Deep Structured Models for Human Activity Recognition

Greg Mori
School of Computing Science
Simon Fraser University
What does activity recognition involve?
Detection: are there people?
Objects and scenes: where are they?
Action recognition: what are they doing?
Intention/social role: why are they doing this? 

- comfort
- watch
- get help
Group activity recognition: what is the overall situation?

help the fallen person
These are inter-related problems: model structures
Desiderata for Activity Recognition Models

Label structure
- long term care facility
- walker
- indoor scene
- floor

Hu et al., CVPR 16
Deng et al., CVPR 16
Nauata et al., CVPRW 17
Deng et al., CVPR 17

Temporal structure
- time

Yeung et al., CVPR 16
Yeung et al., IJCV 17
He et al., WACV 18
Chen et al., ICCVW 17

Group structure
- help the fallen
- person

Ibrahim et al., CVPR 16
Mehrasa et al., arXiv 17
Khodabandeh et al., arXiv 17
Lan et al. CVPR 12
Image Classification

• A natural image can be categorized with labels at different concept layers

Hu, Deng, Zhou, Liao, Learning Structured Inference Neural Networks with Label Relations, CVPR 2016
Label Correlation Helps

• Such categorization at different concept layers can be modeled with label graphs
• It is natural and straightforward to leverage label correlation
Goal: A generic label relation model

- Infer the entire label space from visual input
- Infer missing labels given a few fixed provided labels

Hu, Deng, Zhou, Liao, Learning Structured Inference Neural Networks with Label Relations, CVPR 2016
Top-down Inference Neural Network

- Refine activations for each label
- Pass messages top-down and within each layer of label graph

Activation at current concept layer:

\[ a_t^i = V_{t-1,t} \cdot a_{t-1}^i + H_t \cdot x_t^i + b_t \]

Vertical weight propogates information across concept layers

Horizontal weight propogates information within concept layers

Activation at last concept layer:

\[ x_t^i = W_t \cdot CNN(I^i) + b_t \]

Visual Architecture

Produce initial visual activation from CNN

Top-down inference

Hu, Deng, Zhou, Liao, Learning Structured Inference Neural Networks with Label Relations, CVPR 2016
Bidirectional Inference Neural Network (BINN)

- Bidirectional inference to make information propagate across entire label structure
- Inference in each direction independently and blend results
Structured Inference Neural Network (SINN)

- BINN is hard to train
- Regularize connections with prior knowledge about label correlations
- Decompose connections into Positive correlation + Negative correlation

\[
V_{<t-1,t>}^+ = \begin{bmatrix}
w_{0,0} & w_{0,1} & w_{0,2} & 0 \\
0 & 0 & w_{1,2} & w_{1,3} \\
0 & w_{2,1} & 0 & w_{2,3} \\
0 & 0 & 0 & w_{3,3}
\end{bmatrix}
\]

Positive Correlation

\[
V_{<t-1,t>}^- = \begin{bmatrix}
0 & 0 & 0 & 0 \\
w_{1,0} & 0 & 0 & 0 \\
w_{2,0} & 0 & w_{2,1} & 0 \\
0 & w_{3,1} & w_{3,2} & 0
\end{bmatrix}
\]

Negative Correlation

Hu, Deng, Zhou, Liao, Learning Structured Inference Neural Networks with Label Relations, CVPR 2016
Structured Inference Neural Network (SINN)

- Evolve BINN formulation with regularization in connections

\[ \overrightarrow{a}_i = \gamma(\overrightarrow{V}_{t-1,t} \cdot \overrightarrow{a}_{t-1}) + \gamma(\overrightarrow{H}_t \cdot x_t^i) - \gamma(\overrightarrow{V}_{t-1,t} \cdot \overrightarrow{a}_{t-1}) - \gamma(\overrightarrow{H}_t \cdot x_t^i) + \overrightarrow{b}_t, \]

