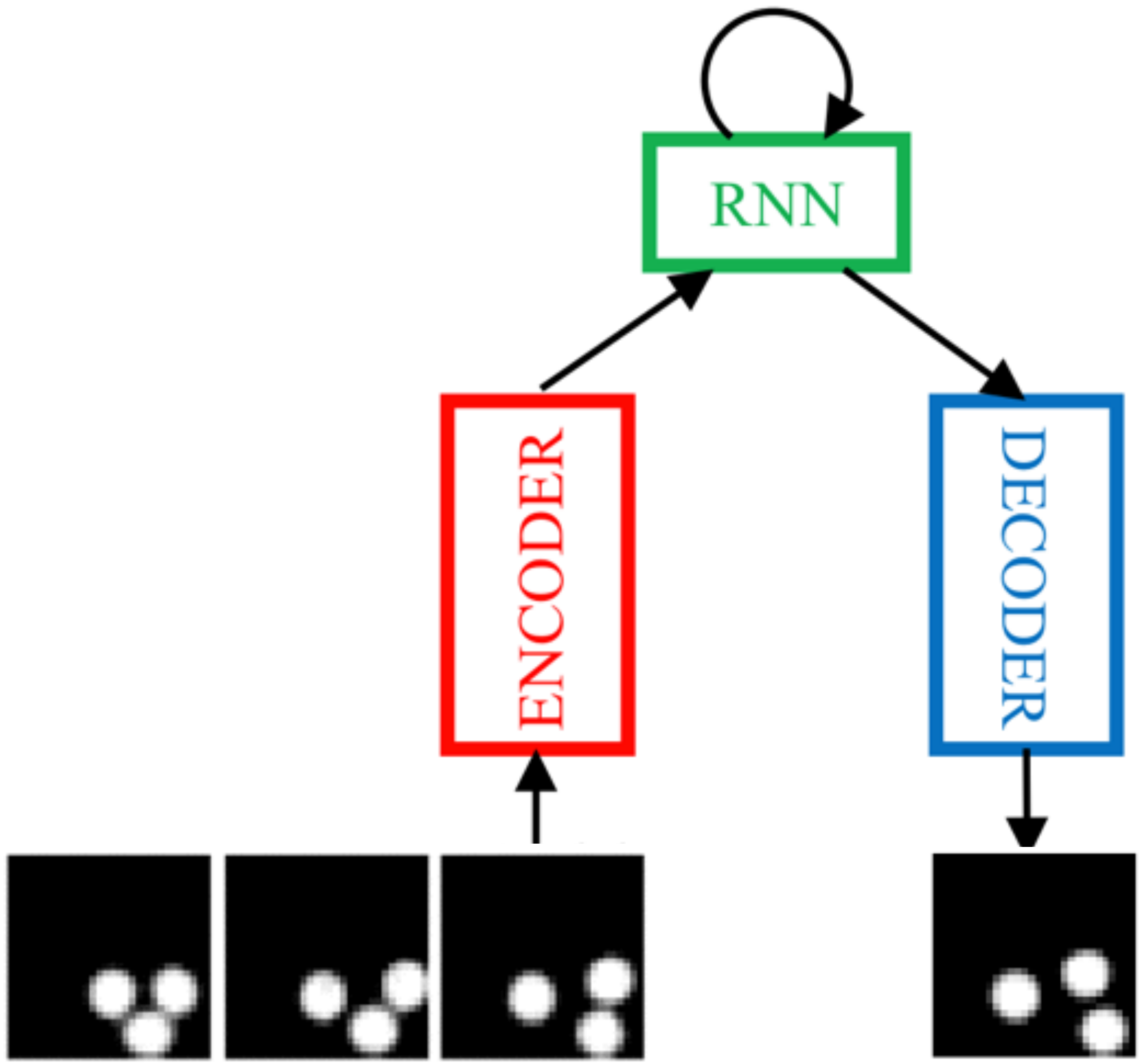


Reconstruction

Can we do prediction?

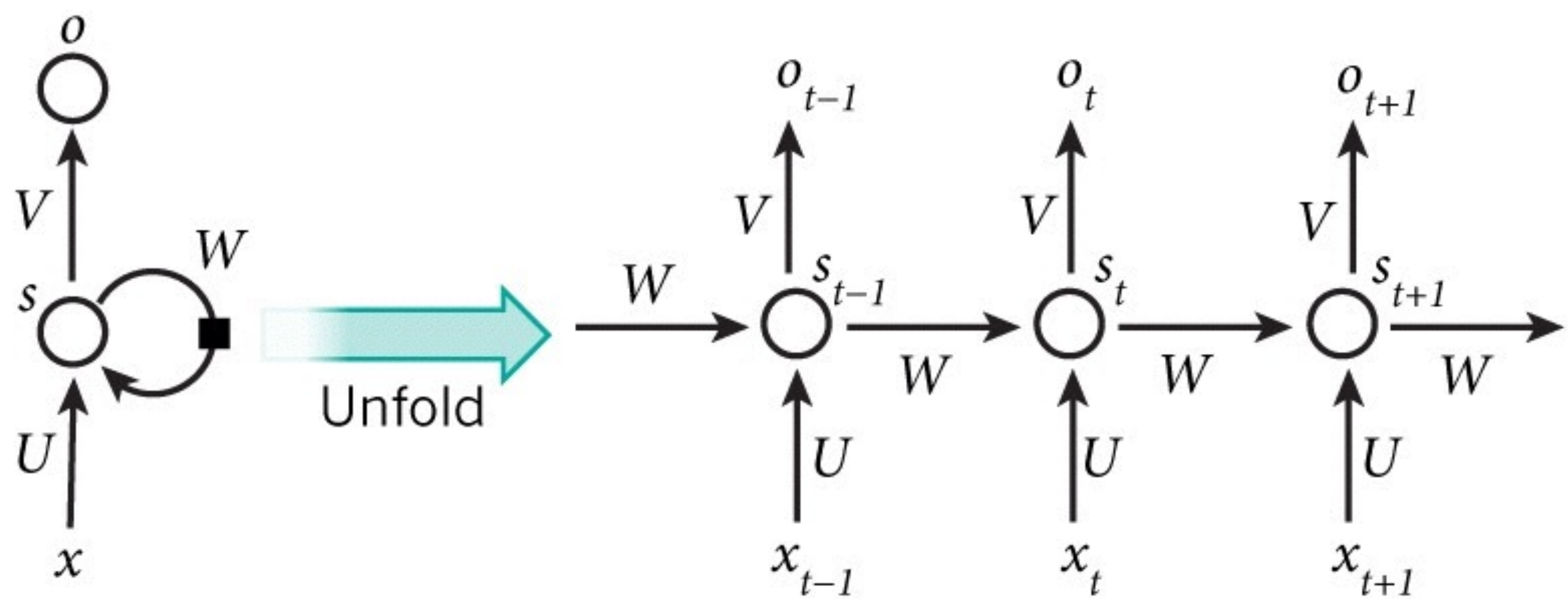


PREDICTIVE GENERATIVE NETWORK
(a.k.a "Predictive Encoder" Palm 2012)

