Interpretable 3D Human Action Analysis with Temporal Convolutional Networks

Tae Soo Kim and Austin Reiter
Johns Hopkins University
July 21, 2017

CVPR 2017: Brave New Ideas for Motion Representations in Videos 2 Workshop
3D Human Activity Recognition

- Sequence of 3D human skeletons to action class label

\[ Y = \{\text{Basketball, Kicking, Hugging, Jumping, Running, Walking} \ldots\} \]
Why 3D Human Activity Recognition?

• Development of better/cheaper depth sensors
Why 3D Human Activity Recognition?

• Development of better/cheaper depth sensors

• Preserves 3D structure
Why 3D Human Activity Recognition?

• Development of better/cheaper depth sensors

• Preserves 3D structure

• Eliminates context
Why 3D Human Activity Recognition?

• Development of better/cheaper depth sensors

• Preserves 3D structure

• Eliminates context

• Available stronger human pose estimators
Why 3D Human Activity Recognition?

• Development of better/cheaper depth sensors

• Preserves 3D structure

• Eliminates context

• Available stronger human pose estimators

• Interpretable input
State-of-the-art Methods

• Recurrent Neural Networks with LSTM neurons

“Semantic joint groups”

Part-aware LSTM[26]
State-of-the-art Methods

- Recurrent Neural Networks with LSTM neurons

- Spatial Knowledge
- Temporal Knowledge

“Semantic joint groups”
Part-aware LSTM[26]

“Joint traversal LSTM”
ST-LSTM [36]
State-of-the-art Methods

• Recurrent Neural Networks with LSTM neurons

- "Spatial-Attention LSTM + Temporal-Attention LSTM"
  - STA-LSTM [29]

- "Semantic joint groups"
  - Part-aware LSTM [26]

- "Joint traversal LSTM"
  - ST-LSTM [36]
Where is the Gap?

• Trend towards *hand-crafted* models
  • More prior knowledge injection through custom LSTM gating mechanisms

• Current methods still remain mostly *black-boxes.*
Where is the Gap?

• Trend towards *hand-crafted* models
  • More prior knowledge injection through custom LSTM gating mechanisms
  • *Can we get a simpler and easier-to-train models?*

• Current methods still remain mostly *black-boxes.*
Where is the Gap?

• Trend towards *hand-crafted* models
  • More prior knowledge injection through custom LSTM gating mechanisms
  • *Can we get a simpler and easier-to-train models?*

• Current methods still remain mostly *black-boxes.*
  • *Can we design a model that is readily interpretable?*
Starting Point: Temporal Convolutional Networks (TCN)

• Activity Segmentation Model
• Input
  • sequence of frame-level features
• Design
  • Stacked blocks of 1D convolution layers followed by non-linear activations (ReLU)
  • Convolutional encoder-decoder design
• Output
  • Frame-wise classification

Original TCN [19]
Interpretability of TCNs

• First convolution layer is **directly interpretable**.
  • Given that every dimension of the input is directly interpretable as well.
  • “Parameter interpretability”

• It is a **ReLU network**
  • Filters are optimized such that *discriminative* parts of the input produces a positive response.
  • “Hidden representation interpretability”
First convolution layer is **directly interpretable.**
Model Design Strategy

• First convolution layer is directly interpretable.
  1. Lets make all deeper layers learn to either add to or subtract from this representation.

• It is a ReLU network
  2. Lets make all deeper layers to reason over only the positive regions.
TCN + Residual Units and Identity Mappings [11]

• Output of first conv layer

\[ X_1 = W_1 \ast X_0 \]

• Lets make all deeper layers learn to either add to or subtract from this representation

\[ X_2 = X_1 + F(W_2, X_1) \]

where \( F(W_2, X_1) = W_2 \ast \text{max}(0, X_1) \)

then

\[ X_N = X_1 + \sum_{i=2}^{N} F(W_i, X_{i-1}) \]
TCN with Residual Units and Identity Mappings