\[ \overleftarrow{a}_i = \gamma(\overleftarrow{V}_{t+1,t} \cdot \overleftarrow{a}_{t+1}) + \gamma(\overleftarrow{H}_t \cdot x_t^i) - \gamma(\overleftarrow{V}_{t+1,t} \cdot \overleftarrow{a}_{t+1}) - \gamma(\overleftarrow{H}_t \cdot x_t^i) + \overleftarrow{b}_t, \]

\[ \overrightarrow{a}_i = \overrightarrow{U}_t \cdot \overrightarrow{a}_t + \overrightarrow{U}_t \cdot \overleftarrow{a}_t + b_t \]

\[ \gamma(x) = ReLU(x) \]
Prediction from Purely Visual Input

- Visual architecture (e.g. Convolutional Neural Network) produces visual activation
- SINN implements information propagation bidirectionally and produces refined output activation

Hu, Deng, Zhou, Liao, Learning Structured Inference Neural Networks with Label Relations, CVPR 2016
Prediction with Partially Observed Labels

- Reverse Sigmoid (logit) neuron produces activation from Partial labels
- SINN adapts both visual activation and activation from partial labels to infer the remaining labels
Reverse sigmoid (logit): produce activation from label

- Reverse the sigmoid function to produce sigmoid input

**Inverse of sigmoid**

\[ y = \sigma(x) = \frac{1}{1 + \exp^{-x}} \]

**Use a small epsilon to keep numerical stability (0.005)**

\[
\begin{align*}
    a(y) &= \log\left(\frac{1}{1 - g(y)}\right), \\
    g(y) &= \begin{cases} 
        y + \epsilon, & \text{if } y = 0, \\
        y - \epsilon, & \text{if } y = 1.
    \end{cases}
\end{align*}
\]
### Image Datasets

- Evaluate with two types of experiments on three datasets

#### Animals with Attributes
[Lampert et al. 2009]

**Labels**
- 28 taxonomy terms
- 50 animal classes
- 85 attributes

**Task:** predict entire label set
- Taxonomy terms are constructed from Word Net as [Hwang et al. 2012]
- Knowledge graph constructed by combining class-attributes graph with taxonomy graph

#### NUS-WIDE
[Chua et al. 2009]

**Labels**
- 698 image groups
- 81 concepts
- 1000 tags

**Task:** predict 81 concepts with observing tags/image groups
- Knowledge graph produced by Word Net using semantic similarity
- 698 image groups constructed from image meta data

#### SUN 397
[Xiao et al. 2012]

**Labels**
- 3 coarse
- 16 general
- 397 fine-grained

**Task 1:** predict entire label set
**Task 2:** predict fine-grained scene given coarse scene category
- Knowledge graph provided by dataset
Ex1: Inference from visual input

- Produce predictions on entire label space
- Evaluate on each concept layer (measured by mAP per class)
- Consistent improvement over baselines on different concept layers

Animal With Attributes

- 28 Taxonomy Terms
- 50 Animal Classes
- 85 Attributes

SUN 397

- 3 Coarse Scene Categories
- 16 General Scene Categories
- 397 Fine-grained Scene Categories
Ex2: Inference from partial labels (NUS-WIDE)

- Produce predictions given partial 1k tags and 698 image groups

Correct predictions are marked in blue while incorrect are marked in red
Ex2: Inference from partial labels (NUS-WIDE)

- Evaluate on standard 81 ground truth classes of NUSWIDE
- Outperform all baselines by large margin

Hu, Deng, Zhou, Liao, Learning Structured Inference Neural Networks with Label Relations, CVPR 2016
Ex2: Inference with partial labels (SUN397)

- Produce predictions given coarse-level labels (3 coarse categories)

Correct predictions are marked in **blue** while incorrect are marked in **red**
Ex2: Inference with partial labels (SUN397)

• Evaluate on 397 fine-grained scene categories
• Significantly improved performance

SUN 397

- Multiclass Accuracy
- mAP per Class

- Image Features + SVM [Xiao et al. 2012]
- CNN + Logistics
- CNN + BINN
- CNN + SINN
- CNN + Partial Labels + Logistics
- CNN + Partial Labels + SINN

