TokyoTech at TRECVID 2016

Localization using Faster R-CNN and Multi-Frame Fusion
Ryosuke Yamamoto, Nakamasa Inoue, Koichi Shinoda
Tokyo Institute of Technology

Motivation
- Localization task now includes not only static object, but also some action concepts
- We focus on “SittingDown”, one of action concepts
- Hard to distinguish from still Sitting only with static image input
- Utilizing dynamic information is important to detect it precisely

Bounding-Box Annotations
- For static objects, annotated on a key-frame for each positive shot
- 31K boxes on 26K shots
- For SittingDown, frame-wisely annotated to train LSTM
- 515 boxes on 92 shots

Our System
- Faster R-CNN (Ren 2015)
  - Efficient End-to-End object localizer
  - Generate region proposals from sparse sliding windows by a network itself
  - Predict each region using CNN features generated while generating proposals
  - We use ZF Net (Zeiler 2014)
- Multi-Frame Score Fusion (Inoue 2015)
  - Average pooling over 4 frames

Multi-Shot Score Boosting (Inoue 2015)
- Add adjacent shot scores

Long-Short Term Memory (Donahue 2015)
- Widely used for action detection
- Applied only to SittingDown

Results
- We archived 2nd among all 3 teams
- We got the best for SittingDown
- Frame-wise annotation helped
- LSTM with 4096 units did not work, seems over-fitted
- After submission, we confirmed LSTM with 64 units works well
- Scores of SittingDown
<table>
<thead>
<tr>
<th>Method</th>
<th>I-frame F-score</th>
<th>Pixel F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Without LSTM*</td>
<td>0.63</td>
<td>0.22</td>
</tr>
<tr>
<td>LSTM with 4096 units*</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>LSTM with 64 units</td>
<td><strong>11.96</strong></td>
<td><strong>4.51</strong></td>
</tr>
</tbody>
</table>

Conclusion
- We achieved 2nd among all 3 teams
- Best for SittingDown, LSTM did not work totally
- After submission, we confirmed LSTM works well

Future Work
- Find better way to detect SittingDown
Multimedia Event Detection Using Deep Features and LSTM
Na Rong, Nakamasa Inoue and Koichi Shinoda, Tokyo Institute of Technology

Proposed Method: Deep Features + LSTM

We propose a system using deep features and LSTM.

Motivation: Unless CNN, LSTM can make use of sequential information, which makes it applicable to MED.

Event detection framework:
Step 1. Extract deep features for each frame of input video
Step 2. Input deep features into an LSTM
There are 21 classes of the LSTM: 20 events and background.

Output

Probabilities for each event

<table>
<thead>
<tr>
<th>Average</th>
<th>1x21</th>
<th>1x21</th>
<th>1x21</th>
</tr>
</thead>
<tbody>
<tr>
<td>Softmax</td>
<td>Softmax</td>
<td>Softmax</td>
<td>Softmax</td>
</tr>
</tbody>
</table>

Deep Feature

Input

Experiments

Experimental Settings
- Extract frame every two seconds.
- Deep features [1, 2] are extracted from the pool5 layer of GoogLeNet trained on ImageNET
- Dimension of deep feature: 1,024
- Compare LSTM (256 units) and SVM

Comparison with other teams (PS 10EX)

Comparison with other teams (PS 100Ex)

Top 1 for “Attempting a bike trick” (LSTM)

Top 1 for “Attempting a bike trick” (SVM)

Conclusion

SVM results are greatly better than the LSTM results in evaluation set while in test dataset the gap between these two methods were not that huge, which may because LSTM is sensitive to the difference between LDC dataset (training and test) and YFCC dataset (evaluation).