An Introduction to Temporal Action Segmentation
From Fully Supervised Learning to Weakly Supervised Learning
Action Recognition

- Large annotated datasets

- UCF101 (98.2%), HMDB (82.5%), Kinetics-400 (82.8%), Epic-Kitchens (36.7%)
- [http://actionrecognition.net](http://actionrecognition.net)
- Continuous data streams
Action Segmentation vs. Action Detection

- Action Detection (THUMOS, ActivityNet)

Ground Truth:

- Action Segmentation (Breakfast, 50 Salads, GTEA)
Action Segmentation vs. Action Detection

- Action Detection (Object Detection)

- Action Segmentation (Semantic Segmentation)
Action Segmentation vs. Action Detection

- Action Detection (THUMOS, ActivityNet)

- Action Segmentation (Breakfast, 50Salads, GTEA)
Why Action Segmentation?
Datasets

• Breakfast
  https://serre-lab.clps.brown.edu/resource/breakfast-actions-dataset/

• 50 Salads
  https://cvip.computing.dundee.ac.uk/datasets/foodpreparation/50salads/

• GTEA
  http://cbs.ic.gatech.edu/fpv/#gtea

• COIN
  https://coin-dataset.github.io/
Let’s build a baseline…
Hidden Markov Model

\[ Pr(x_1...N, w_1...N) = \left( \prod_{n=1}^{N} Pr(x_n|w_n) \right) \left( \prod_{n=2}^{N} Pr(w_n|w_{n-1}) \right) \]

Inference

HMM:

\[
Pr(x_{1...N}, w_{1...N}) = \left( \prod_{n=1}^{N} Pr(x_n | w_n) \right) \left( \prod_{n=2}^{N} Pr(w_n | w_{n-1}) \right)
\]

MAP inference:

\[
\hat{w}_{1...N} = \arg\max_{w_{1...N}} [Pr(x_{1...N}, w_{1...N})]
\]

\[
= \arg\min_{w_{1...N}} [-\log Pr(x_{1...N}, w_{1...N})]
\]

Substituting:

\[
\hat{w}_{1...N} = \arg\min_{w_{1...N}} \left[ - \sum_{n=1}^{N} \log Pr(x_n | w_n) - \sum_{n=2}^{N} \log Pr(w_n | w_{n-1}) \right]
\]

Global minimum by dynamic programming

Features: Dense Trajectories

- Dense sampling of features
- Feature tracking

Hidden Markov Model

- Hidden Markov Model (HMM) for each activity

[ H. Kuehne et al. An end-to-end generative framework for video segmentation and recognition. WACV 2016 ]
Baseline

- HMM + GMM (IDT)

[ H. Kuehne et al. An end-to-end generative framework for video segmentation and recognition. WACV 2016 ]
Baseline

- HMM + GMM (IDT)

[H. Kuehne et al. An end-to-end generative framework for video segmentation and recognition. WACV 2016]
Baseline

- HMM + GMM (IDT)

[H. Kuehne et al. *An end-to-end generative framework for video segmentation and recognition. WACV 2016*]
Grammar

• Transitions between activity HMMs are modeled by context free grammar

• SIL: start and end points
• Transition probability is 1 if connection exists otherwise 0

[H. Kuehne et al. An end-to-end generative framework for video segmentation and recognition. WACV 2016]
Baseline

- Breakfast dataset (~65 hours)

<table>
<thead>
<tr>
<th>Method</th>
<th>Frame-wise Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kuehne et al. 2016 (HMM+GMM)</td>
<td>56.3</td>
</tr>
</tbody>
</table>

[H. Kuehne et al. An end-to-end generative framework for video segmentation and recognition. WACV 2016]
Hybrid RNN-HMM

- HMM + RNN with Gated Recurrent Units (GRU)

$$p(x_t | s) = \text{const} \cdot \frac{p(s | x_t)}{p(s)}$$
Gated Recurrent Units (GRU)

- Similar to LSTM, but it does not need an additional memory cell

\[
\begin{align*}
    z_t &= \sigma \left( W_z \cdot [h_{t-1}, x_t] \right) \\
    r_t &= \sigma \left( W_r \cdot [h_{t-1}, x_t] \right) \\
    \tilde{h}_t &= \tanh \left( W \cdot [r_t \ast h_{t-1}, x_t] \right) \\
    h_t &= (1 - z_t) \ast h_{t-1} + z_t \ast \tilde{h}_t
\end{align*}
\]

