Visual Dynamics: Probabilistic Future Frame Synthesis via Cross Convolutional Networks

Tianfan Xue*  Jiajun Wu*  Katie Bouman  Bill Freeman

NIPS 2016

VGG Reading Group, 24 Feb 2017
Ankush Gupta
Task: future frame prediction
Deterministic predictions fail to model uncertainty
Deterministic predictions fail to model uncertainty

Frame 1 → Deterministic neural network → Prediction

What is the problem?
Deterministic predictions fail to model uncertainty

Frame 1

Deterministic neural network

Prediction

What is the problem?
Sample different future frames

Input frame $\rightarrow$ Synthesis network $\rightarrow$ Sampled future frame

Input random motion vector $z \sim p_z(z)$
Sample different future frames

Input frame \rightarrow \text{Synthesis network} \rightarrow \text{Sampled future frame}

Input random motion vector $z \sim p_z(z)$
Synthesize using different transformations

Input frame \rightarrow Segments \rightarrow Transformed segments \rightarrow Another sampled future frame

Input random motion vector $z \sim p_z(z)$
Training

Input frame → Encoding network → Motion vector z → Synthesis network → Sampled future frame

Future frame (ground truth)
Training

Main idea

Network structure

Outline

What the network learns

Result

Input frame → Encoding network → Motion vector $z$ → Synthesis network → Future frame (prediction)

Future frame (ground truth)

Training samples (Label-free)
Training

Objective function:
\[ \|I_{syn} - I_{gt}\| + D_{KL}(z \| N(0, I)) \]

Reconstruction loss

Future frame \(I_{gt}\) (ground truth)

Input frame

Future frame \(I_{syn}\) (prediction)

Encoding network

Motin vector \(z\)

Synthesis network

Outline

Main idea

Network structure

What the network learns

Result
Training

Objective function:
\[ \|I_{syn} - I_{gt}\| + D_{KL}(z||N(0, I)) \]

KL-divergence loss

Variational Autoencoder
[Kingma and Welling, 2014]
Testing

Future frame $I_{gt}$ (ground truth)

Input frame

Input random motion vector $z \sim p_z(z)$

Encoding network

Future frame $I_{syn}$ (prediction)

Real output from our network
Synthesize by transforming segments

Input random motion vector $z$

Main idea

Network structure

Image segments

Find segments

Transform segments

Convolution

Result

Outline

What the network learns

Future frame

Input frame
Movement can be synthesized through convolution

Outline
Main idea
Network structure
What the network learns
Result
Applying motion to each segment

The decoding network generates a motion kernel for each corresponding segment

[Brabandere et al. 2016]
[Finn et al. 2016]
What is encoded in the motion vector?

Encoding network

Motion vector $z$

Synthesis network

Input frame

Future frame

Input frame

Future frame
Each dimension encodes a type of motion

Motion vector $z$  
Upward motion when changing this dimension
Each dimension encodes a type of motion

Motion vector $z$  
Leg motion when changing this dimension
Results: toy example

- Simulated shapes

- Training samples
Network automatically detects segments

Input

Learned segments

Triangles

Circles

Outline  Main idea  Network structure  What the network learns  Result
Network learns the correlation between appearance and motion

Input

Sampled next frame

Ground truth distribution

Sample distribution

Outline
Main idea
Network structure
What the network learns
Result
Challenge: large motion

Input

Two sampled future frames

Artifacts appear when motion is large
Mechanical Turk study to assess synthesis quality

Labeled as real

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline: Transfer flow</td>
<td>25.5 %</td>
</tr>
<tr>
<td>Our method</td>
<td>31.3 %</td>
</tr>
</tbody>
</table>

Ideal synthesis algorithm achieves 50%
Contributions

• Sample multiple future frames that are consistent with the input

• Synthesize frames by transforming segments

• Learn a motion representation without supervision