Neural Kinematic Networks for Unsupervised Motion Retargetting

Authors: Ruben Villegas, Jimei Yang, Duygu Ceylan, Honglak Lee

Presenter: Muhammad Moiz Sakha

Seminar: Motion Synthesis for Virtual Characters.
Dr. Klaus Fischer

Date: 26/06/2019
Motion Retargeting Problem Statement

Adapt an existing motion to a character with different proportions

Figure: (a) Source motion (b) Retargetted motion

Figure: (a) Source motion (b) Retargetted motion

Source: Shin, Jooh S., et al.
Motivation

Source: Chan, Caroline, et al. [Source: Chan, Caroline, et al.](https://www.youtube.com/watch?v=PCBTZh41Ris)

**Everybody Dance Now, 2018**
Caroline Chan, Shiry Ginosar, Tinghui Zhou, Alexei A. Efros
Motivation -II

Robust Real-time Whole-Body Motion Retargeting from Human to Humanoid, 2018

Motivation - III

Source: https://www.youtube.com/watch?v=HeqPGfrjub8
Why motion retargeting is challenging?

Figure: Imitation Failures (shoulder joint)
Source: Sermanet, Pierre, et al.

Figure: (a) Source motion
(b) Retargetted motion
Source: Shin, Jooh S., et al.

*Time-Contrastive Networks: Self-Supervised Learning from Video, 2017*
Pierre Sermanet, Corey Lynch, Yevgen Chebotar, Jasmine Hsu, Eric Jang, Stefan Schaal, Sergey Levine
Related Work

How to formulate and solve motion retargetting problem?
How to formulate and solve motion retargetting problem?

Gleicher et al. solve spacetime optimization with kinematic constraints for entire motion sequence.

Figure: Differently sized characters pick up an object. Left: original actress. Center: figure 60% as large. Right: figure with extremely short legs & long body
(Source: Gleicher, M., et al.)
Related Work

How to formulate and solve motion retargetting problem?

Lee et al. Maybe, solve IK problem, and then fit a multi-level B-spline for smoother results

Figure: Rope Climbing for different characters. (Source: Lee, J., et al.)
Related Work

How to formulate and solve motion retargetting problem?

Tak and Ko

Sounds good, but we can add dynamic constraints to perform sequential filtering for physically plausible motion

Figure: A dancing motion of the character in the middle is retargeted to left and right.
(Source: Tak, S., et al.)
How to formulate and solve motion retargetting problem?

Gleicher et al: Solve spacetime optimization with kinematic constraints for entire motion sequence.

Lee et al: Maybe, solve IK problem, and then fit a multi-level B-spline for smoother results.

Tak and Ko: Sounds good, but we can add dynamic constraints to perform sequential filtering for physically plausible motion.
### Related Work

**How to formulate and solve motion retargetting problem?**

<table>
<thead>
<tr>
<th>Author(s)</th>
<th>Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gleicher et al</td>
<td>Solve spacetime optimization with kinematic constraints for entire motion sequence</td>
</tr>
<tr>
<td>Lee et al</td>
<td>Maybe, solve IK problem, and then fit a multi-level B-spline for smoother results</td>
</tr>
<tr>
<td>Tak and Ko</td>
<td>Sounds good, but we can add dynamic constraints to perform sequential filtering for physically plausible motion</td>
</tr>
<tr>
<td>Villegas et al</td>
<td>Hmm, all approaches above do iterative optimization with hand-designed kinematic constraints. Why not learn to produce smooth changes of joint angles?</td>
</tr>
</tbody>
</table>
1. A Novel Neural Kinematic Network

Combination of two RNNs and a forward kinematics (FK) layer.
1. **A Novel Neural Kinematic Network**
Combination of two RNNs and a forward kinematics (FK) layer.

2. **Sequence-level adversarial cycle objective function**
A sequence-level adversarial cycle consistency objective function for unsupervised learning for motion retargetting.
Related Work: DL based Motion Synthesis Networks

A Deep Learning Framework for Character Motion Synthesis and Editing

Source: Holden et al., 2016

Auto-Conditioned Recurrent Networks For Extended Complex Human Motion Synthesis

Source: Zhou et al., 2018
Related Work: DL based Motion Synthesis Networks

Fail to preserve target skeleton structure
Refresher - Forward Kinematics

joints rotations & initial position → Forward Kinematics → skeleton joints position
Joint Space → Cartesian Space 3D
Forward kinematics layer

**Forward Kinematics (FK) layer** maps joint rotations to actual joint locations.

- **input**: 3D joint rotation parameterized by unit quaternions $q^n_t \in \mathbb{R}^4$
- **output**: updated joint positions of condition skeleton

Figure: Forward kinematics for T-pose skeleton. (Source: Villegas, Ruben, et al.)
Data-Preprocessing and Skeleton

Separating motion into local and global motion

- Global Motion $v_t$: velocity of root in 3D + rotation around axis perp. to ground.
- Local Motion $p_t$: remove global motion (root joint velocity) from motion.