Hu, Deng, Zhou, Liao, Learning Structured Inference Neural Networks with Label Relations, CVPR 2016
Video Dataset: YouTube-8M

- Youtube-8M V1 / V2
  - 8 million / 7 million videos
  - ~500K hours of video
  - 4800 possible labels
  - 1.8 / 3.4 labels per video average

- Inception V3 frame features
- Neural network audio features
## Results

<table>
<thead>
<tr>
<th>Method</th>
<th>mAP / gAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>YouTube-8M v1</td>
<td></td>
</tr>
<tr>
<td>YouTube-8M v2</td>
<td></td>
</tr>
<tr>
<td>LSTM [Abu El Haija et al.]</td>
<td>26.6 / N/A</td>
</tr>
<tr>
<td>Logistic regression [Abu El Haija et al.]</td>
<td>28.1 / N/A</td>
</tr>
<tr>
<td>CNN features</td>
<td>27.98 / 60.34</td>
</tr>
<tr>
<td>BINN</td>
<td>31.18 / 64.74</td>
</tr>
<tr>
<td></td>
<td>40.19 / 76.33</td>
</tr>
</tbody>
</table>

**Abstract**

Videos are a rich source of high-dimensional structured data, with a wide range of interacting components at varying levels of granularity. In order to improve understanding of unconstrained internet videos, it is important to consider the role of labels at separate levels of abstraction. In this paper, we consider the use of the Bidirectional Inference Neural Network (BINN) for performing graph-based inference in label space for the task of video classification. We take advantage of the inherent hierarchy between labels at increasing granularity. The BINN is evaluated on the first and second release of the YouTube-8M large scale multi-label video dataset. Our results demonstrate the effectiveness of BINN, achieving significant improvements against baseline models.

**1. Introduction**

The proliferation of large-scale video datasets ([16], [19], [1], [13]), coupled with increasingly powerful computational resources allow for applications of learning on an unprecedented level. In particular, the task of labelling videos is of relevance with the massive flow of unlabelled user-uploaded video on social media. The complex, rich nature of video data strongly motivates the use of deep learning, and has seen measurable success in recent applications of video classification, captioning, and question answering ([6], [23], [22]).

Different labels such as outdoors and mountain, or beer and Irish pub have intrinsic dependencies on each other that are difficult for standard deep learning methods to model, as labels are generally assumed to be pairwise independent. Graphical models have seen promising results on incorporating label-space inference in image classification ([5], [9]). In particular, the work in [9] develops a Structured Inference Neural Network (SINN) and a Bidirectional Inference Neural Network (BINN) that performs hierarchical inference for image labelling. However, an image is a static stream of data relative to video. Models of temporal dependencies in sequential data have seen common use in both computer vision and natural language processing. In this paper, we investigate methods of incorporating recent methods of label inference with convolutional neural networks, effectively combining spatial and hierarchical information into a single end-to-end trainable network. Previous successful approaches to the problem of video classification include Convolutional Neural Networks (CNNs) [14], Recurrent Neural Networks (RNNs) such as LSTMs ([6], [17]), Improved Dense Trajectories (IDT) [24], and Histogram of Oriented Optical Flows (HOOF) [2]. However, all of these previous models discount the hierarchical dependencies between labels that could be leveraged to improve predictions – this paper is an attempt at resolving this disconnect. The explicit contribution of this paper is to extend the application of the BINN previously presented in [9] as a module for performing video classification.

![Diagram of proposed model for performing video label inference.](https://example.com/diagram.png)
Summary

• Inference in structured label space

• Relations within and across levels of a label space

• Model positive and negative correlations between labels in end-to-end trainable model
Desiderata for Activity Recognition Models

**Label structure**
- long term care facility
- walker
- indoor scene
- floor

Hu et al., CVPR 16
Deng et al., CVPR 16
Nauata et al., CVPRW 17
Deng et al., CVPR 17

**Temporal structure**
- time

Yeung et al., CVPR 16
Yeung et al., IJCV 17
He et al., WACV 18
Chen et al., ICCVW 17