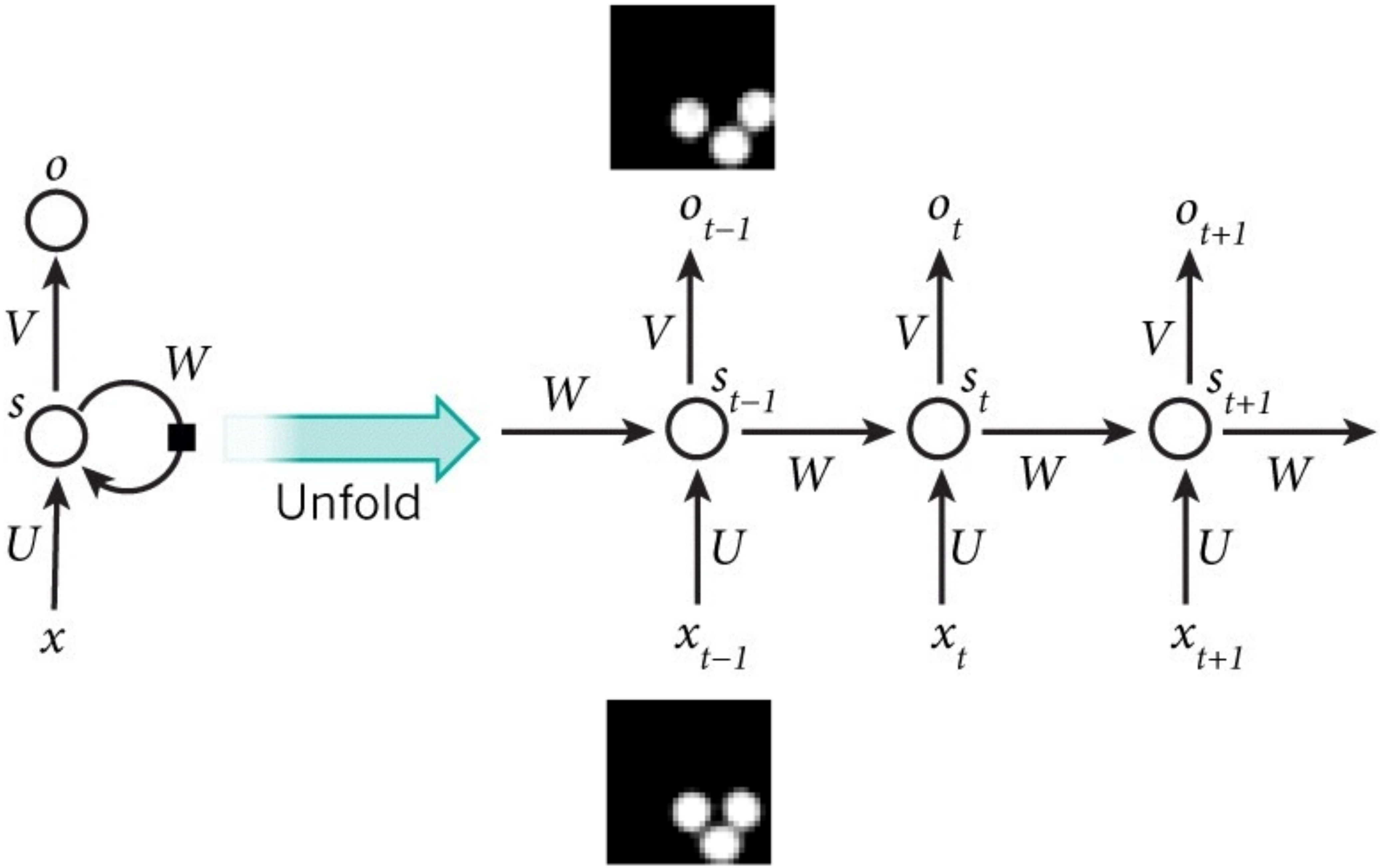


$x^i_{1:t}$ $G(x^i_{1:t})$ vs. x^i_{t+1}

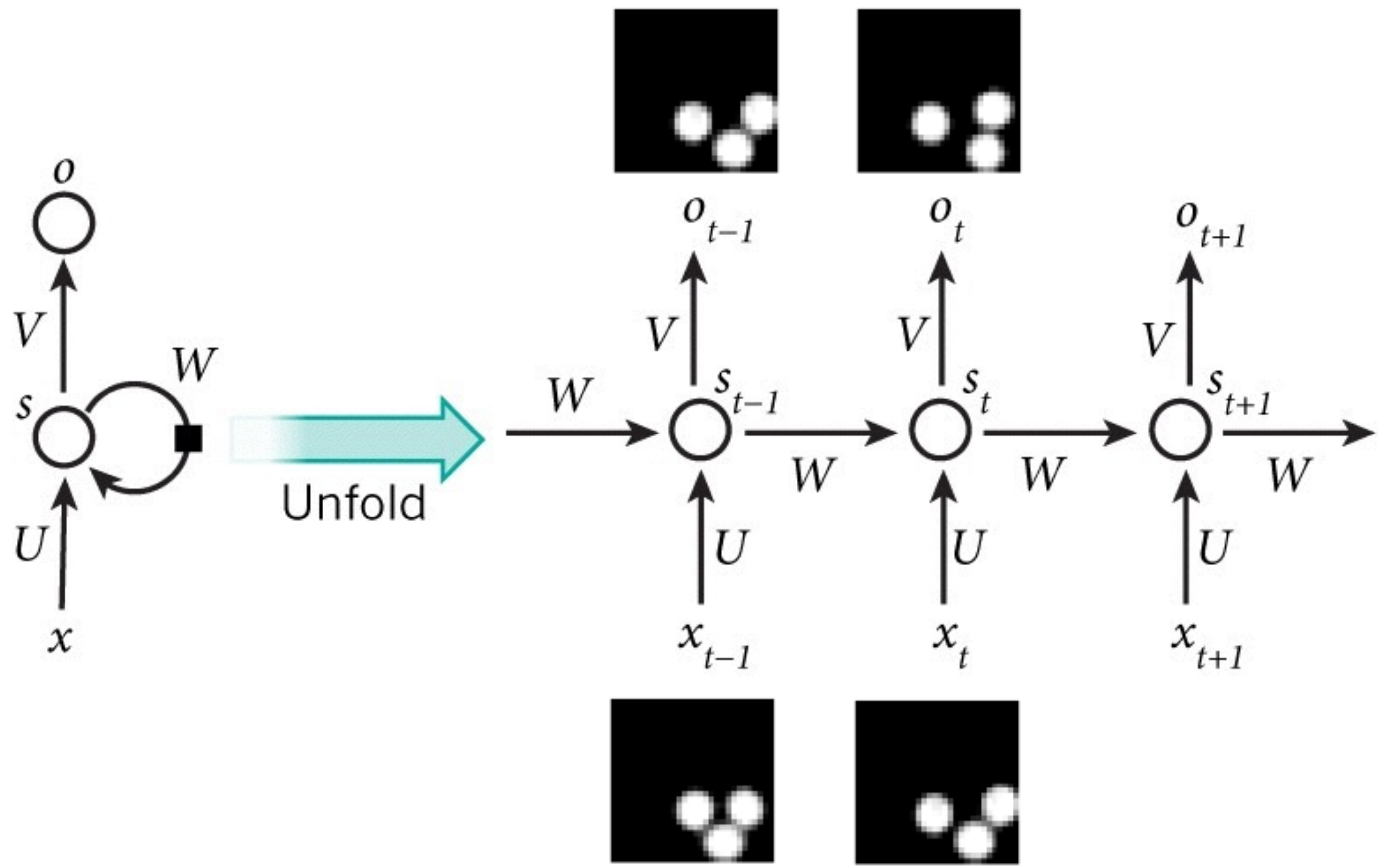
Recurrent Neural Network



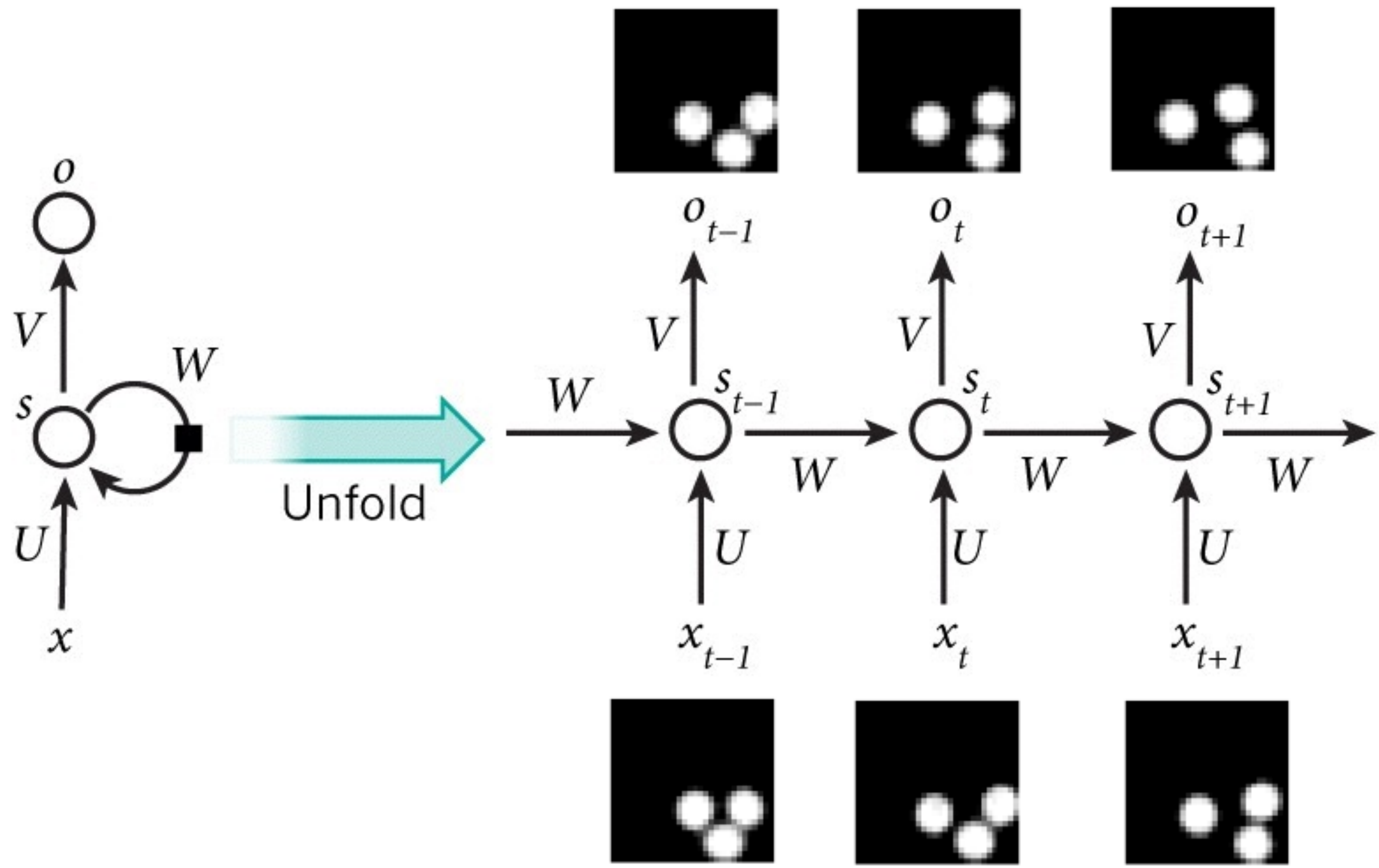
Recurrent Neural Network



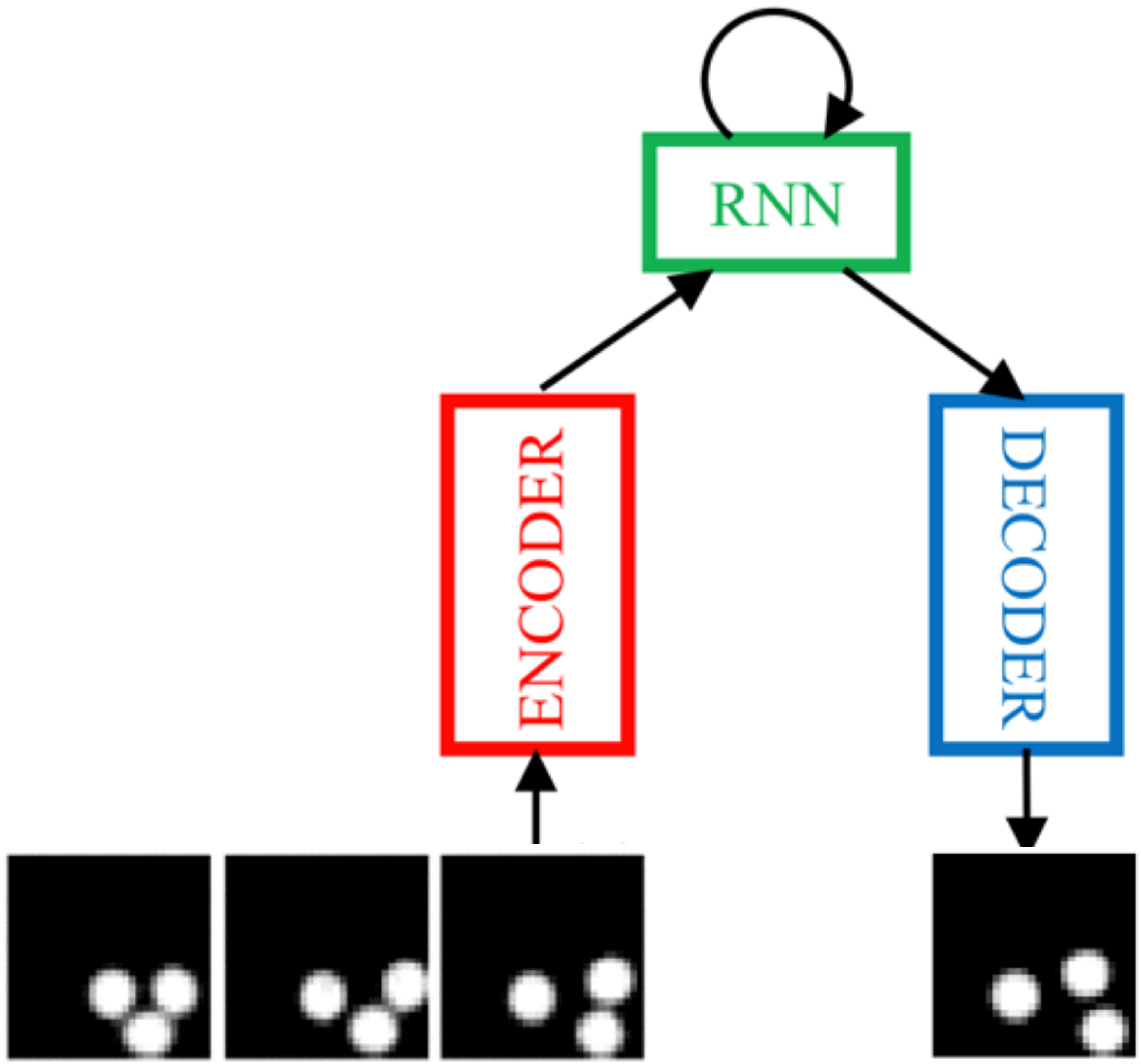
Recurrent Neural Network



Recurrent Neural Network

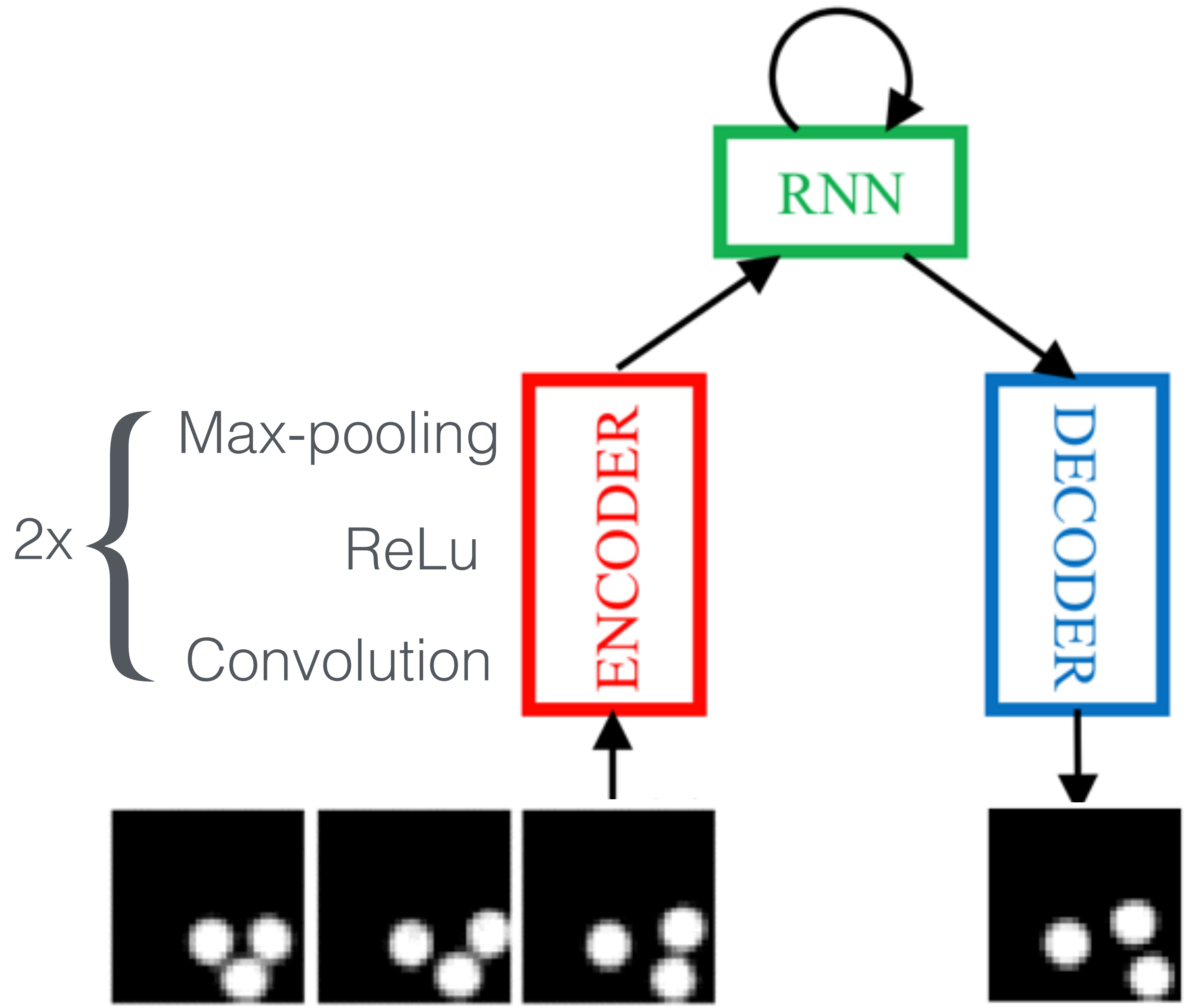


Predictive Generative Network
(a.k.a "Predictive Encoder" Palm 2012)



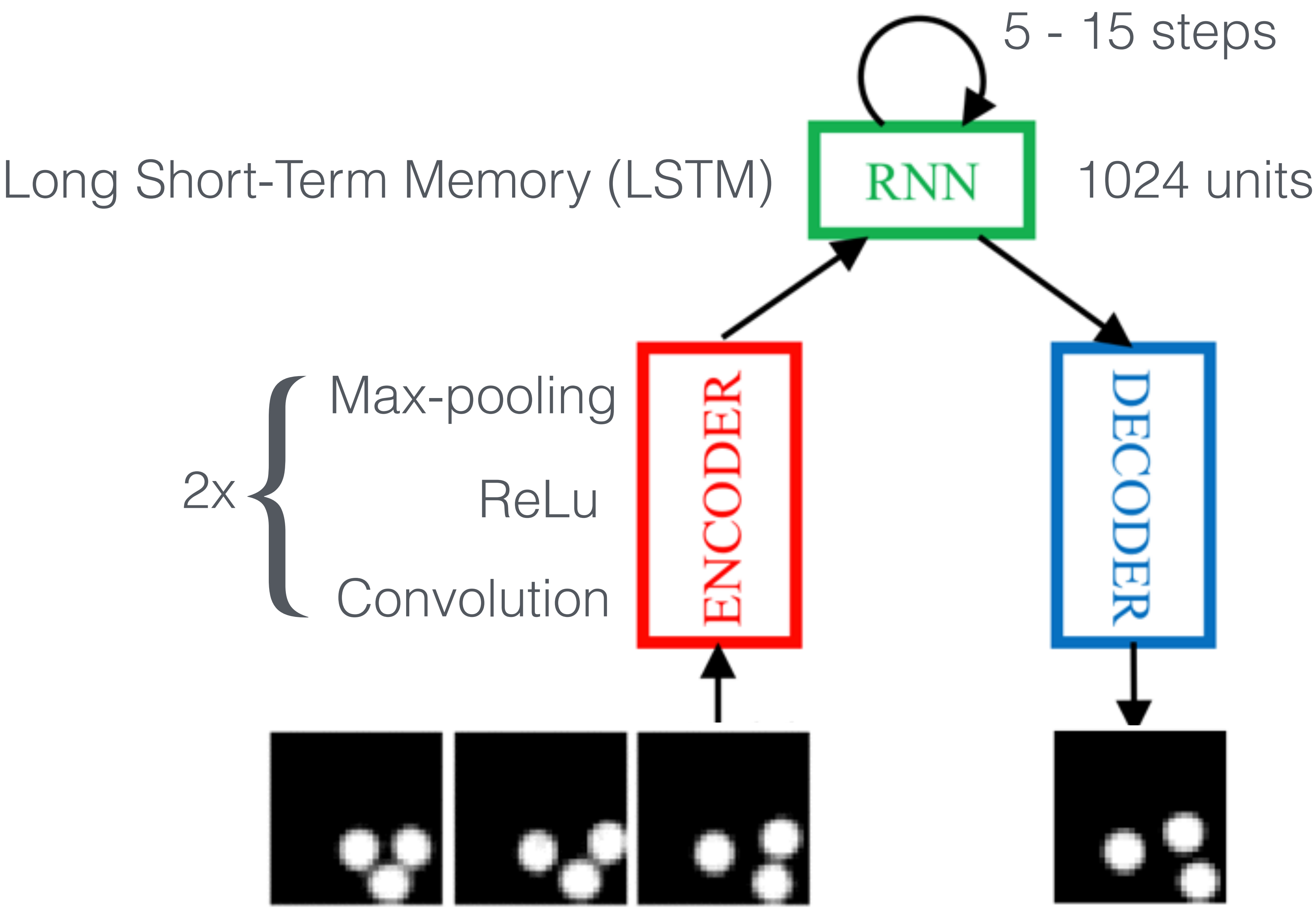
$x^i_{1:t}$ $G(x^i_{1:t})$ vs. x^i_{t+1}

Predictive Generative Network
(a.k.a "Predictive Encoder" Palm 2012)