Figure: T-Pose skeleton with 22 joints.
Source: Villegas, Ruben, et al.

Figure: Illustration of global motion (root joint velocity)
Source: Choi, B., et al.

SketchiMo: sketch-based motion editing for articulated characters, 2016
Neural Kinematics Network (NKN)

Figure: Neural kinematic networks for motion synthesis
Source: Villegas, Ruben, et al.

Notations

\[ x_t = [p_t, v_t], \text{i/p motion sequence} \]
\[ \hat{x}_t = \text{retargetted motion sequence} \]
\[ p_t = \text{local motion 3D (x,y,z)} \]
\[ v_t = \text{global motion (root velocity 3D, rot. around axis perp. to ground)} \]
\[ \hat{q}_t = \text{unit quaternion} \]
\[ \bar{S} = \text{condition or target skeleton} \]
\[ h_t = \text{hidden state RNN} \]
\[ A = \text{superscript A, source} \]
\[ B = \text{superscript B, target} \]
\[ t = \text{subscript t, time-step} \]
\[ ^\wedge = \text{hat symbol for retargetted} \]
\[ h_{t}^{enc} = RNN^{enc}(x_t, h_{t-1}; W^{enc}) \]
\( h_t^{enc} = RNN^{enc}(x_t, h_{t-1}; W^{enc}) \)

\( x_t = [p_t, v_t] \)
\[ h_t^{enc} = RNN^{enc}(x_t, h_{t-1}; W^{enc}) \]

\[ x_t = [p_t, v_t] \]
\[ h_t^{\text{dec}} = RNN^{\text{dec}}(\hat{x}_{t-1}, h_t^{\text{enc}}, \hat{s}, h_t^{\text{dec}}, W^{\text{dec}}) \]

\[ h_t^{\text{enc}} = RNN^{\text{enc}}(x_t, h_{t-1}; W^{\text{enc}}) \]
\( \hat{x}_{t-1} = [\hat{p}_{t-1}, \hat{v}_{t-1}] \)

\( h_t^{dec} = RNN^{dec}(\hat{x}_{t-1}, h_t^{enc}, \bar{s}, h_{t-1}^{dec}, W^{dec}) \)

\( h_t^{enc} = RNN^{enc}(x_t, h_{t-1}; W^{enc}) \)

\( x_t = [p_t, v_t] \)
\[ h_t^{\text{dec}} = \text{RNN}^{\text{dec}}(\hat{x}_{t-1}, h_t^{\text{enc}}, \tilde{s}, h_{t-1}^{\text{dec}}, W^{\text{dec}}) \]

\[ h_t^{\text{enc}} = \text{RNN}^{\text{enc}}(x_t, h_{t-1}; W^{\text{enc}}) \]
\begin{align*}
    h_{t}^{\text{dec}} &= \text{RNN}^{\text{dec}}(\hat{x}_{t-1}, h_{t}^{\text{enc}}, \bar{s}, h_{t-1}^{\text{dec}}, W^{\text{dec}}) \\
    h_{t}^{\text{enc}} &= \text{RNN}^{\text{enc}}(x_{t}, h_{t-1}; W^{\text{enc}})
\end{align*}
\( h_t^{\text{dec}} = \text{RNN}^{\text{dec}}(\hat{x}_{t-1}, h_{t-1}^{\text{enc}}, \tilde{s}, h_t^{\text{dec}}, W^{\text{dec}}) \)

\( h_t^{\text{enc}} = \text{RNN}^{\text{enc}}(x_t, h_{t-1}; W^{\text{enc}}) \)