Hybrid RNN-HMM

- Breakfast dataset (~65 hours)

<table>
<thead>
<tr>
<th>Method</th>
<th>Frame-wise Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kuehne et al. 2016 (HMM+GMM)</td>
<td>56.3</td>
</tr>
<tr>
<td>Richard et al. 2017 (HMM+RNN)</td>
<td>60.6</td>
</tr>
<tr>
<td>Kuehne et al. 2020 (HMM+RNN)</td>
<td>61.3</td>
</tr>
</tbody>
</table>

Temporal Convolutional Neural Network

[ C. Lea et al. Temporal Convolutional Networks for Action Segmentation and Detection. CVPR 2017 ]
Temporal Convolutional Network

- Breakfast dataset (~65 hours)

<table>
<thead>
<tr>
<th>Method</th>
<th>Frame-wise Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lea et al. 2017 (ED-TCN)*</td>
<td>43.3</td>
</tr>
<tr>
<td>Kuehne et al. 2016 (HMM+GMM)</td>
<td>56.3</td>
</tr>
<tr>
<td>Richard et al. 2017 (HMM+RNN)</td>
<td>60.6</td>
</tr>
<tr>
<td>Kuehne et al. 2020 (HMM+RNN)</td>
<td>61.3</td>
</tr>
</tbody>
</table>

*[L. Ding and C. Xu. Weakly-supervised action segmentation with iterative soft boundary assignment. CVPR 2018 ]

[ C. Lea et al. Temporal Convolutional Networks for Action Segmentation and Detection. CVPR 2017 ]
Temporal Convolutional Neural Network

- Dilated convolutions for audio

Temporal Convolutional Neural Network

Dilated convolutions capture long temporal receptive field

Causal convolutions: Input for t depends only on previous observations

[ C. Lea et al. Temporal Convolutional Networks for Action Segmentation and Detection. CVPR 2017 ]
## Temporal Convolutional Network

- 50 Salads

<table>
<thead>
<tr>
<th>Method</th>
<th>Frame-wise Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lea et al. 2017 (ED-TCN)</td>
<td>64.7</td>
</tr>
<tr>
<td>Lea et al. 2017 (Dilated TCN)</td>
<td>59.3</td>
</tr>
</tbody>
</table>

[C. Lea et al. Temporal Convolutional Networks for Action Segmentation and Detection. CVPR 2017]
Temporal Convolutional Network

- **50 Salads**

<table>
<thead>
<tr>
<th>Method</th>
<th>1 – Norm. Edit Distance (%)</th>
<th>Frame-wise Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lea et al. 2017 (ED-TCN)</td>
<td>59.8</td>
<td>64.7</td>
</tr>
<tr>
<td>Lea et al. 2017 (Dilated TCN)</td>
<td>43.1</td>
<td>59.3</td>
</tr>
</tbody>
</table>

- **Edit distance (sensitive to oversegmentation):**

  * B I O G R A P H Y
  A U T O G R A P H
  i s s
  d
Multi-Stage Temporal Convolutional Network

[ Y. Abu Farha and J. Gall. MS-TCN: Multi-Stage Temporal Convolutional Network for Action Segmentation. CVPR 2019 ]
Multi-Stage Temporal Convolutional Network

[ Y. Abu Farha and J. Gall. MS-TCN: Multi-Stage Temporal Convolutional Network for Action Segmentation. CVPR 2019 ]
Multi-Stage Temporal Convolutional Network

[ Y. Abu Farha and J. Gall. MS-TCN: Multi-Stage Temporal Convolutional Network for Action Segmentation. CVPR 2019 ]
Over-segmentation

- Frame-wise classification loss:

\[
\mathcal{L}_{cls} = \frac{1}{T} \sum_t -\log(y_{t,c})
\]

- Additional loss is required to avoid over-segmentation:

[ Y. Abu Farha and J. Gall. **MS-TCN: Multi-Stage Temporal Convolutional Network for Action Segmentation.** CVPR 2019 ]
Loss