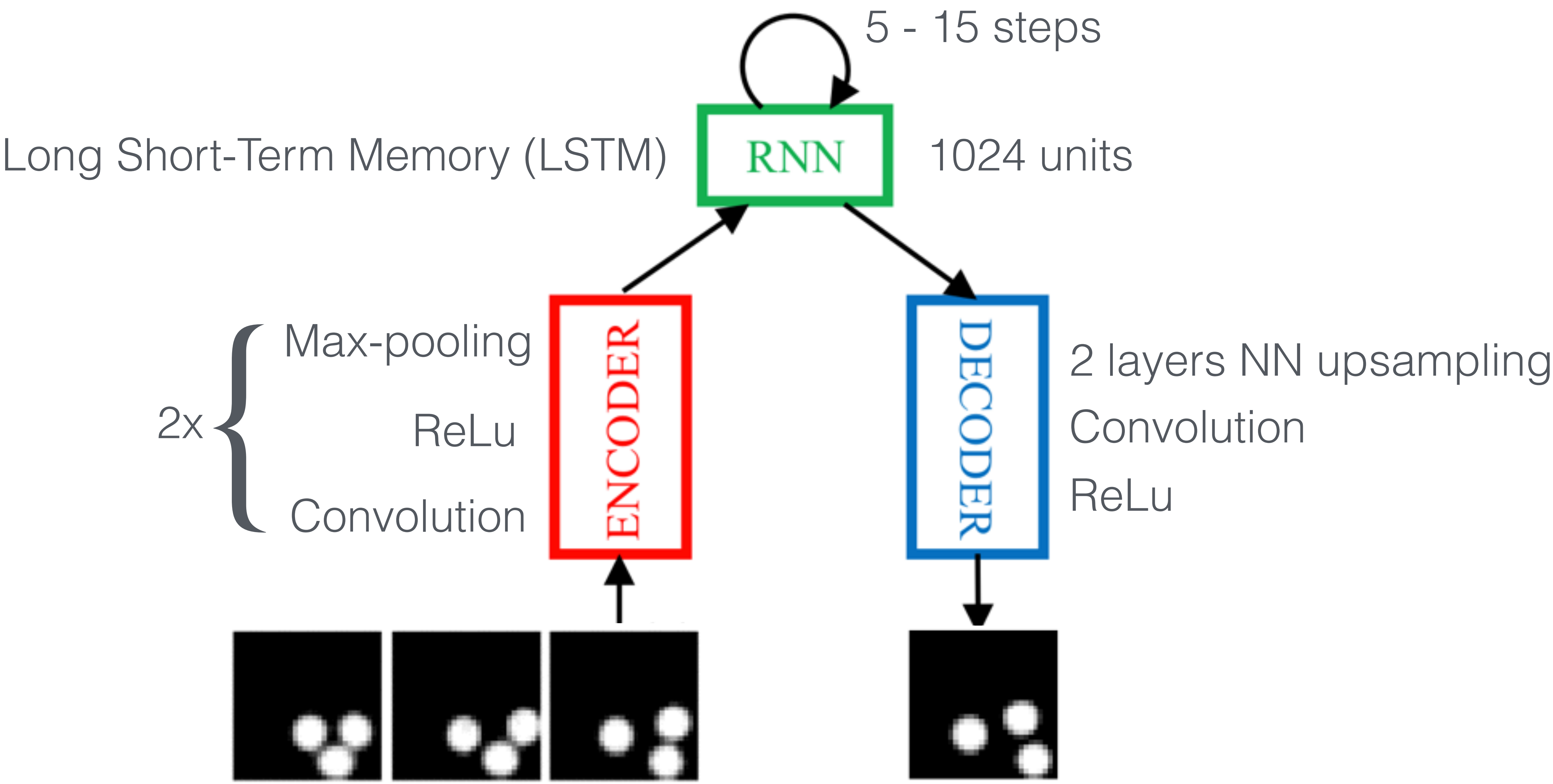
$x^i_{1:t}$ $G(x^i_{1:t})$ vs. x^i_{t+1}

Predictive Generative Network
(a.k.a "Predictive Encoder" Palm 2012)



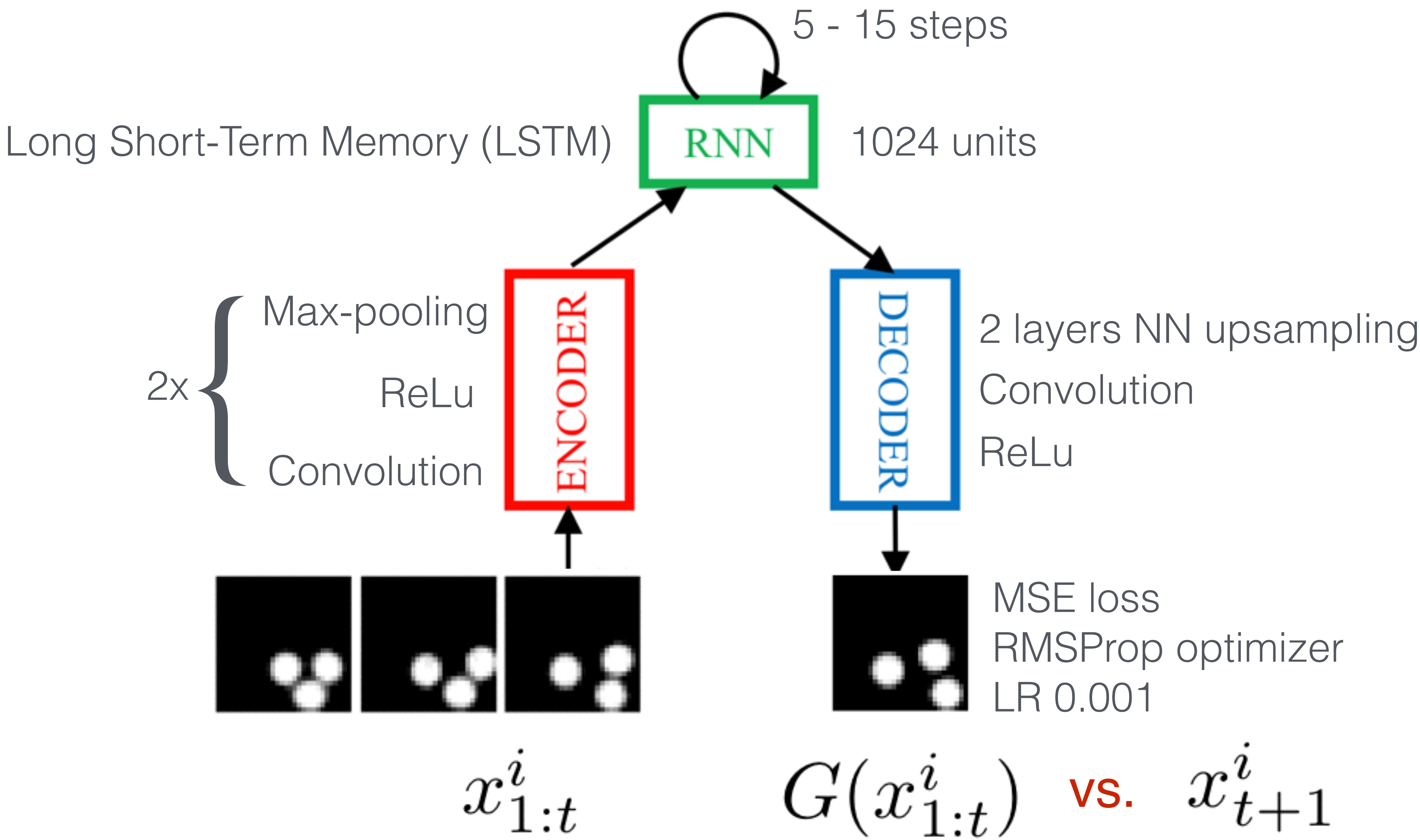
$x^i_{1:t}$ $G(x^i_{1:t})$ vs. x^i_{t+1}

Predictive Generative Network
(a.k.a "Predictive Encoder" Palm 2012)

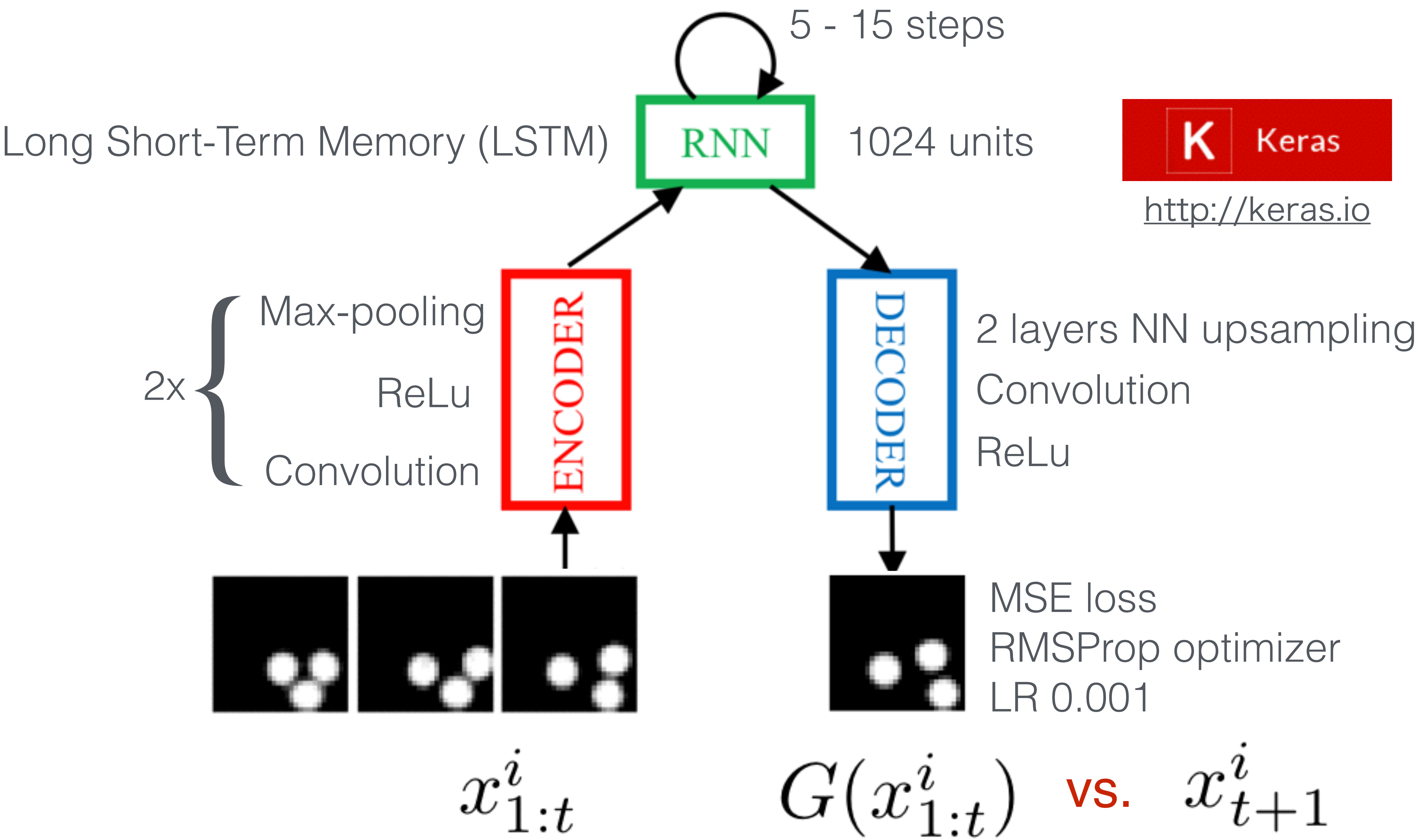


$x^i_{1:t}$ $G(x^i_{1:t})$ vs. x^i_{t+1}

Predictive Generative Network
(a.k.a "Predictive Encoder" Palm 2012)



PREDICTIVE GENERATIVE NETWORK
(a.k.a "Predictive Encoder" Palm 2012)



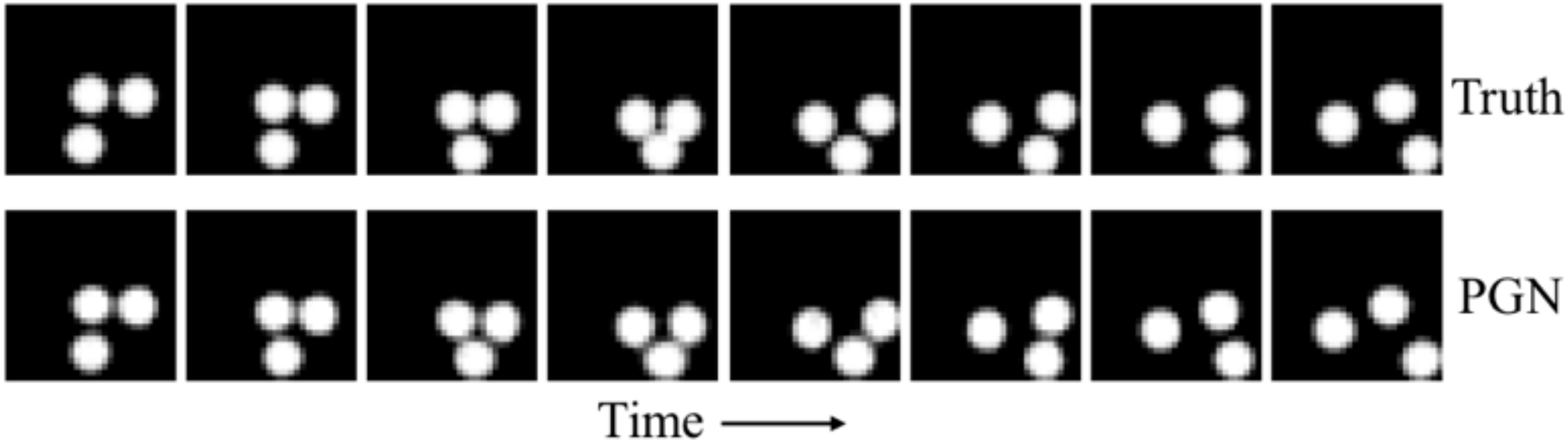
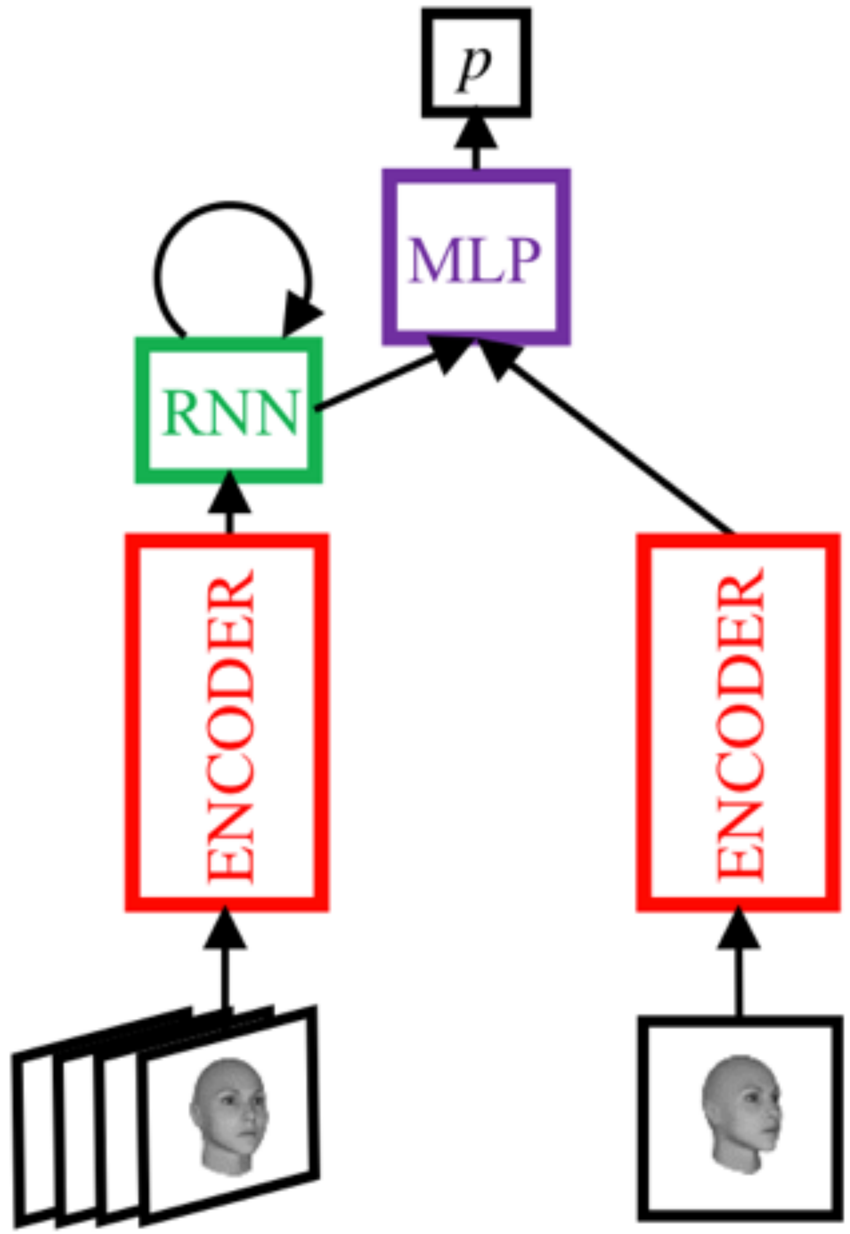