\( x_t = [p_t, v_t] \)
\[ \hat{p}_t = FK(\hat{q}_t, \bar{s}) \]

\[ h_{t}^{\text{dec}} = RNN^{\text{dec}}(\hat{x}_{t-1}, h_{t}^{\text{enc}}, \bar{s}, h_{t-1}^{\text{dec}}, W^{\text{dec}}) \]

\[ h_{t}^{\text{enc}} = RNN^{\text{enc}}(x_t, h_{t-1}; W^{\text{enc}}) \]

\[ x_t = [p_t, v_t] \]
\[ \hat{p}_t = FK(\tilde{q}_t, \tilde{s}) \]

\[ h_{t}^{\text{dec}} = \text{RNN}^{\text{dec}}(\tilde{x}_{t-1}, h_{t}^{\text{enc}}, \tilde{s}, h_{t-1}^{\text{dec}}, W^{\text{dec}}) \]

\[ h_{t}^{\text{enc}} = \text{RNN}^{\text{enc}}(x_{t}, h_{t-1}; W^{\text{enc}}) \]

\[ x_t = [p_t, v_t] \]
\( \hat{p}_t = FK (\hat{q}_t, \tilde{s}) \)

\[
\begin{align*}
\hat{h}_t^{dec} &= RNN^{dec}(\hat{x}_{t-1}, \hat{h}_t^{enc}, \tilde{s}, \tilde{h}_{t-1}^{dec}, W^{dec}) \\
\hat{h}_t^{enc} &= RNN^{enc}(x_t, h_{t-1}; W^{enc}) \\
x_t &= [p_t, v_t]
\end{align*}
\]
\[ h_t^{enc} = RNN^{enc}(x_t, h_{t-1}; W^{enc}) \]

\[ x_t = [p_t, v_t] \]
Cycle Consistency, Cycle GAN Principle

- Acquiring paired training input/output is expensive

Jun-Yan Zhu et al. Cycle GAN, 2017

Slide adapted from: https://drive.google.com/file/d/1ITi1ay19_FKPpmhztGosPqYXgoaw5-xj/view, Author: Ruben Villegas
Cycle Consistency, Cycle GAN Principle

- Acquiring paired training input/output is expensive

Jun-Yan Zhu et al. Cycle GAN, 2017

Slide adapted from: https://drive.google.com/file/d/1ITi1ay19_FKPpmhztGosPqXgoaw5-xj/view, Author: Ruben Villegas
Method: Adversarial Cycle Consistency Training

\[
\min_f \max_d \left( C(\hat{x}_1^A, x_1^A) + R(\hat{x}_1^B, x_1^A) + \lambda J(\hat{q}_1^B, \hat{q}_1^A) + \omega S(\hat{\dot{q}}_1^B, \hat{\dot{q}}_1^A) \right)
\]

- cycle consistency loss

\(C\) - the adversarial loss
\(R\) - the joint twist loss
\(J\) - the velocity smoothing loss
\(S\) - the velocity smoothing loss
Method: Adversarial Cycle Consistency Training

Cycle Consistency loss

- map own retargetted motion back to the original motion

\[ C(\hat{x}_{1:T}^A, x_{1:T}^A) = \| x_{1:T}^A - \hat{x}_{1:T}^A \|_2^2 \]
Cycle Consistency Loss

\[ x^A_{1:T} = \text{i/p motion sequence of } A \]
\[ \hat{x}^A_{1:T} = \text{motion retargetted back to } A \]

Figure: Illustration of cycle consistency loss
Cycle Consistency Loss

Figure: Illustration of cycle consistency loss

\[ x^A_{1:T} = \text{i/p motion sequence of A} \]
\[ \hat{x}^A_{1:T} = \text{motion retargetted back to A} \]
Cycle Consistency Loss

\[ x_{1:T}^A = \text{i/p motion sequence of A} \]
\[ \hat{x}_{1:T}^A = \text{motion retargetted back to A} \]

Figure: Illustration of cycle consistency loss
Method: Adversarial Cycle Consistency Training

\[ \min_f \max_d \left( C(\hat{x}^A_{1:T}, x^A_{1:T}) + R(\hat{x}^B_{1:T}, x^A_{1:T}) + \lambda J(\hat{\eta}^B_{1:T}, \hat{\eta}^A_{1:T}) + \omega S(\hat{\eta}^B_{1:T}, \hat{\eta}^A_{1:T}) \right) \]

- cycle consistency loss
- the adversarial loss
Method: Adversarial Cycle Consistency Training

Adversarial Loss

- Retargetted motion is passed to discriminator to judge realism

\[
R(\hat{x}_1^B, x_1^A) = \begin{cases} 
\|\hat{x}_1^B - x_1^A\|^2, & \text{if } B = A \\
\log r^A + \beta \log(1 - r^B), & \text{otherwise.}
\end{cases}
\]
Adversarial Training