- Frame-wise classification loss $\mathcal{L}_{cls}$
- Additional loss is required to avoid over-segmentation:

$$\mathcal{L}_{T-MSE} = \frac{1}{TC} \sum_{t,c} \tilde{\Delta}_{t,c}^2$$

$$\tilde{\Delta}_{t,c} = \begin{cases} \Delta_{t,c} : \Delta_{t,c} \leq \tau \\ \tau : \text{otherwise} \end{cases}$$

$$\Delta_{t,c} = |\log y_{t,c} - \log y_{t-1,c}|$$

- Loss functions of all stages $s$:

$$\mathcal{L} = \sum_{s} \mathcal{L}_s \quad \mathcal{L}_s = \mathcal{L}_{cls} + \lambda \mathcal{L}_{T-MSE}$$

[ Y. Abu Farha and J. Gall. MS-TCN: Multi-Stage Temporal Convolutional Network for Action Segmentation. CVPR 2019 ]
Loss

- Additional loss is required to avoid oversegmentation

<table>
<thead>
<tr>
<th></th>
<th>Edit</th>
<th>Acc</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathcal{L}_{cls}$</td>
<td>64.2</td>
<td>79.9</td>
</tr>
<tr>
<td>$\mathcal{L}<em>{cls} + \lambda \mathcal{L}</em>{T-MSE}$</td>
<td>67.9</td>
<td>80.7</td>
</tr>
</tbody>
</table>
Impact of stages

- Impact of stages (50 Salads)

<table>
<thead>
<tr>
<th></th>
<th>Edit</th>
<th>Acc</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS-TCN</td>
<td>20.5</td>
<td>78.2</td>
</tr>
<tr>
<td>MS-TCN (2 stages)</td>
<td>47.9</td>
<td>79.8</td>
</tr>
<tr>
<td>MS-TCN (3 stages)</td>
<td>64.0</td>
<td>78.6</td>
</tr>
<tr>
<td>MS-TCN (4 stages)</td>
<td>67.9</td>
<td>80.7</td>
</tr>
<tr>
<td>MS-TCN (5 stages)</td>
<td>69.2</td>
<td>79.5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Edit</th>
<th>Acc</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS-TCN (48 layers)</td>
<td>40.7</td>
<td>78.0</td>
</tr>
<tr>
<td>MS-TCN</td>
<td>67.9</td>
<td>80.7</td>
</tr>
</tbody>
</table>

[ Y. Abu Farha and J. Gall. MS-TCN: Multi-Stage Temporal Convolutional Network for Action Segmentation. CVPR 2019 ]
Multi-Stage Temporal Convolutional Network

- Breakfast dataset (~65 hours)

<table>
<thead>
<tr>
<th>Method</th>
<th>Frame-wise Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lea et al. 2017 (ED-TCN)*</td>
<td>43.3</td>
</tr>
<tr>
<td>Kuehne et al. 2016 (HMM+GMM)</td>
<td>56.3</td>
</tr>
<tr>
<td>Richard et al. 2017 (HMM+RNN)</td>
<td>60.6</td>
</tr>
<tr>
<td>Kuehne et al. 2020 (HMM+RNN)</td>
<td>61.3</td>
</tr>
<tr>
<td>MS-TCN (TCN)</td>
<td>65.1</td>
</tr>
<tr>
<td>MS-TCN (TCN+I3D)</td>
<td>66.3</td>
</tr>
</tbody>
</table>

[ Y. Abu Farha and J. Gall. MS-TCN: Multi-Stage Temporal Convolutional Network for Action Segmentation. CVPR 2019 ]
Temporal Action Segmentation

(prediction: background
correct: background

prediction
gr. truth)
MS-TCN

Y. Abu Farha and J. Gall. MS-TCN: Multi-Stage Temporal Convolutional Network for Action Segmentation CVPR 2019
MS-TCN++

[ S. Li et al. MS-TCN++: Multi-Stage Temporal Convolutional Network for Action Segmentation. arXiv ]
### MS-TCN++

- **Breakfast dataset**

<table>
<thead>
<tr>
<th>Method</th>
<th>Frame-wise Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lea et al. 2017 (TCN)*</td>
<td>43.3</td>
</tr>
<tr>
<td>Kuehne et al. 2016 (HMM+GMM)</td>
<td>56.3</td>
</tr>
<tr>
<td>Richard et al. 2017 (HMM+RNN)</td>
<td>60.6</td>
</tr>
<tr>
<td>Kuehne et al. 2020 (HMM+RNN)</td>
<td>61.3</td>
</tr>
<tr>
<td>MS-TCN (TCN)</td>
<td>65.1</td>
</tr>
<tr>
<td>MS-TCN (TCN+I3D)</td>
<td>66.3</td>
</tr>
<tr>
<td><strong>MS-TCN++ (TCN+I3D)</strong></td>
<td><strong>67.6</strong></td>
</tr>
</tbody>
</table>