• Output of first conv layer

\[ X_1 = W_1 \ast X_0 \]

• Lets make all deeper layers learn to either add to or subtract from this representation

\[ X_2 = X_1 + F(W_2, X_1) \]

where

\[ F(W_2, X_1) = W_2 \ast \max(0, X_1) \]

then

\[ X_N = X_1 + \sum_{i=2}^{N} F(W_i, X_{i-1}) \]

"Residual Unit"

"Pre-activation"

Only reason over the positive region
Res-TCN

*: downsample with stride 2
\( f \): conv filter length

"Conv layer with 64 filters of length 8"
NTURGBD [26] Experiments

• Currently the largest 3D skeleton dataset
  • 60 activity classes

• Implementation details
  • No skeleton normalization: raw input
  • Keras + Tensorflow
  • See https://github.com/TaeSoo-Kim/TCNAActionRecognition for implementation and training parameters.
    • Note: Keras and NTU have been updated since the code release (Code was released April 2017).
Model Input

- $x(t) \in R^D$
- $X_0 \in R^{T \times D}$

$$x(t) = [1^a_x, 1^a_y, 1^a_z, \ldots, 25^a_x, 25^a_y, 25^a_z, 1^b_x, 1^b_y, 1^b_z, \ldots, 25^b_x, 25^b_y, 25^b_z]$$

$D = 25$ joints * at most 2 actors * 3 dim (xyz) = 150
How Did My Model Come to this Conclusion?

“Kicking”
How Did My Model Come to this Conclusion?

• Example hidden representation for input with label = ‘kicking’
- **Green filter**: \{left ankle, left hip\}
- **Yellow filter**: \{right knee\}
- **Blue filter**: \{right ankle, left wrist\}
the left ankle and hip joints first translate followed by a sudden movement of the right knee, all the while the left wrist and the right ankle undergo a swinging motion.

- Green filter: \{left ankle, left hip\}
- Yellow filter: \{right knee\}
- Blue filter: \{right ankle, left wrist\}
Spatio-Temporal Attention

• \{left ankle, left hip\} activation close to 0 at climax of ‘kicking’ motion.
• Filters corresponding to the kicking leg peaks at the climax.
Comparison to other State-of-the-art

- CS: Cross-Subject validation
- CV: Cross-View validation

- Model design with interpretability as target but competitive recognition performance as well.
  - A simpler model that can be more easily optimized
  - Better optimization leads to a deeper model: stronger representation power

<table>
<thead>
<tr>
<th>Methods</th>
<th>CS</th>
<th>CV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dynamic Skeletons [15]</td>
<td>60.2</td>
<td>65.2</td>
</tr>
<tr>
<td>HBRNN [5]</td>
<td>59.1</td>
<td>64.0</td>
</tr>
<tr>
<td>Deep LSTM [26]</td>
<td>60.7</td>
<td>67.3</td>
</tr>
<tr>
<td>P-LSTM [26]</td>
<td>62.9</td>
<td>70.3</td>
</tr>
<tr>
<td>Trust Gate [22]</td>
<td>69.2</td>
<td>77.7</td>
</tr>
<tr>
<td>STA-LSTM [29]</td>
<td>73.4</td>
<td>81.2</td>
</tr>
<tr>
<td>Res-TCN</td>
<td>74.3</td>
<td>83.1</td>
</tr>
</tbody>
</table>

At time of submission (April 2017)
Visualizations

• Directly plot conv1 filters.
Visualizations

• View filters “velocities”
  • ie. original_joint_pos + original_joint_pos*filter_params
Conclusion

• A temporal convolutional neural network approach for 3D human activity recognition

• A model design that explicitly yields both interpretable model parameters and hidden representations.

• A step towards non-black-box models

• Future work:
  • now that we understand that these conv filters learn snippets of motion, can we design a model with temporal scale invariance?
  • Can we formalize this discussion and ground the math behind this intuition?
Thank you!
Questions?

Code can be found here: https://github.com/TaeSoo-Kim/TCNActionRecognition