**Group structure**
- help the fallen person

Ibrahim et al., CVPR 16
Mehrasa et al., arXiv 17
Khodabandeh et al., arXiv 17
Lan et al. CVPR 12
### MultiTHUMOS

Dense labels on 30 hours of THUMOS’14

<table>
<thead>
<tr>
<th></th>
<th>THUMOS</th>
<th>MultiTHUMOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Annotations</td>
<td>6,365</td>
<td>38,690</td>
</tr>
<tr>
<td>Classes</td>
<td>20</td>
<td>65</td>
</tr>
<tr>
<td>Labels per frame</td>
<td>0.3</td>
<td>1.5</td>
</tr>
<tr>
<td>Classes per video</td>
<td>1.1</td>
<td>10.5</td>
</tr>
<tr>
<td>Max actions per frame</td>
<td>2</td>
<td>9</td>
</tr>
<tr>
<td>Max actions per video</td>
<td>3</td>
<td>25</td>
</tr>
</tbody>
</table>

Modeling dense, multilabel actions

Standard LSTM: Single input, single output
Hochreiter 1997, Donahue 2014

Modeling dense, multilabel actions

All information about previous frames must be captured by current hidden state

Standard LSTM: Single input, single output
Hochreiter 1997, Donahue 2014

MultiLSTM

Frame class predictions

Input video frames

Standard LSTM: Single input, single output

Frame class predictions

Input video frames

MultiLSTM: Multiple inputs, multiple outputs

MultiLSTM

Frame class predictions

Expanded temporal receptive field of input and output connections reduces burden on hidden state

Input video frames

Standard LSTM: Single input, single output

MultiLSTM: Multiple inputs, multiple outputs

MultiLSTM

Soft attention over multiple inputs:

$$\alpha_{it} \propto \exp(w_{ae}^T \tanh(W_{ht}h_{i-1}) \odot \tanh(W_{va}v_t))$$

Standard LSTM: Single input, single output

MultiLSTM: Multiple inputs, multiple outputs

MultiLSTM

Frame class predictions

Weighted average over multiple outputs:

\[ y_t = \sum_i \beta_{it} p_{it} \]

Soft attention over multiple inputs:

\[ \alpha_{it} \propto \exp(w^T_{ae} [\tanh(W_{ha} h_{i-1}) \odot \tanh(W_{va} v_t)]) \]

Standard LSTM: Single input, single output

MultiLSTM: Multiple inputs, multiple outputs

**MultiLSTM**

Multilabel loss (per-class binary cross entropy):
\[ L(y|x) = \sum_{t,c} z_{tc} \log(\sigma(y_{tc})) + (1 - z_{tc}) \log(1 - \sigma(y_{tc})) \]

Weighted average over multiple outputs:
\[ y_t = \sum_i \beta_{it} p_{it} \]

Soft attention over multiple inputs:
\[ \alpha_{it} \propto \exp(w_{ae}^T [\tanh(W_{ha} h_{i-1}) \odot \tanh(W_{va} v_t)]) \]

**Frame class predictions**

**Input video frames**

**Standard LSTM:** Single input, single output

**MultiLSTM:** Multiple inputs, multiple outputs

## MultiLSTM

<table>
<thead>
<tr>
<th>Model</th>
<th>THUMOS mAP</th>
<th>MultiTHUMOS mAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDT</td>
<td>13.6</td>
<td>13.3</td>
</tr>
<tr>
<td>Single-frame CNN</td>
<td>34.7</td>
<td>25.4</td>
</tr>
<tr>
<td>Two-stream CNN</td>
<td>36.2</td>
<td>27.6</td>
</tr>
<tr>
<td>LSTM</td>
<td>39.3</td>
<td>28.1</td>
</tr>
<tr>
<td>LSTM+i</td>
<td>39.5</td>
<td>28.7</td>
</tr>
<tr>
<td>LSTM+i+a</td>
<td>39.7</td>
<td>29.1</td>
</tr>
<tr>
<td>MultiLSTM</td>
<td><strong>41.3</strong></td>
<td><strong>29.7</strong></td>
</tr>
</tbody>
</table>