[S. Li et al. MS-TCN++: Multi-Stage Temporal Convolutional Network for Action Segmentation. arXiv]
MS-TCN++ vs. MS-TCN

(4x speed)

MS-TCN: action_start
MS-TCN++: action_start
gr. truth: action_start

MS-TCN

MS-TCN++

gr. truth
Weakly Supervised Learning

- Training video

- Fully supervised:

- Weakly supervised (transcripts)

A → C → F → D → A → E → H

Recall: Hybrid RNN-HMM

- HMM + RNN with Gated Recurrent Units (GRU)

\[
p(x_t | s) = \text{const} \cdot \frac{p(s | x_t)}{p(s)}
\]

input: video \( x_1^T \)

targets: subaction labels

\[
p(s | x_1) \quad p(s | x_2) \quad \cdots \quad p(s | x_T)
\]
Weakly Supervised Learning

Initial uniform splitting from transcripts:

- take_cup
- pour_milk
- spoon_powder
- pour_milk
- spoon_powder
- stir_milk
- stir_milk
Weakly Supervised Learning

Initial uniform splitting from transcripts:

RNN model training and optimization
The transcripts define the order of activities:

Action transcript:

action_1  action_2  action_3
The transcripts define the order of activities:

Action transcript:

action_1  action_2  action_3
Model

- The transcripts define the order of activities:

  Action transcript:

  \( \text{action}_1 \ \text{action}_2 \ \text{action}_3 \)
Weakly Supervised Learning

Action transcript:

action_1  action_2  action_3

linear segmentation

(Initialization)

Weakly Supervised Learning

Weakly Supervised Learning

Action transcript:

\textit{action\_1 action\_2 action\_3}

Weakly Supervised Learning

Weakly Supervised Learning

Action transcript:
- action_1
- action_2
- action_3

(linear segmentation)

(linear alignment to the subactions)

(retrain)

(train RNN)

(train HMM)

(realignment)

(linear alignment to the new subactions)

(retrain)

(train RNN)

(train HMM)

(realignment)


03.08.2020

Juergen Gall – Institute of Computer Science III – Computer Vision Group
### Results

<table>
<thead>
<tr>
<th>Breakfast frame accuracy (%)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>pseudo-GT (HMM+RNN)</td>
<td>33.3</td>
</tr>
<tr>
<td>Richard et al. 2017</td>
<td></td>
</tr>
<tr>
<td>pseudo-GT (HMM+RNN)</td>
<td>36.7</td>
</tr>
<tr>
<td>Kuehne et al. 2020</td>
<td></td>
</tr>
<tr>
<td>Fully supervised (HMM+RNN)</td>
<td>61.3</td>
</tr>
<tr>
<td>Kuehne et al. 2020</td>
<td></td>
</tr>
</tbody>
</table>

- Disadvantage: Offline and sensitive to initialization


Incremental learning

\[ c_1 \rightarrow \cdots \rightarrow c_N \]

(action transcript)

Viterbi Decoding

\[ \mathcal{L} = -\sum_{t=1}^{T} \log p(c_n(t)|x_t) \]

\[ p(c|x_1), \ldots, p(c|x_T) \]

Neural Network

\[ x_1, \ldots, x_T \]

(input video)

## Results

<table>
<thead>
<tr>
<th>Breakfast frame accuracy (%)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>pseudo-GT (HMM+RNN)</strong></td>
<td>33.3</td>
</tr>
<tr>
<td>Richard et al. 2017</td>
<td></td>
</tr>
<tr>
<td><strong>pseudo-GT (HMM+RNN)</strong></td>
<td>36.7</td>
</tr>
<tr>
<td>Kuehne et al. 2020</td>
<td></td>
</tr>
<tr>
<td><strong>NN-Viterbi (HMM+RNN)</strong></td>
<td>43.0</td>
</tr>
<tr>
<td>Richard et al. 2018</td>
<td></td>
</tr>
<tr>
<td><strong>Fully supervised (HMM+RNN)</strong></td>
<td>61.3</td>
</tr>
<tr>
<td>Kuehne et al. 2020</td>
<td></td>
</tr>
</tbody>
</table>

[A. Richard et al. **NeuralNetwork-Viterbi: A Framework for Weakly Supervised Video Learning.** CVPR 2018]
Pseudo GT vs. NN-Viterbi