Figure 2: Example prediction sequence the for bouncing balls dataset. Predictions are repeatedly generated one step ahead using the prior ten frames as input.

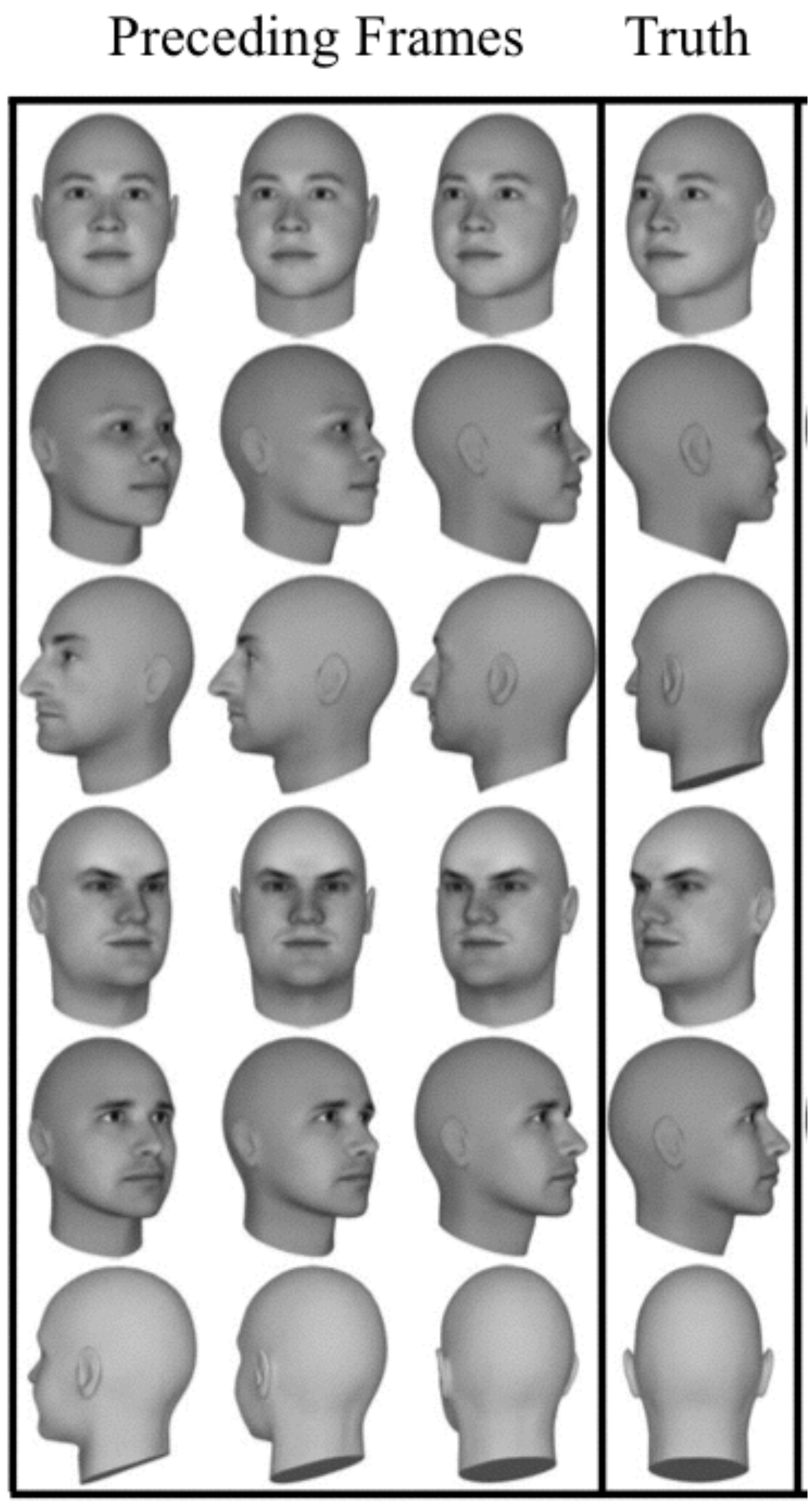
Table 1: Average prediction error for the bouncing balls dataset. †(Gan et al., 2015) ◇(Mittelman et al., 2014)

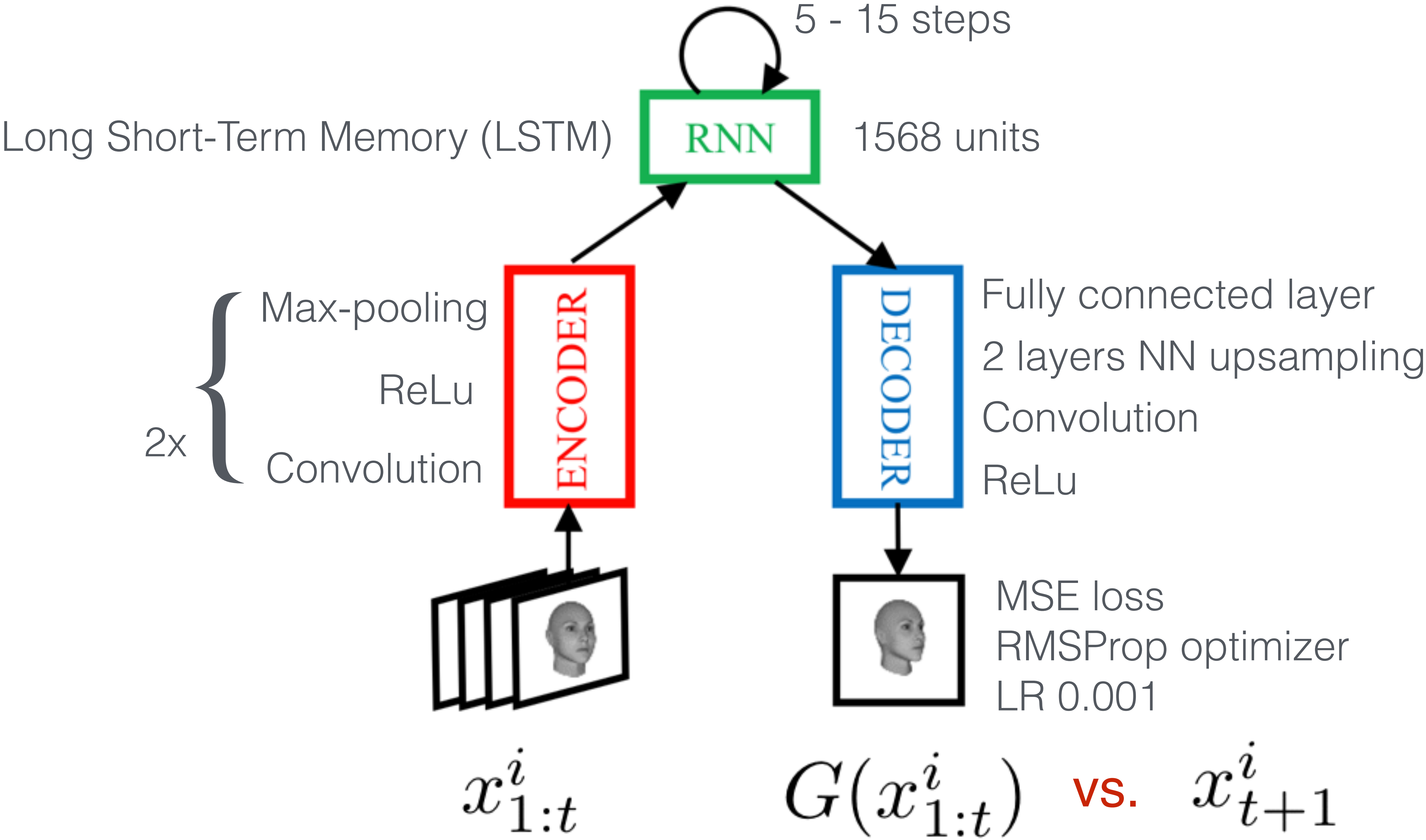
Model	Error
PGN	$\mathbf{0.65 \pm 0.11}$
DTSBN †	2.79 ± 0.39
SRTRBM ◇	3.31 ± 0.33
RTRBM ◇	3.88 ± 0.33