Retrieving sequential and co-occurring actions

Sequential actions

Pass, then Shot

Throw, then One-handed catch

Jump, then Fall

Retrieving sequential and co-occurring actions

Sequential actions

Pass, then Shot  
Throw, then One-handed catch  
Jump, then Fall

Co-occurring actions

Dive & No Bodyroll  
Dive & Bodyroll  
Shot & Guard  
Shot & No Guard

Talk & Sit  
Talk & Stand

Task: action detection

Dominant paradigm: Dense processing

Standard in THUMOS challenge action detection entries
Oneata et al. 2014
Wang et al. 2014
Oneata et al. 2014
Yuan et al. 2015

Sliding windows

Action proposals

Efficiently detecting actions

Our model for efficient action detection

Detected actions

Video

t = 0
t = T

Our model for efficient action detection

Detected actions

Video

Our model for efficient action detection

Our model for efficient action detection

Detected actions

Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Video

Our model for efficient action detection

- **Detected actions**
- **Video**
- **Output**

**Outputs:**
- Detection *instance hypothesis* [start, end]
- **Recurrent neural network**
  (time information)
- **Convolutional neural network**
  (frame information)

Our model for efficient action detection

Detected actions

Outputs:
Detection instance hypothesis [start, end]
Emission indicator

Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Our model for efficient action detection

Detected actions

Outputs:
Detection instance hypothesis [start, end]
Emission indicator
Next frame to glimpse
Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Our model for efficient action detection

Outputs:
- Detection instance hypothesis [start, end]
- Emission indicator
- Next frame to glimpse

Recurrent neural network (time information)

Convolutional neural network (frame information)

Our model for efficient action detection

Detected actions

Outputs:
- Detection instance hypothesis [start, end]
- Emission indicator
- Next frame to glimpse

Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Our model for efficient action detection

Detected actions

Outputs:
Detection instance hypothesis [start, end]
Emission indicator
Next frame to glimpse
Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Our model for efficient action detection

Outputs:
Detection *instance hypothesis* [start, end]
Emission indicator
Next frame to glimpse
Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Our model for efficient action detection

Outputs:
- Detection instance hypothesis [start, end]
- Emission indicator
- Next frame to glimpse

Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Our model for efficient action detection

Detected actions

Video

Output

Output

Output

Outputs:
Detection instance hypothesis [start, end]
Emission indicator
Next frame to glimpse
Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Our model for efficient action detection

Outputs:
- Detection instance hypothesis [start, end]
- Emission indicator
- Next frame to glimpse

Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Our model for efficient action detection

Detected actions

Output

Output

Output

Outputs:
Detection *instance hypothesis* [start, end]
Emission indicator
Next frame to glimpse

Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Video

Training the detection instance output

\[ \mathcal{L}(D, G) = \sum_{i} \mathcal{L}_{cls}(d_i, y_i > 0) + \gamma \sum_{i:y_i > 0} \mathcal{L}_{loc}(d_i, g_{y_i}) \]

\( y_1 = 1 \quad y_2 = 1 \quad y_3 = 2 \quad y_4 = 0 \)

Training the non-differentiable outputs

Training data

\[ t = 0 \]

\[ t = T \]

Detections

\[ t = 0 \]

\[ t = T \]

Training the non-differentiable outputs

Training data

Detections

Model's action sequence $a$

Frame 1
Frame 8
Frame 6
Frame 15

(1) whether to predict a detection
(2) where to look next

Train an policy $\pi_\theta$ for actions (1) and (2) using REINFORCE [Williams 1992]

Train an policy $\pi(\theta)$ for actions (1) and (2) using REINFORCE [Williams 1992]

Reward for an action sequence $a$: $r(a) = N^+ - \alpha N^-$

Train an policy $\pi_\theta$ for actions (1) and (2) using REINFORCE [Williams 1992]