(2.0x speed)

pseudoGt: Background
NN-Vit.: Background
correct: Background

pseudoGt
NN-Vit.
correct
Evaluation Issues

• Weakly supervised approaches are sensitive to initialization

<table>
<thead>
<tr>
<th>Model</th>
<th>MoF Reported</th>
<th>MoF Avg (± Std)</th>
<th>MoF Max</th>
<th>MoF Min</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISBA [4]</td>
<td>38.4</td>
<td>36.4 (± 1.0)</td>
<td>37.6</td>
<td>35.1</td>
</tr>
<tr>
<td>NNV [13]</td>
<td>43.0</td>
<td>39.7 (± 2.4)</td>
<td>43.5</td>
<td>37.5</td>
</tr>
<tr>
<td>CDFL [11]</td>
<td>50.2</td>
<td>48.1 (± 2.5)</td>
<td>50.9</td>
<td>44.6</td>
</tr>
</tbody>
</table>

[ L. Ding and C. Xu. Weakly-supervised action segmentation with iterative soft boundary assignment. CVPR 2018 ]
[ J. Li et al. Weakly supervised energy-based learning for action segmentation. ICCV 2019 ]
Features

- Some approaches struggle with pre-trained features (I3D)
- Dimensionality is just one issue

<table>
<thead>
<tr>
<th>Approach</th>
<th>Features</th>
<th>Average MoF</th>
</tr>
</thead>
<tbody>
<tr>
<td>NNV</td>
<td>IDT</td>
<td>40.6</td>
</tr>
<tr>
<td>NNV</td>
<td>I3D</td>
<td>11.4</td>
</tr>
<tr>
<td>NNV</td>
<td>PCA-I3D</td>
<td>23.2</td>
</tr>
<tr>
<td>CDFL</td>
<td>IDT</td>
<td>48.9</td>
</tr>
<tr>
<td>CDFL</td>
<td>I3D</td>
<td>34.9</td>
</tr>
<tr>
<td>CDFL</td>
<td>PCA-I3D</td>
<td>38.0</td>
</tr>
</tbody>
</table>

Weakly Supervised Learning

• Training video

• Fully supervised:

• Weakly supervised (transcripts)

Weakly Supervised Learning

• Fully supervised:

  A → C → F → D → A → E → H

• Weakly supervised (transcripts)
  A → C → F → D → A → E → H

• Weakly supervised (action set)
  \{A, C, D, E, F, H\}
  • Order unknown
  • Number of occurrence unknown

[ M. Fayyaz and J. Gall. SCT: Set Constrained Temporal Transformer for Set Supervised Action Segmentation. CVPR 2020 ]
## Results

<table>
<thead>
<tr>
<th>Supervision</th>
<th>frame accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCT (TCN+I3D)</td>
<td><strong>Action set</strong></td>
</tr>
<tr>
<td>Fayyaz and Gall 2020</td>
<td></td>
</tr>
<tr>
<td>pseudo-GT (HMM+RNN)</td>
<td><strong>Transcript</strong></td>
</tr>
<tr>
<td>Kuehne et al. 2020</td>
<td></td>
</tr>
<tr>
<td>NN-Viterbi (HMM+RNN)</td>
<td><strong>Transcript</strong></td>
</tr>
<tr>
<td>Richard et al. 2018</td>
<td></td>
</tr>
<tr>
<td>HMM+RNN</td>
<td><strong>Full</strong></td>
</tr>
<tr>
<td>Kuehne et al. 2020</td>
<td></td>
</tr>
<tr>
<td>MS-TCN++ (TCN+I3D)</td>
<td><strong>Full</strong></td>
</tr>
<tr>
<td>Li et al. arXiv</td>
<td></td>
</tr>
</tbody>
</table>
Source Code

- MS-TCN: https://github.com/yabufarha/ms-tcn
- ISBA: https://github.com/Zephyr-D/TCFPN-ISBA
- NN-Viterbi: https://github.com/alexanderrichard/NeuralNetwork-Viterbi
- CDFL: https://github.com/JunLi-Galios/CDFL
- Action sets: https://github.com/alexanderrichard/action-sets
- SCT: https://github.com/MohsenFayyaz89/SCT (Codes not uploaded yet)
- Unsupervised learning: https://github.com/Annusha/unsup_temp_embed
Thank you for your attention.