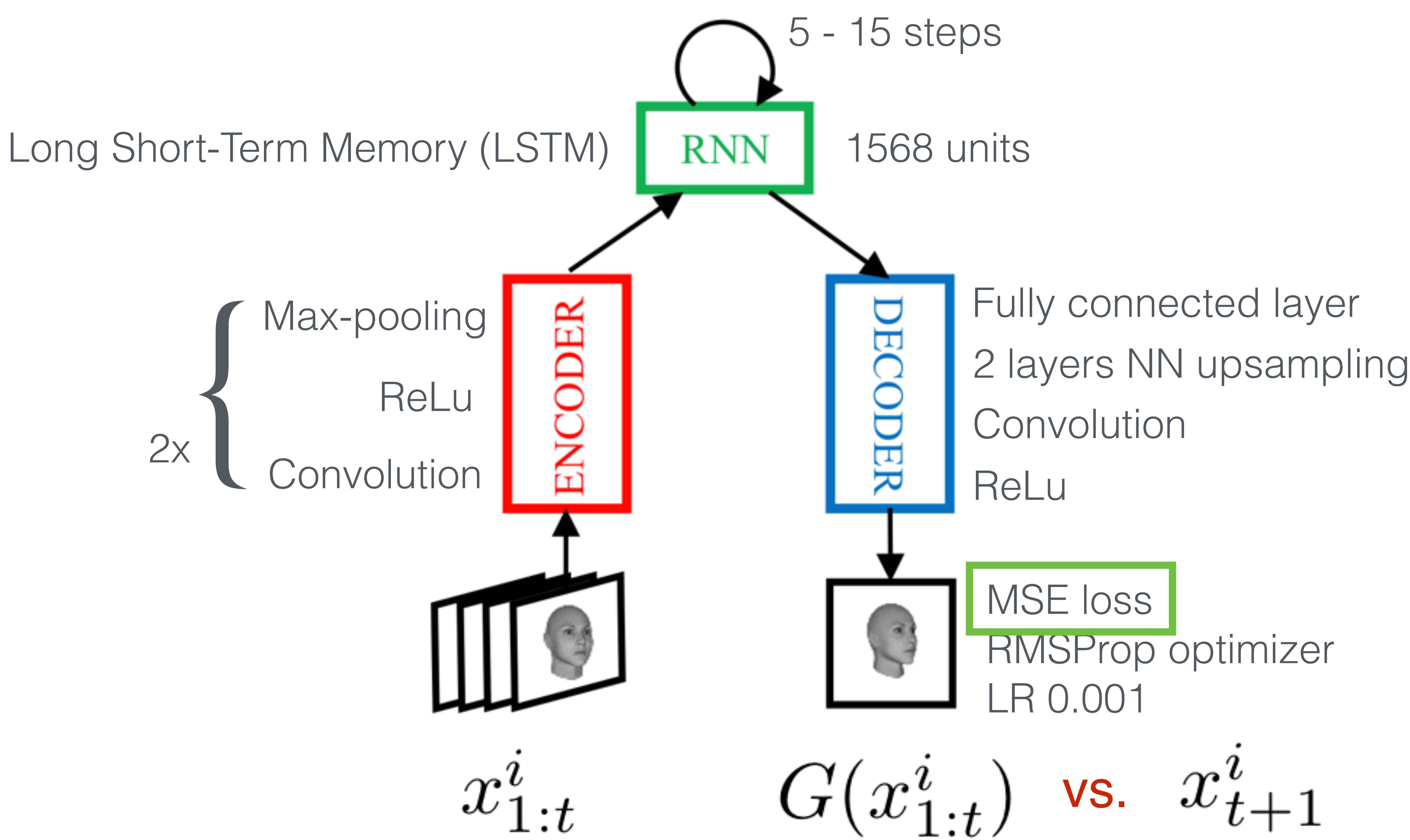


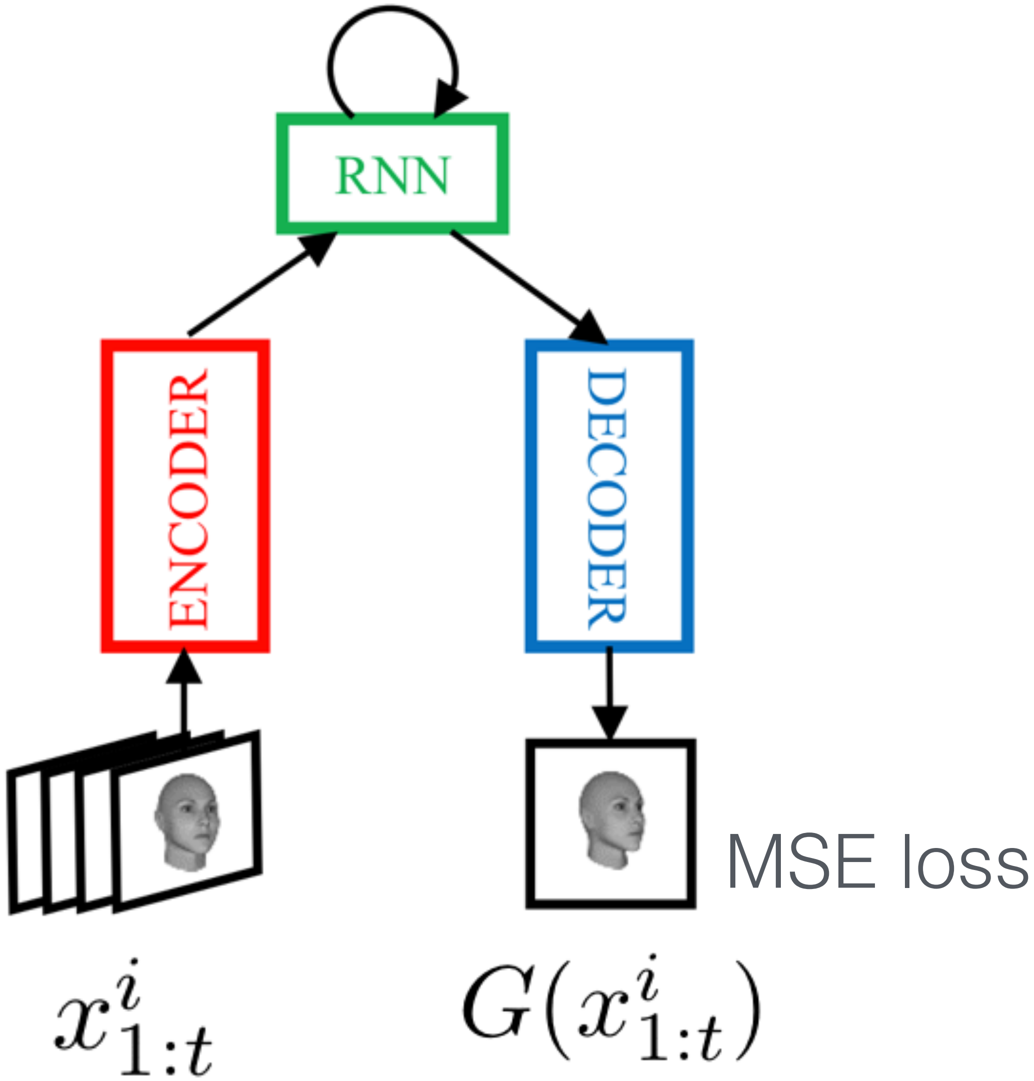
Part II
Adversarial Loss

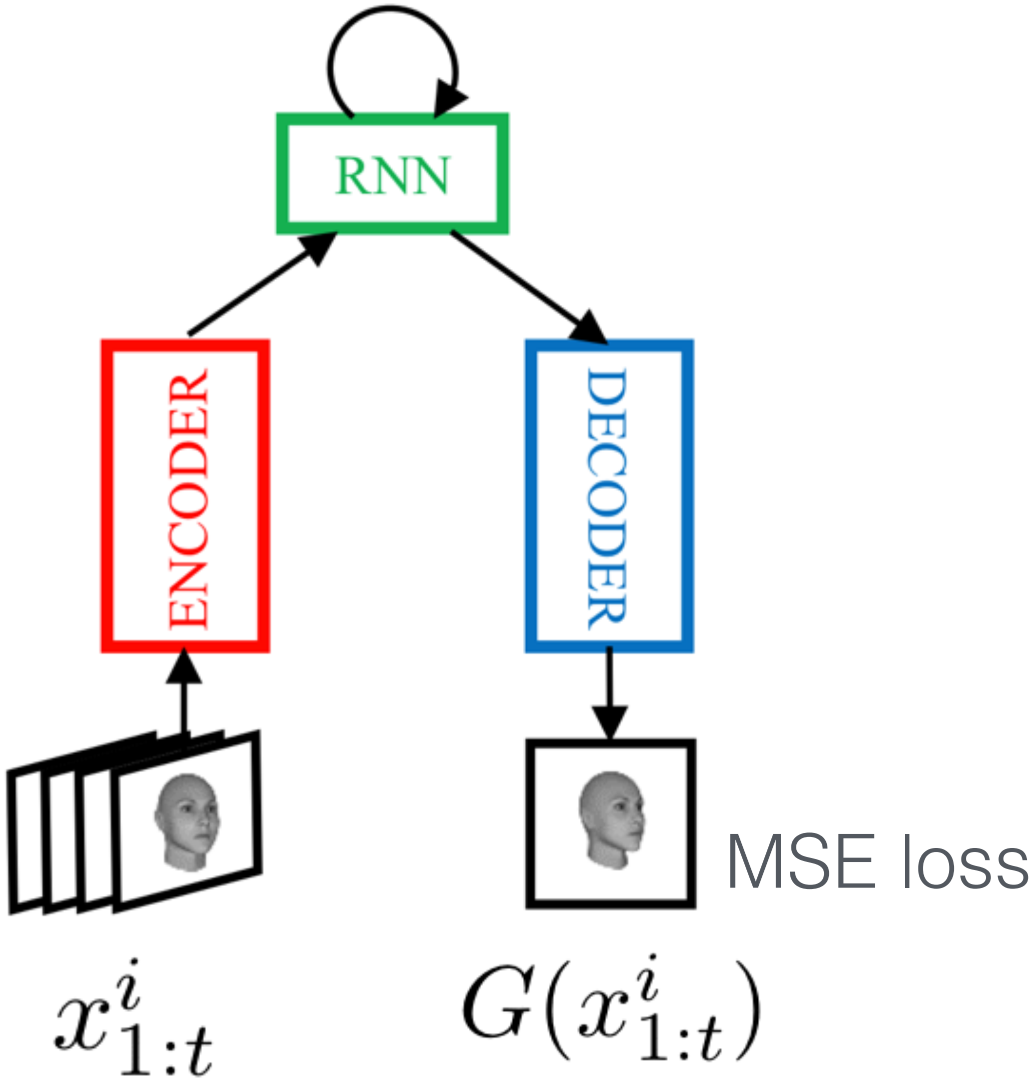
"the generator is trained to maximally confuse the adversarial discriminator"

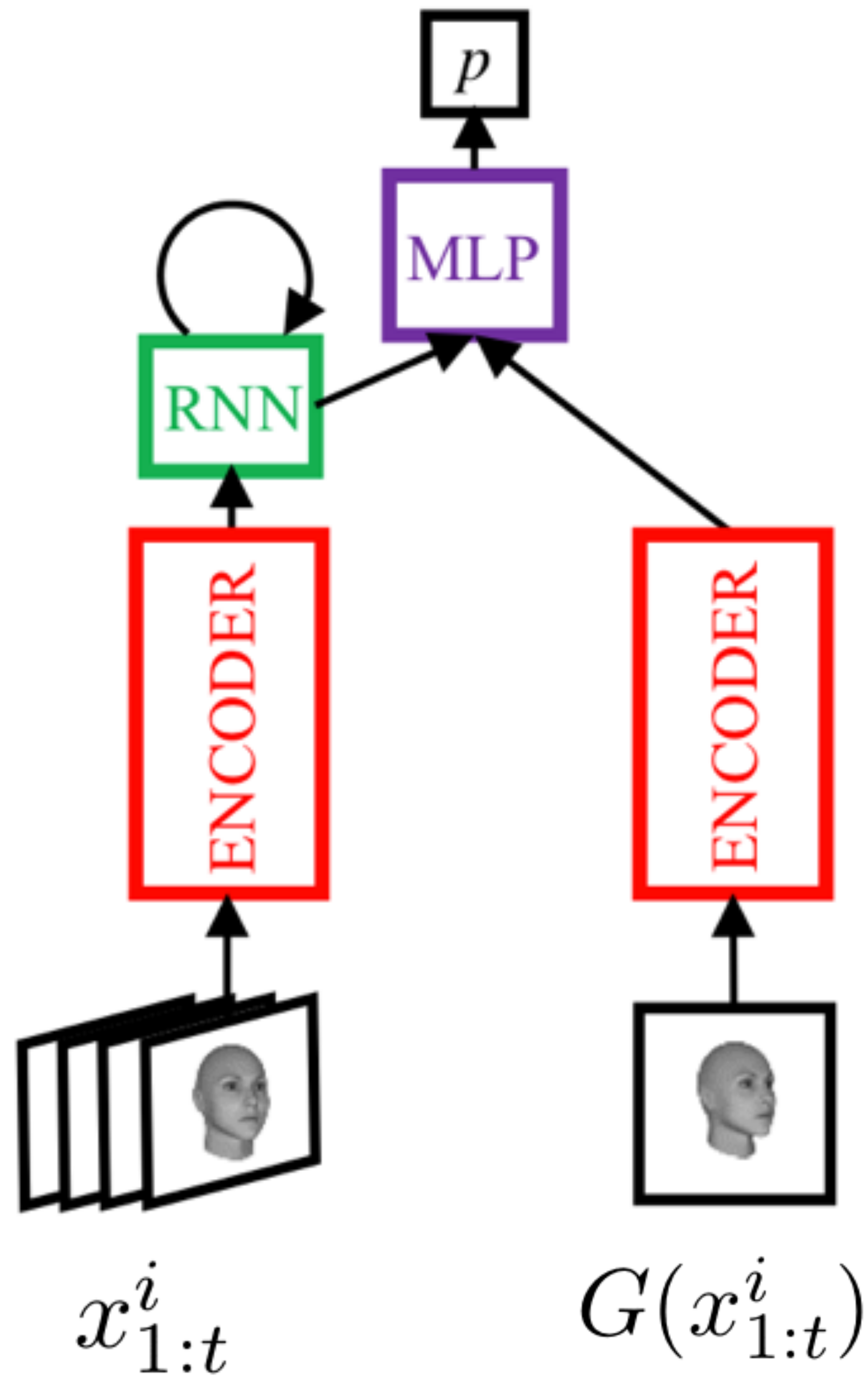

Preceding Frames
Truth

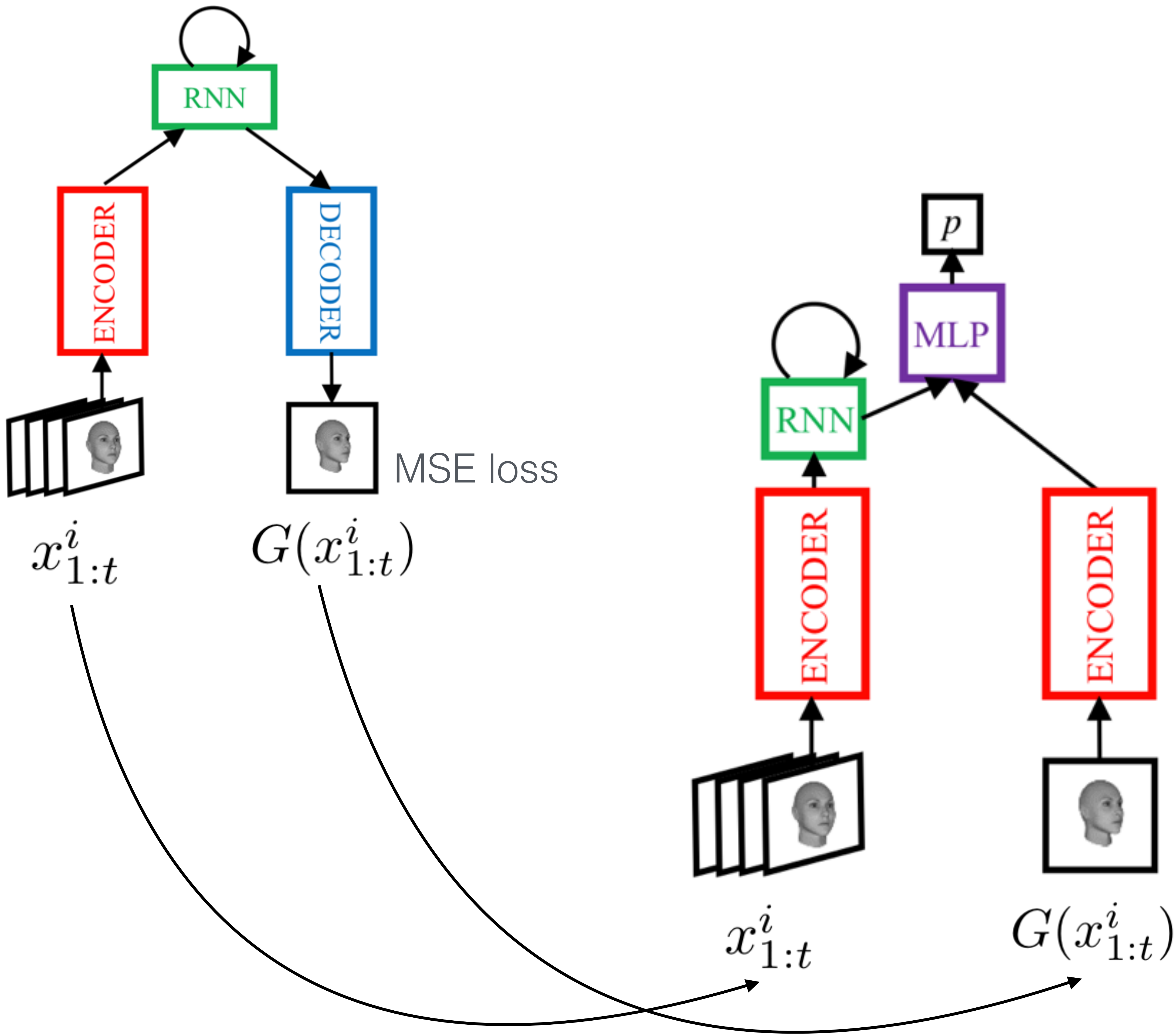

5 - 15 steps
Long Short-Term Memory (LSTM)
RNN
1568 units
2x
Max-pooling
ReLu
Convolution
ENCODER
DECODER
Fully connected layer
2 layers NN upsampling
Convolution
ReLu
MSE loss
RMSProp optimizer
LR 0.001
$x^i_{1:t}$
$G(x^i_{1:t})$ vs. x^i_{t+1}


5 - 15 steps
Long Short-Term Memory (LSTM)
RNN
1568 units
Max-pooling
ReLu
Convolution
2x
ENCODER
DECODER
Fully connected layer
2 layers NN upsampling
Convolution
ReLu
MSE loss
RMSProp optimizer
LR 0.001
$x^i_{1:t}$
$G(x^i_{1:t})$ vs. x^i_{t+1}