Generator Network

Generated motion sequence
Adversarial Training

Real motion sequence

Generated motion sequence

Generator Network

Discriminator Network

Realism Score
Adversarial Training

Real motion sequence

$r^A = g(p^A_{2:T} - p^A_{1:T-1}, v^A_{1:T-1}, \bar{s}^A)$

Generated motion sequence

Realism score for generated/fake motion sequence

$r^B = g(\hat{p}^B_{2:T} - \hat{p}^B_{1:T-1}, \hat{v}^B_{1:T-1}, \bar{s}^B)$

Realism score for real motion sequence

Generator Network

Discriminator Network
Method: Adversarial Cycle Consistency Training

\[
\min_f \max_d \quad C(\hat{x}_A^{1:T}, x_A^{1:T}) + R(\hat{x}_B^{1:T}, x_A^{1:T}) + \lambda \ J(\hat{q}_B^{1:T}, \hat{q}_A^{1:T}) + \omega \ S(\hat{\theta}_B^{1:T}, \hat{\theta}_A^{1:T})
\]

- cycle consistency loss
- the adversarial loss
- the joint twist loss
Method: Adversarial Cycle Consistency Training

Twist Loss

- to prevent excessive bone twisting

\[ J(\hat{q}_1^B, \hat{q}_1^A) = \sum_{i=A,B} (\max(0, |\text{euler}(\hat{q}_1^i)| - \alpha))^2 \]
Twist Loss

Figure: Illustration of why twist loss helps

Retargeted Motion with front facing head
Skeleton is same for left & right retargeted motion
Retargeted motion with back facing head

https://www.mixamo.com/#/?page=1&query=remy&type=Character
Method: Adversarial Cycle Consistency Training

\[
\min_f \max_d C(\hat{x}_{1:T}^A, x_{1:T}^A) + R(\hat{x}_{1:T}^B, x_{1:T}^A) + \lambda J(\hat{q}_{1:T}^B, \hat{q}_{1:T}^A) + \omega S(\hat{v}_{1:T}^B, \hat{v}_{1:T}^A)
\]

- cycle consistency loss
- the adversarial loss
- the joint twist loss
- the velocity smoothing loss
Method: Adversarial Cycle Consistency Training

**Smoothing Loss**

- to ensure global motion change smoothly through time

\[
S(\hat{v}^B_{1:T}, \hat{v}^A_{1:T}) = \sum_{i=A,B} (\hat{v}^i_{2:T} - \hat{v}^i_{1:T-1})^2
\]
Smoothness Loss

\[ \hat{\mathbf{v}}_{2:T}^B - \hat{\mathbf{v}}_{1:T-1}^B \]

Difference in velocities of root joint of retargeted motion to B

\[ \hat{\mathbf{v}}_{2:T}^A + \hat{\mathbf{v}}_{1:T-1}^A \]

Difference in velocities of root joint of motion retargeted back to A

Figure: Smoothness loss penalizes large global motion difference
**Dataset (Mixamo)**

**Training dataset:** 7 different characters.

**Test dataset:** 6 different characters (3 unseen).

https://www.mixamo.com/
MSE computed b/w retargetted sequences & ground truth in world 3D coordinates.

<table>
<thead>
<tr>
<th>Method</th>
<th>MSE*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours: Autoencoder Objective</td>
<td>10.25</td>
</tr>
<tr>
<td>Ours: Cycle Consistency Objective</td>
<td>8.51</td>
</tr>
<tr>
<td>Ours: Adversarial Cycle Consistency Objective</td>
<td>7.10</td>
</tr>
<tr>
<td>Baseline: Conditional RNN</td>
<td>13.65</td>
</tr>
<tr>
<td>Baseline: Conditional MLP</td>
<td>17.02</td>
</tr>
<tr>
<td>Baseline: Copy input quaternions and velocities</td>
<td>9.00</td>
</tr>
</tbody>
</table>

Table: Quantitative evaluation of online motion retargetting. (Source: Villegas, Ruben, et al.)