Reward for an action sequence $a$: $r(a) = N^+ - \alpha N^-$

Objective: $J(\theta) = \sum_a p_\theta(a) r(a)$

Gradient: $\nabla J(\theta) = \sum_a p_\theta(a) r(a) \nabla \log p_\theta(a)$

Monte-Carlo approximation: $\nabla J(\theta) \approx \frac{1}{K} \sum_{k=1}^K \sum_{t=1}^T \nabla \log \pi_\theta(a_t^k | M_t^k)$

Action detection results

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Detection AP at IOU 0.5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>State-of-the-art</td>
</tr>
<tr>
<td>THUMOS 2014</td>
<td>14.4</td>
</tr>
<tr>
<td>ActivityNet sports</td>
<td>33.2</td>
</tr>
<tr>
<td>ActivityNet work</td>
<td>31.1</td>
</tr>
</tbody>
</table>

While glimpsing only 2% of frames

Learned policies

Learned policies

Importance of prediction indicator output

<table>
<thead>
<tr>
<th></th>
<th>mAP (IOU = 0.5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours (full model)</td>
<td>17.1</td>
</tr>
<tr>
<td>Ours w/o prediction indicator output (always predict)</td>
<td>12.4</td>
</tr>
</tbody>
</table>

Deciding when to output a prediction (learning to do non-maximum suppression) matters.

Importance of location output

<table>
<thead>
<tr>
<th></th>
<th>mAP (IOU = 0.5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours (full model)</td>
<td>17.1</td>
</tr>
<tr>
<td>Ours w/o prediction indicator output (always predict)</td>
<td>12.4</td>
</tr>
<tr>
<td>Ours w/o location output (uniform sampling)</td>
<td>9.3</td>
</tr>
</tbody>
</table>

Deciding where to look next (location output) has even greater effect.

Importance of location output

Uniform sampling does not always have sufficient temporal resolution where it’s needed.

Removing both prediction indicator and location outputs

<table>
<thead>
<tr>
<th></th>
<th>mAP (IOU = 0.5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours (full model)</td>
<td>17.1</td>
</tr>
<tr>
<td>Ours w/o prediction indicator output (always predict)</td>
<td>12.4</td>
</tr>
<tr>
<td>Ours w/o location output (uniform sampling)</td>
<td>9.3</td>
</tr>
<tr>
<td>Ours w/o prediction indicator w/o location output (always predict, with uniform sampling)</td>
<td>8.6</td>
</tr>
</tbody>
</table>
## Importance of location regression

<table>
<thead>
<tr>
<th></th>
<th>mAP (IOU = 0.5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours (full model)</td>
<td>17.1</td>
</tr>
<tr>
<td>Ours w/o prediction indicator output (always predict)</td>
<td>12.4</td>
</tr>
<tr>
<td>Ours w/o location output (uniform sampling)</td>
<td>9.3</td>
</tr>
<tr>
<td>Ours w/o prediction indicator w/o location output (always predict, with uniform sampling)</td>
<td>8.6</td>
</tr>
<tr>
<td>Ours w/o location regression (always output mean action duration)</td>
<td>5.5</td>
</tr>
</tbody>
</table>

Simply outputting mean action duration gives significantly worse performance.

Desiderata for Activity Recognition Models

**Label structure**
- long term care facility
- walker
- indoor scene
- floor

Hu et al., CVPR 16
Deng et al., CVPR 16
Nauata et al., CVPRW 17
Deng et al., CVPR 17

**Temporal structure**
- time

Yeung et al., CVPR 16
Yeung et al., IJCV 17
He et al., WACV 18
Chen et al., ICCVW 17

**Group structure**
- help the fallen person

Ibrahim et al., CVPR 16
Mehrasa et al., arXiv 17
Khodabandeh et al., arXiv 17
Lan et al. CVPR 12
Role of Context in Actions

Who has the puck?
Analyzing Human Trajectories to Recognize Actions

Which team is it?
Who was player X?
Will the shot be successful?

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Motivation

Using trajectories of players on the rink:

- Player 1 is passing the puck to player 5
- Player 2 is trying to block player 1

Trajectory definition: sequence of player movements across space over time

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Motivation

locations matter!