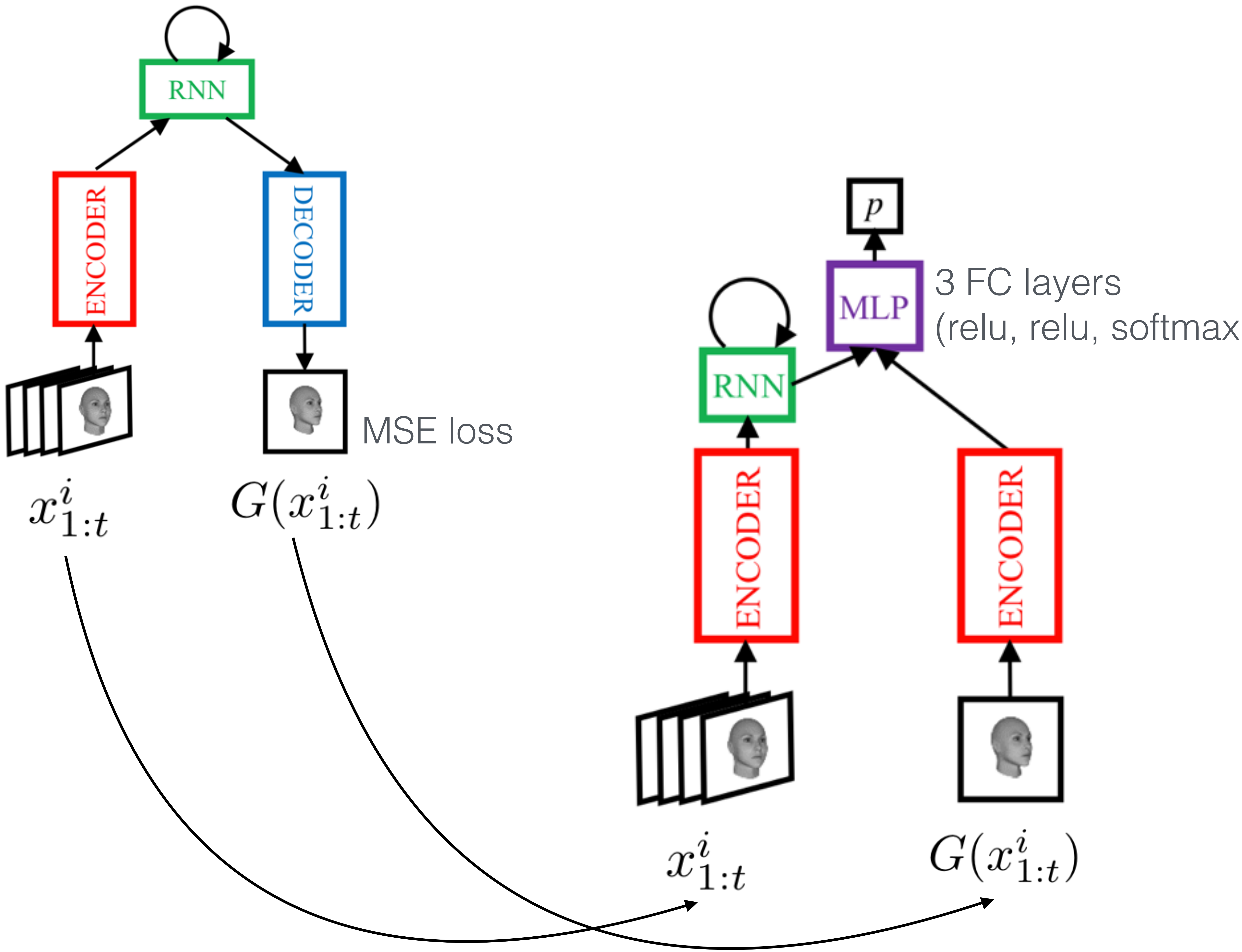

RNN
ENCODER
DECODER
MSE loss
$x^i_{1:t}$
$G(x^i_{1:t})$


RNN
ENCODER
DECODER
MSE loss
$x^i_{1:t}$
$G(x^i_{1:t})$

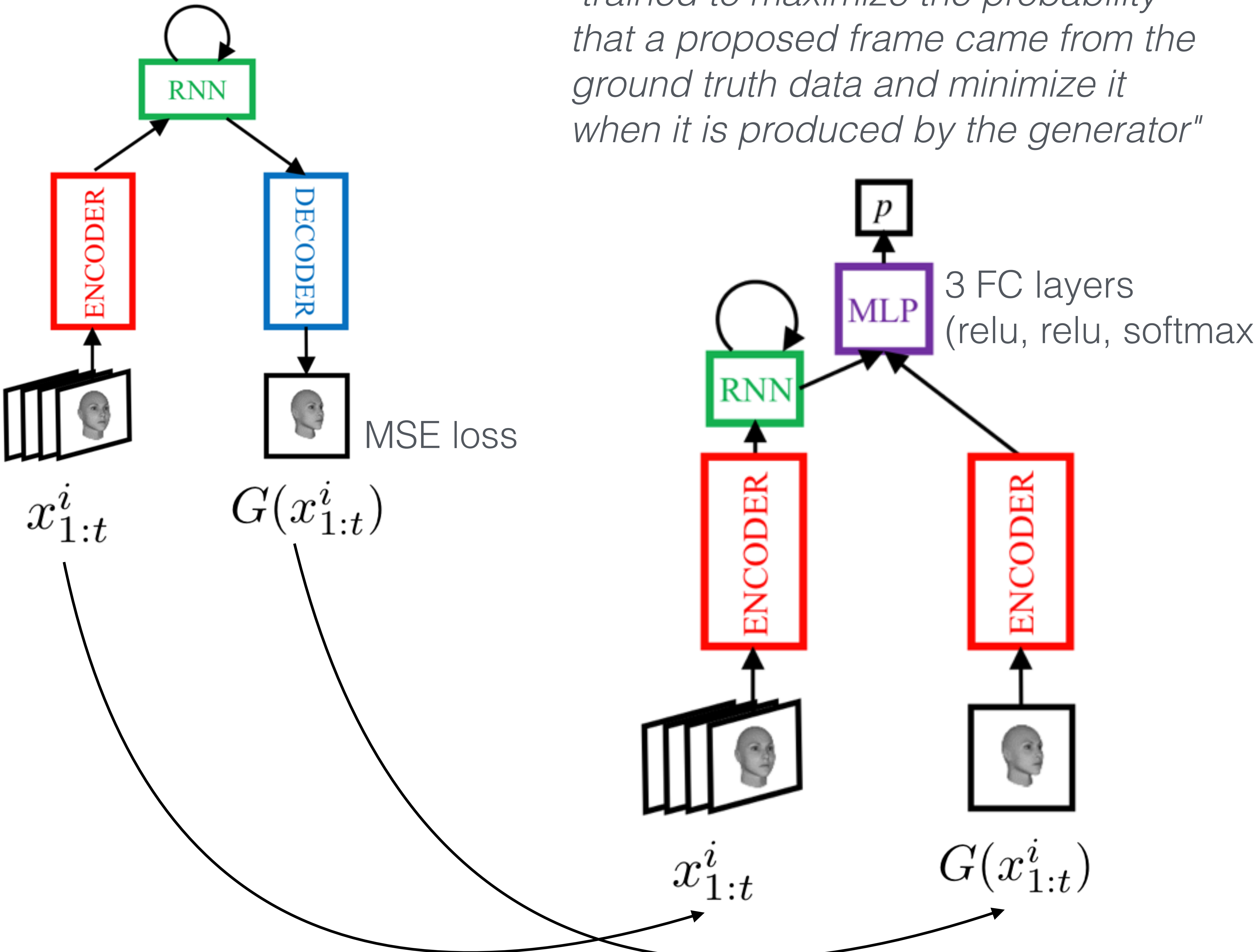

p
MLP
RNN
ENCODER
ENCODER
$x^i_{1:t}$
$G(x^i_{1:t})$

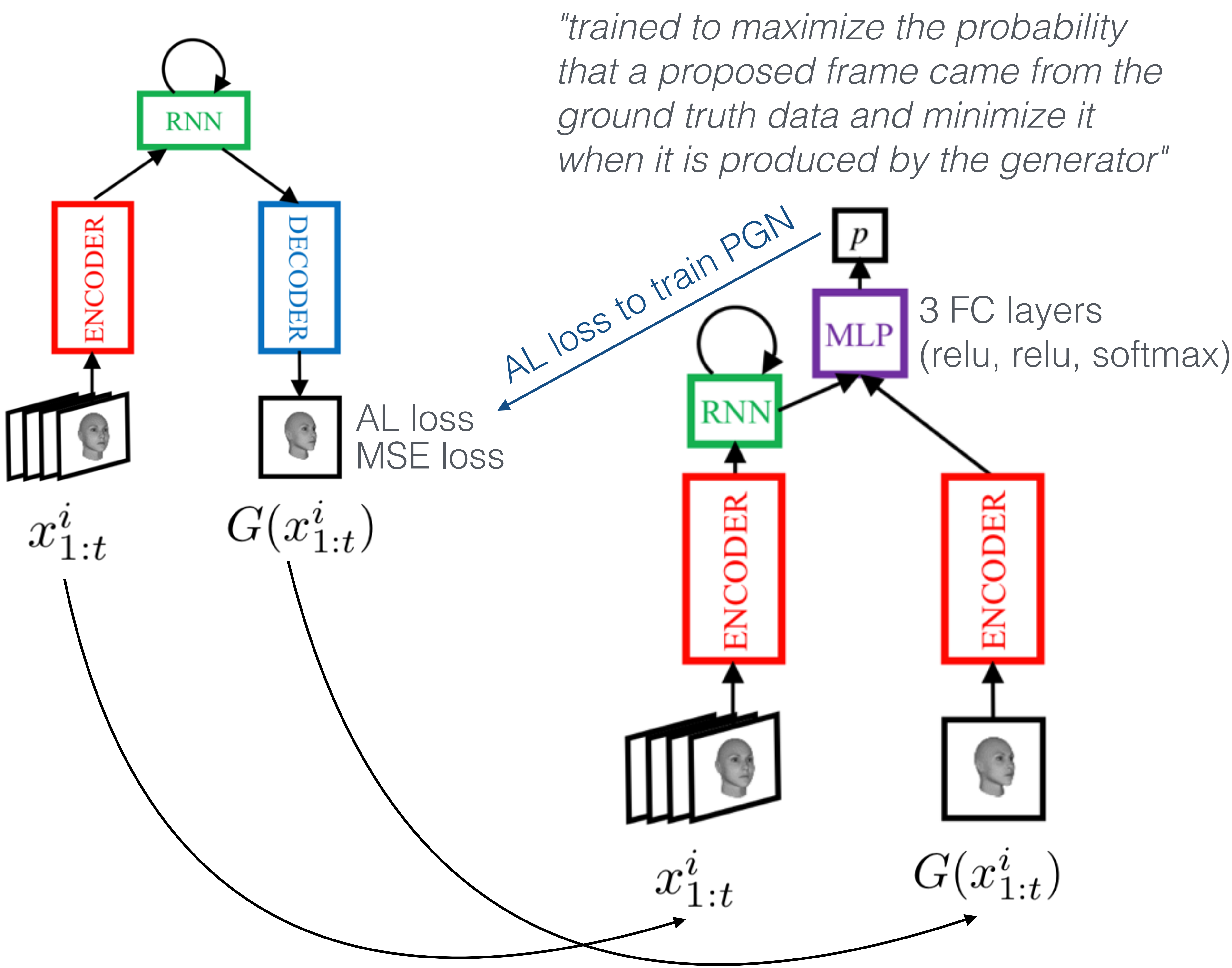


RNN
ENCODER
DECODER
MSE loss
$x^i_{1:t}$
$G(x^i_{1:t})$
p
MLP
RNN
ENCODER
ENCODER
$x^i_{1:t}$
$G(x^i_{1:t})$


RNN
ENCODER
DECODER
MSE loss
$x^i_{1:t}$
$G(x^i_{1:t})$
p
MLP
3 FC layers
(relu, relu, softmax)
RNN
ENCODER
ENCODER
$x^i_{1:t}$
$G(x^i_{1:t})$

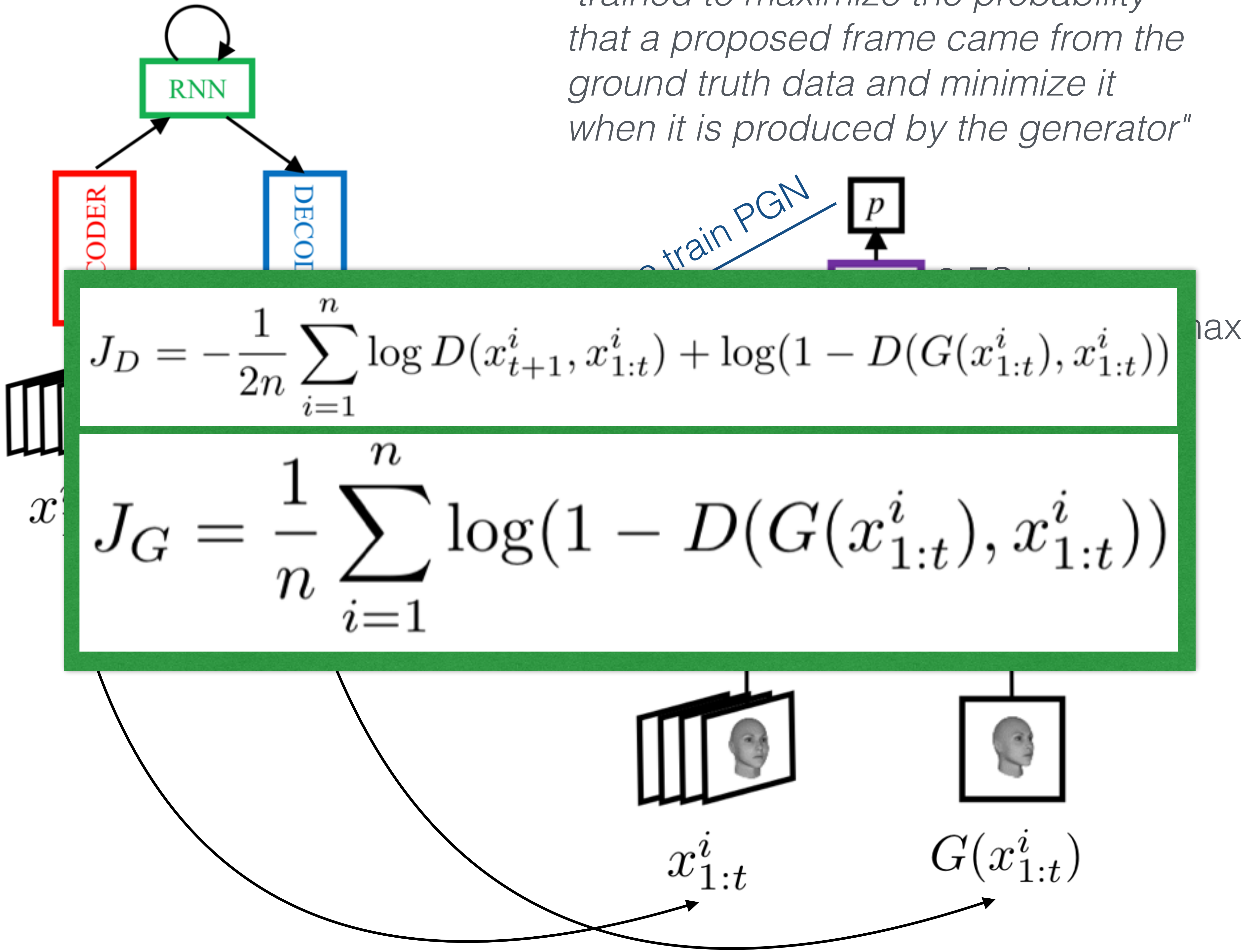
"trained to maximize the probability that a proposed frame came from the ground truth data and minimize it when it is produced by the generator"