*Mean Square Error
Experiments

https://sites.google.com/umich.edu/nik
Figure: Qualitative evaluation. (Source: Villegas, Ruben, et al.)
https://sites.google.com/umich.edu/nik
Limitations and Future Work

Limitations

- Retargetting on a fixed number of joints
- Assumes target environment lacks physical constraints such as gravity
- Input still requires 3D information :

Future Work

- Equip n/w with physics simulators for more realistic motion
- Train n/w using 2D monocular videos
Limitations and Future Work

Limitations

- Retargetting on a fixed number of joints
- Assumes target environment lacks physical constraints such as gravity
- Input still requires 3D information :(

Future Work

- Equip n/w with physics simulators for more realistic motion
- Train n/w using 2D monocular videos
Limitations and Future Work

Limitations

- Retargetting on a fixed number of joints
- Assumes target environment lacks physical constraints such as gravity
- Input still requires 3D information :( 

Future Work

- Equip n/w with physics simulators for more realistic motion
- Train n/w using 2D monocular videos
Limitations and Future Work

Limitations

- Retargetting on a fixed number of joints
- Assumes target environment lacks physical constraints such as gravity
- Input still requires 3D information :( 

Future Work

- Equip n/w with physics simulators for more realistic motion
- Train n/w using 2D monocular videos
Limitations and Future Work

Limitations

- Retargetting on a fixed number of joints
- Assumes target environment lacks physical constraints such as gravity
- Input still requires 3D information :(

Future Work

- Equip n/w with physics simulators for more realistic motion
- Train n/w using 2D monocular videos
Learning Character-Agnostic Motion for Motion Retargeting in 2D

KFIR ABERMAN, Tel-Aviv University, AICFVE Beijing Film Academy
RUNDI WU, Peking University
DANI LISCHINSKI, Shandong University, Hebrew University of Jerusalem
BAOQUAN CHEN*, Peking University
DANIEL COHEN-OR, Tel-Aviv University

Analyzing human motion is a challenging task with a wide variety of applications in computer vision and in graphics. One such application, of particular importance in computer animation, is the retargeting of motion from one performer to another. While humans move in three dimensions, the vast majority of human motions are captured using video, requiring 2D-to-3D pose and camera recovery, before existing retargeting approaches may be applied. In this paper, we present a new method for retargeting video-captured motion between different human performers, without the need to explicitly reconstruct 3D poses and/or camera parameters.

In order to achieve our goal, we learn to extract, directly from a video, a high-level latent motion representation, which is invariant to the skeleton geometry and the camera view. Our key idea is to train a deep neural network to decompose temporal sequences of 2D poses into three components: motion, skeleton, and camera view-angle. Having extracted such a representation, we are able to re-combine motion with novel skeletons and camera views, and decode a retargeted temporal sequence, which we compare to a ground truth from a synthetic dataset.

We demonstrate that our framework can be used to robustly extract human motion from videos, bypassing 3D reconstruction, and outperforming existing retargeting methods, when applied to videos in-the-wild. It also enables additional applications, such as performance cloning, video-driven cartoons, and motion retrieval.

Webpage (code and data): https://motionretargeting2d.github.io/

CCS Concepts: • Computing methodologies → Motion processing; Neural networks.

Additional Key Words and Phrases: Motion retargeting, autoencoder, motion

Fig. 1. Given two videos of different performers, our approach enables to extract character-agnostic motion from each video, and transfer it to a new skeleton and view angle (top-left and bottom-right), directly in 2D. In addition, separate latent representations for motion, skeleton, and view-angle are extracted, enabling control and interpolation of these parameters.
Latest in Motion Retargeting -II

Latest in Motion Retargeting -III

Application: Motion Retrieval, search in dataset of videos in the wild for similar motion

Source: Aberman, Kfir, et al. [https://www.youtube.com/watch?v=fR4h4OjZSdU](https://www.youtube.com/watch?v=fR4h4OjZSdU)
**Conclusion**

**Papers Discussed**

- **Neural Kinematic Networks for Unsupervised Motion Retargeting**

  - **Limitations**
    - Retargetting on fixed no. of joints
    - Requires 3D i/p
    - Lacks physical constraints i.e. gravity
  
  - **Future Work**
    - Equip with physics simulator
    - Train on monocular videos

- **Learning Character-Agnostic Motion for Motion Retargeting in 2D**

  - **Limitations**
    - fails to transfer large scale or view angle variation
  
  - **Future Work**
    - Use it to assist reconstruction of 3D skeleton from video

**Contributions Summarized**

- Forward Kinematics layer: helps discover motion features through rotation.

- Cycle consistent adversarial training: enables unsupervised motion retargeting to unseen skeleton.

- Extract character, camera-agnostic, latent representation of human motion directly from ordinary video and apply to different skeleton from another video.
References