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Key Player Definition
Model and Approach

- Shared-Compare Trajectory Network
- Stacked Trajectory Network

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Shared-Compare Trajectory Network

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Shared-Compare Trajectory Network

Shared Trajectory Net

Shared Comparison Net

Passed

Dump out

Dump in

Puck Protection

Carry

Shot
Shared Trajectory Network

- Consists of 1D convolution and max-pooling layers
- Learning generic representation for each individual

1D max-pooling layer

Kernel Size = C \times K \times M

Pooling stride = 2

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Shared-Compare Trajectory Network

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Shared Compare Network

Input:

- Pairs of individual trajectory features provided by Shared Trajectory Network
- Pairs are formed relative to a “key player”

Learning:

- The relative motion patterns of pairs
- Interaction cues of players

Output: relative motion pattern representation of each pair

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Players Ordering

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Relative Ordering

- Spatial proximity to the key player
- Key person may not be available in a general non-sports setting
- Average pooling strategy when key player is not provided

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Model and Approach

- Shared-Compare Trajectory Network
- Stacked Trajectory Network

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Stacked Trajectory Network

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Stacked Trajectory Network

- Learning overall group dynamics

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Experiments

- Event Recognition on the Sportlogiq Dataset
- Team Identification on the NBA Dataset

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Event recognition using Sportlogiq dataset

Task Definition

- Event classification
- 6 event classes
  - pass, dump in, dump out, shot, carry, puck protection
- Dataset: Sportlogiq hockey dataset

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Event recognition using Sportlogiq dataset

How the Sportlogiq dataset looks

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Event recognition using Sportlogiq dataset

- Sportlogiq Dataset Information
  - State of the art algorithms are used to automatically detect and track players in raw broadcast video
  - Trajectory data are estimated using homography
  - Trajectory length: 16 frames
  - # players used is fixed: 5
  - # of samples of each event
    - 4 games for training, 2 games for validation, and 2 games for testing

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Event recognition using Sportlogiq dataset

Baselines:

- IDT
  - Same input data as in our method
  - Each trajectory as IDT Trajectory shape descriptor
  - Normalized displacement vector of trajectory
  - SVM with RBF kernel and ‘one vs. rest’ mechanism

Event recognition using Sportlogiq dataset

Baselines:
- C3D
  - Trained from scratch
  - Fine-tuned from a model pretrained on Sports-1M
  - Same ordering as in our approach

Event recognition using Sportlogiq dataset

- Training phase:
  - Key player is provided
  - Remaining players are ranked by proximity to the key player

- Test phase:
  - Both cases of known and unknown key player
  - Average pooling strategy for the case of unknown key player
## Event recognition on Sportlogiq dataset

### Unknown Key Player

<table>
<thead>
<tr>
<th>Action</th>
<th>IDT</th>
<th>C3D</th>
<th>Fine-tuned C3D</th>
<th>Shared-Cmp</th>
</tr>
</thead>
<tbody>
<tr>
<td>pass</td>
<td>72.86%</td>
<td>71.10%</td>
<td>77.45%</td>
<td>78.13%</td>
</tr>
<tr>
<td>dump out</td>
<td>13.75%</td>
<td>11.66%</td>
<td>18.15%</td>
<td>22.14%</td>
</tr>
<tr>
<td>dump in</td>
<td>6.35%</td>
<td>7.58%</td>
<td>19.04%</td>
<td>26.63%</td>
</tr>
<tr>
<td>shot</td>
<td>13.05%</td>
<td>23.37%</td>
<td>38.96%</td>
<td>40.52%</td>
</tr>
<tr>
<td>carry</td>
<td>45.66%</td>
<td>64.75%</td>
<td>65.65%</td>
<td>61.10%</td>
</tr>
<tr>
<td>puck protection</td>
<td>6.28%</td>
<td>6.50%</td>
<td>7.98%</td>
<td>8.72%</td>
</tr>
<tr>
<td>mAP</td>
<td>26.32%</td>
<td>30.83%</td>
<td>37.87%</td>
<td>39.54%</td>
</tr>
</tbody>
</table>