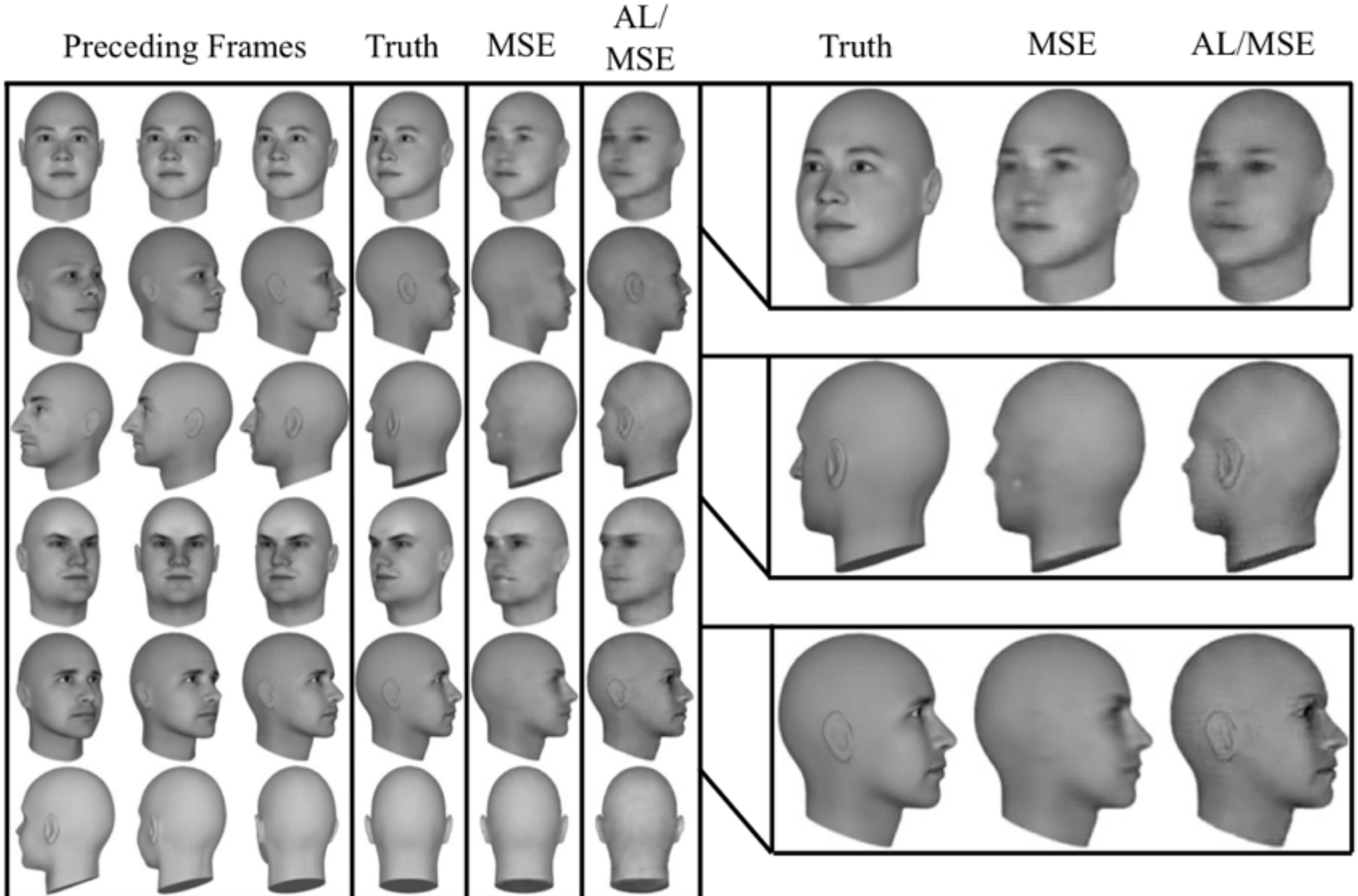
"trained to maximize the probability that a proposed frame came from the ground truth data and minimize it when it is produced by the generator"
RNN
ENCODER
DECODER
AL loss
MSE loss
AL loss to train PGN
p
MLP
3 FC layers
(relu, relu, softmax)
RNN
ENCODER
ENCODER
$x^i_{1:t}$
$G(x^i_{1:t})$
$x^i_{1:t}$
$G(x^i_{1:t})$

"trained to maximize the probability that a proposed frame came from the ground truth data and minimize it when it is produced by the generator"



$$J_D = -\frac{1}{2n}\sum_{i=1}^{n} \log D(x^i_{t+1}, x^i_{1:t}) + \log(1 - D(G(x^i_{1:t}), x^i_{1:t}))$$

$$J_G = \frac{1}{n}\sum_{i=1}^{n} \log(1 - D(G(x^i_{1:t}), x^i_{1:t}))$$



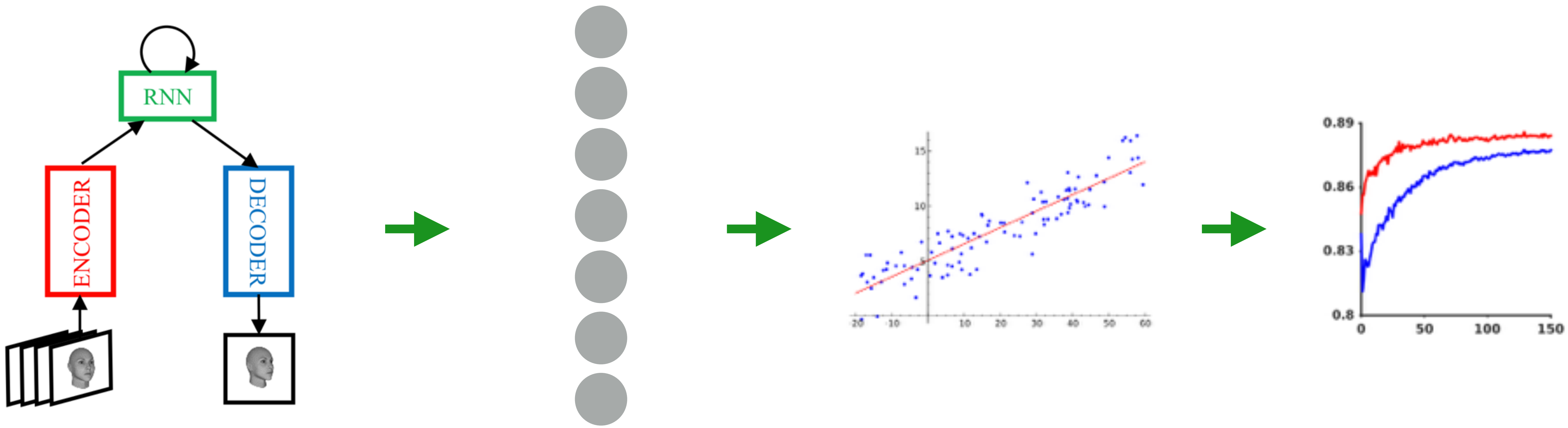
*"with **adversarial loss alone** the generator easily found solutions that fooled the discriminator, but did not look anything like the correct samples"*

***MSE** model is fairly faithful to the identities of the faces, but produces blurred versions*

***combined AL/MSE** model tends to underfit the identity towards a more average face*

Part III
Internal Representations and Latent Variables

"we are interested in understanding the representations learned by the models"

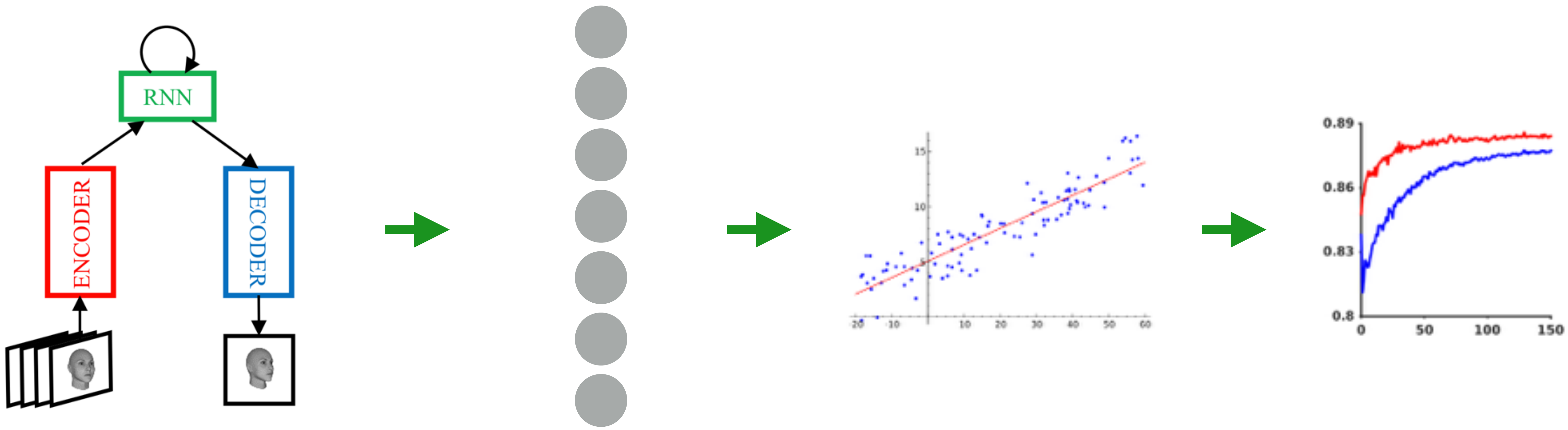


PGN model

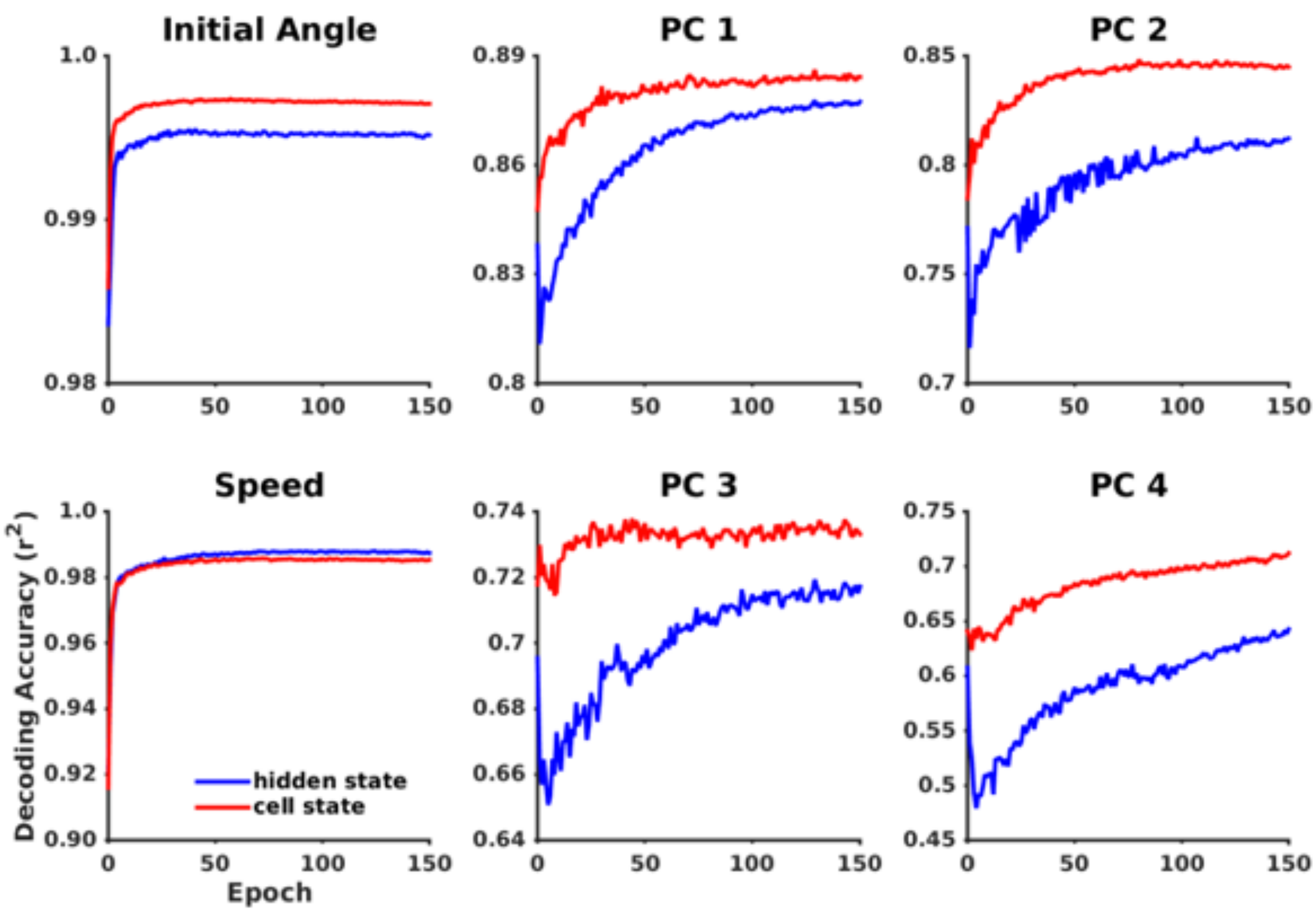
LSTM activities

L2 regression

Value of a latent variable



PGN model LSTM activities L2 regression Value of a latent variable



MULTIDIMENSIONAL SCALING

"An MDS algorithm aims to place each object in N-dimensional space such that the between-object distances are preserved as well as possible."

WIKIPEDIA
The Free Encyclopedia

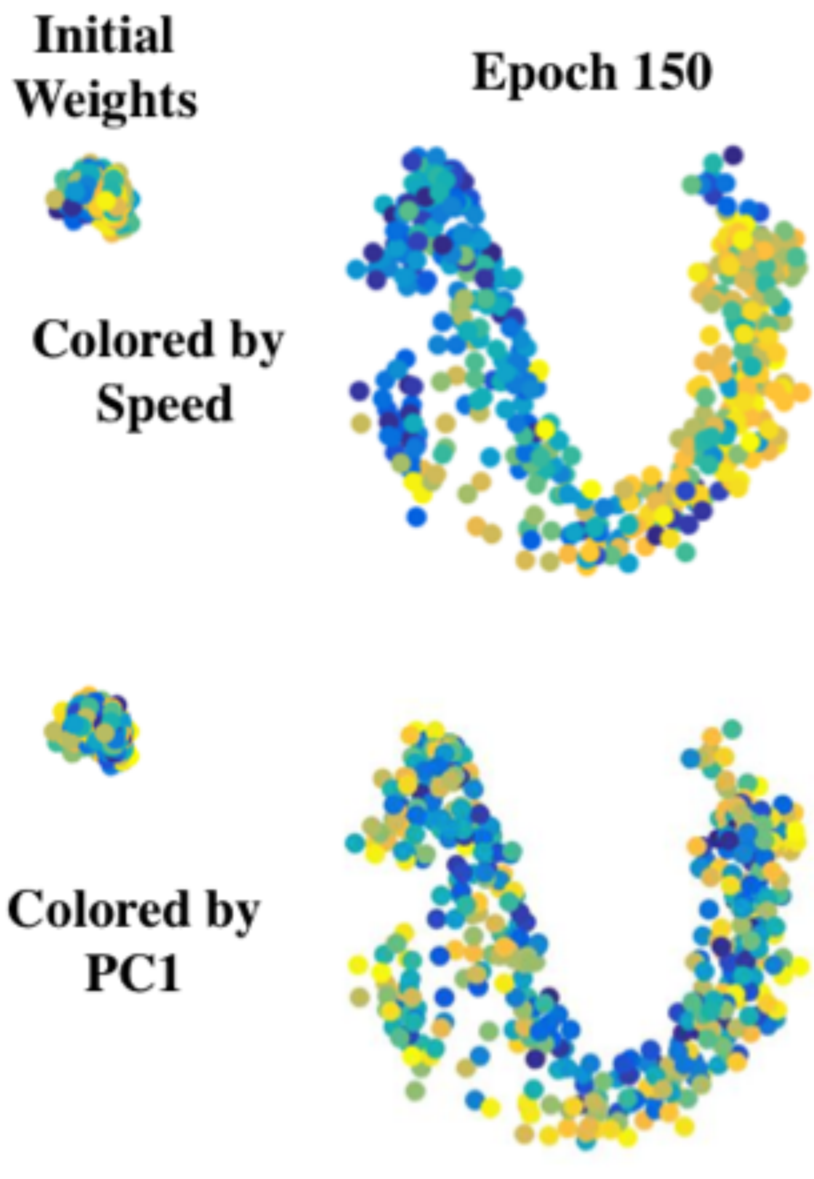


Figure 5: MDS of LSTM Space

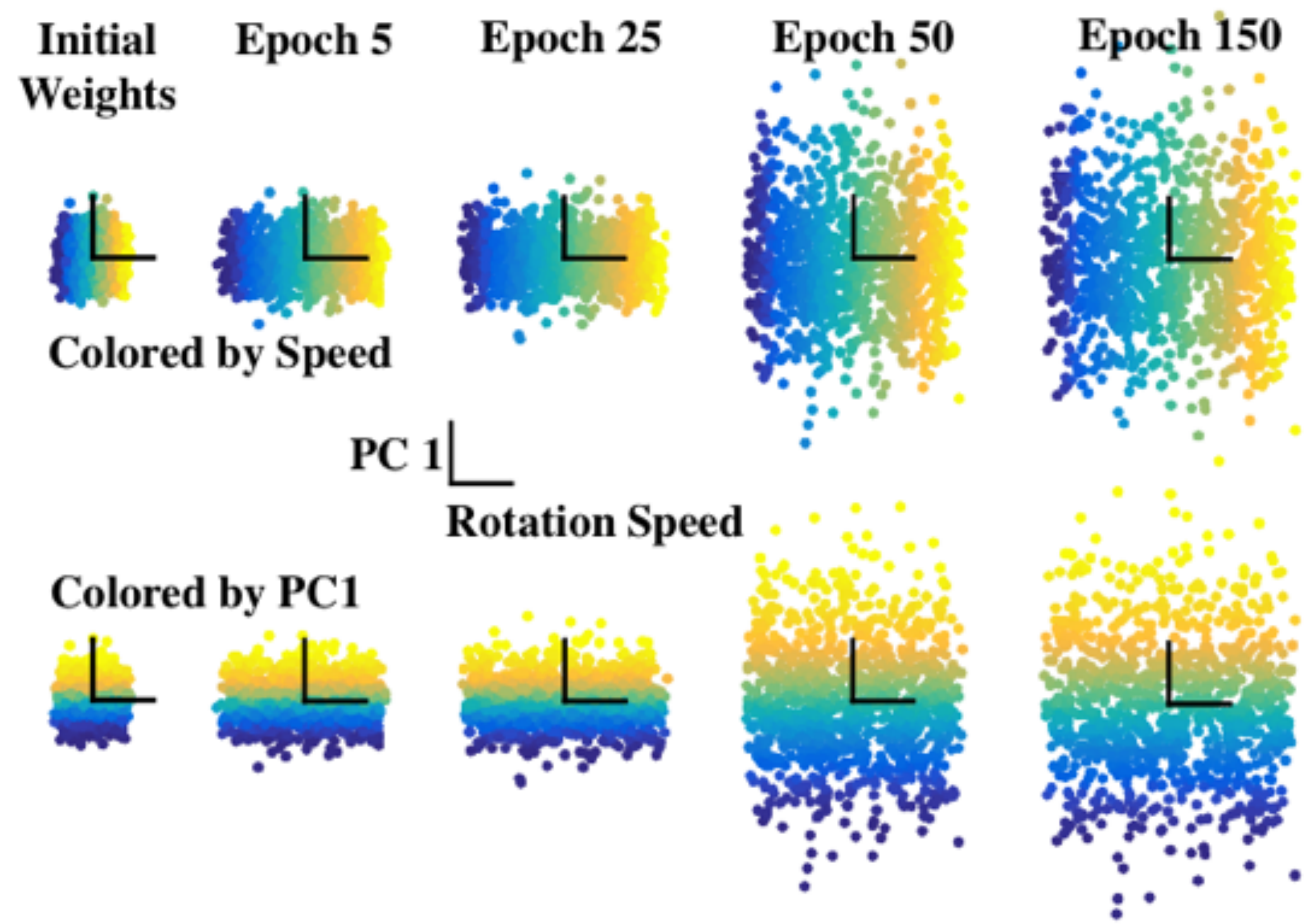
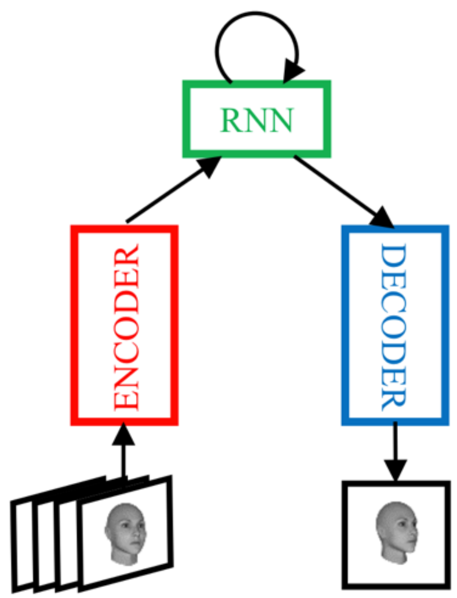


Figure 6: Projection of LSTM feature space on latent variables axes. Axes are in the direction of regression coefficients. A different regression was fit for each epoch.

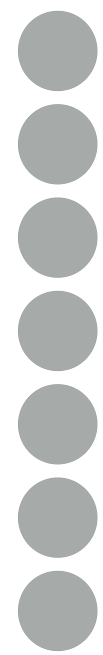
*"representations trained with a **predictive loss outperform** other models of comparable complexity in a **supervised classification problem**"*

Part IV
Usefulness of Predictive Learning

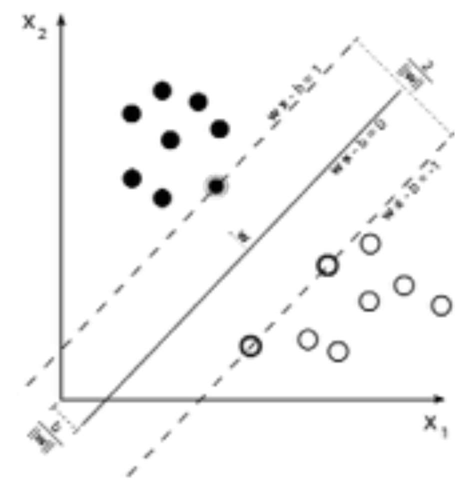
The Task: 50 randomly generated faces (12 angles per each)



Generative models:



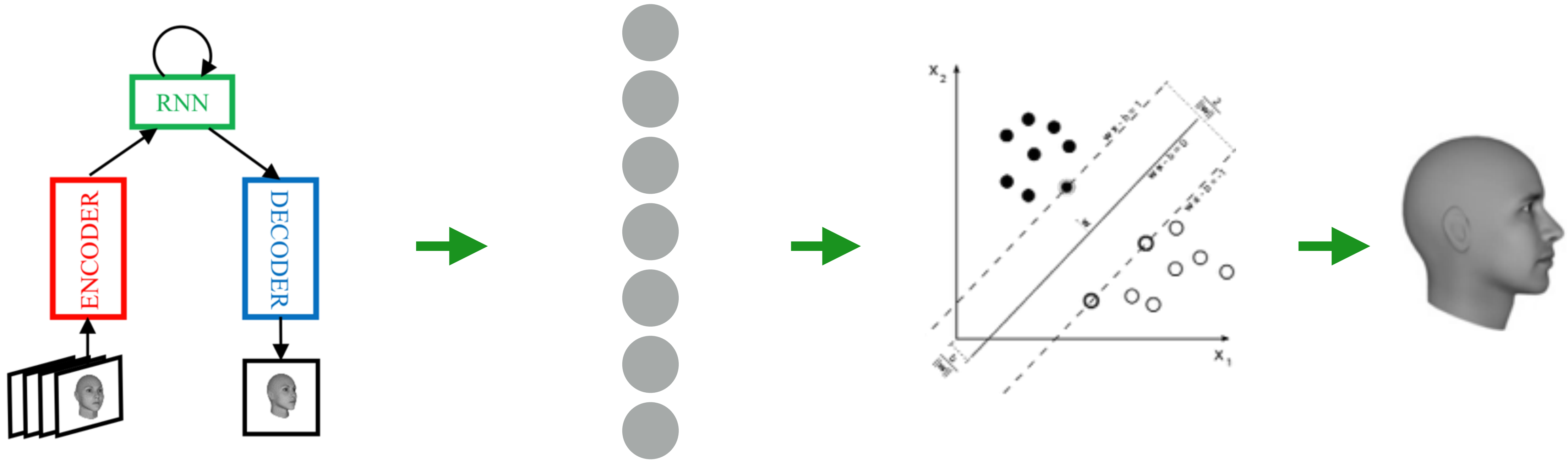
Internal representation



SVM

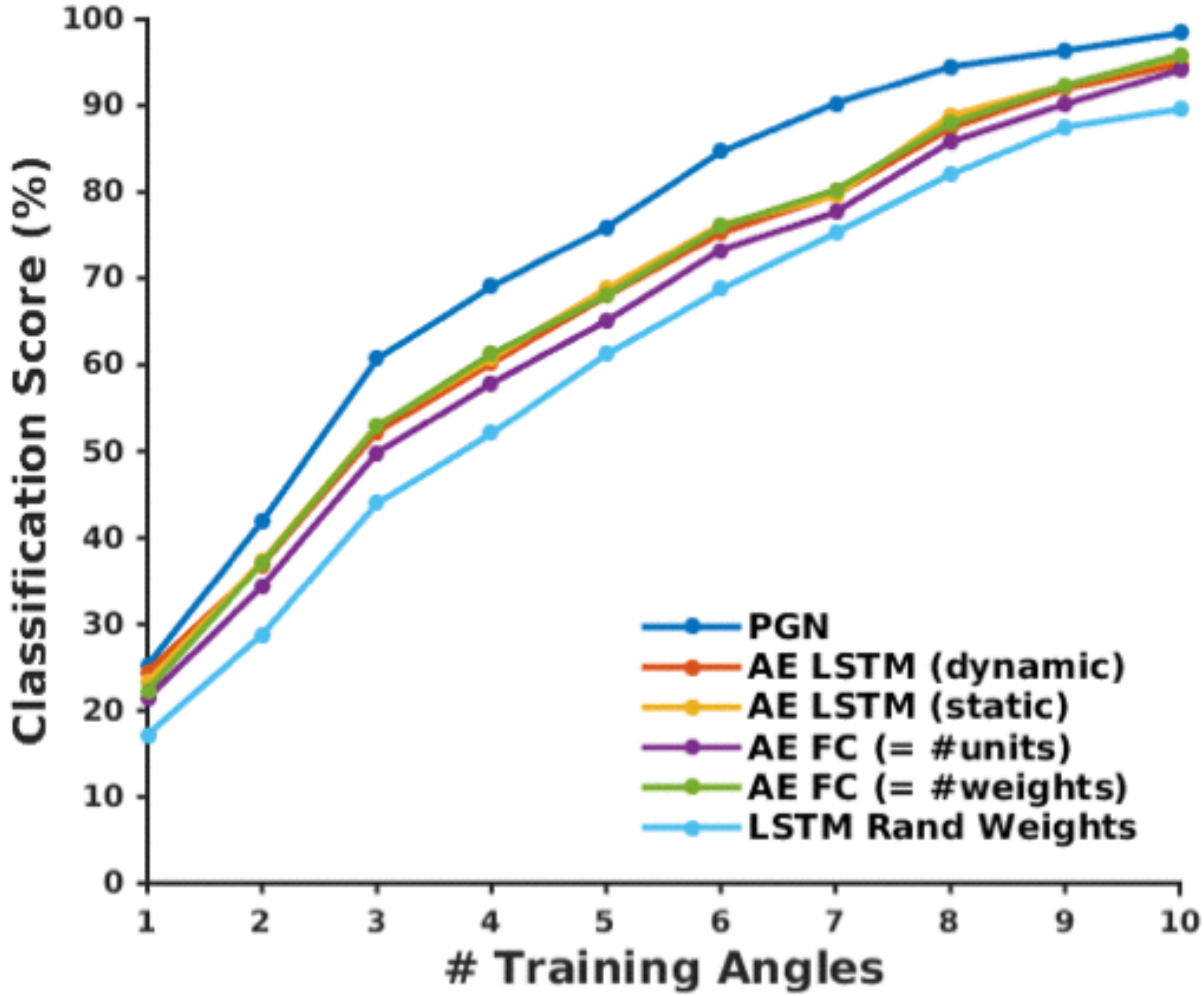
Identify class

The Task: 50 randomly generated faces (12 angles per each)



Generative models: | Internal representation | SVM | Identify class

- Encoder-LSTM-Decoder to predict **next** frame (PGN)
- Encoder-LSTM-Decoder to predict **last** frame (AE LSTM dynamic)
- Encoder-LSTM-Decoder on frames made into **static** movies (AE LSTM static)
- Encoder-FC-Decoder with **#weights** as in LSTM (AE FC #weights)
- Encoder-FC-Decoder with **#units** as in LSTM (AE FC #units)


Classification Score (%)
100
90
80
70
60
50
40
30
20
10
0
1
2
3
4
5
6
7
8
9
10
Training Angles
PGN
AE LSTM (dynamic)
AE LSTM (static)
AE FC (= #units)
AE FC (= #weights)
LSTM Rand Weights