- In comparison to IDT 13.2 higher mAP
- In comparison to C3D trained from scratch 8.7 higher mAP
- In comparison to fine-tuned C3D 1.7 higher mAP

### Known Key Player

<table>
<thead>
<tr>
<th>Action</th>
<th>IDT</th>
<th>C3D</th>
<th>Fine-tuned C3D</th>
<th>Shared-Cmp</th>
</tr>
</thead>
<tbody>
<tr>
<td>pass</td>
<td>73.35%</td>
<td>77.30%</td>
<td>84.34%</td>
<td>81.33%</td>
</tr>
<tr>
<td>dump out</td>
<td>14.34%</td>
<td>10.17%</td>
<td>17.10%</td>
<td>23.11%</td>
</tr>
<tr>
<td>dump in</td>
<td>5.77%</td>
<td>10.25%</td>
<td>24.83%</td>
<td>50.04%</td>
</tr>
<tr>
<td>shot</td>
<td>13.07%</td>
<td>34.17%</td>
<td>58.88%</td>
<td>48.51%</td>
</tr>
<tr>
<td>carry</td>
<td>47.38%</td>
<td>86.37%</td>
<td>90.10%</td>
<td>85.96%</td>
</tr>
<tr>
<td>puck protection</td>
<td>7.28%</td>
<td>11.83%</td>
<td>13.99%</td>
<td>11.54%</td>
</tr>
<tr>
<td>mAP</td>
<td>26.86%</td>
<td>38.35%</td>
<td>48.21%</td>
<td>50.08%</td>
</tr>
</tbody>
</table>
Event recognition on Sportlogiq dataset

Precision-recall curve
Experiments

- Event Recognition on the Sportlogiq Dataset
- Team Identification on the NBA Dataset
Team Identification on the NBA Dataset

Task Definition

- Team Identification
- Stacked Trajectory Network
- 30 NBA teams
- Dataset: NBA basketball dataset

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Team Identification on the NBA Dataset

How the NBA dataset looks like

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Team Identification using NBA dataset

- Dataset Information
  - Trajectory data are acquired by a multi-camera system
  - Sampling rate: 25Hz
  - Extract 137176 possessions from 1076 games
  - 200 frames per possession
  - 82375 poss. for training, 27437 poss. for testing, and 27437 poss. for validation
  - Number of poss. per team

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017
Team Identification on the NBA Dataset

Results

<table>
<thead>
<tr>
<th>layers</th>
<th>acc</th>
<th>hit@2</th>
<th>hit@3</th>
<th>game acc</th>
</tr>
</thead>
<tbody>
<tr>
<td>2conv</td>
<td>10.68%</td>
<td>18.09%</td>
<td>24.31%</td>
<td>50.00%</td>
</tr>
<tr>
<td>3conv</td>
<td>18.86%</td>
<td>28.89%</td>
<td>36.47%</td>
<td>87.05%</td>
</tr>
<tr>
<td>4conv</td>
<td>22.34%</td>
<td>33.03%</td>
<td>40.47%</td>
<td>93.41%</td>
</tr>
<tr>
<td>5conv</td>
<td>24.78%</td>
<td>35.61%</td>
<td>42.95%</td>
<td>95.91%</td>
</tr>
<tr>
<td>5conv+2fc</td>
<td>25.08%</td>
<td>35.83%</td>
<td>42.85%</td>
<td>94.32%</td>
</tr>
</tbody>
</table>
Team Identification on the NBA Dataset

Baseline: [1]

- IDT
  - Same input data as in our method
  - Each trajectory as IDT Trajectory shape descriptor
  - SVM with RBF kernel and ‘one vs. rest’ mechanism

<table>
<thead>
<tr>
<th>models</th>
<th>acc</th>
<th>game acc</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDT</td>
<td>5.74%</td>
<td>9.10%</td>
</tr>
<tr>
<td>Stacked Traj. Net</td>
<td>25.78%</td>
<td>95.91%</td>
</tr>
</tbody>
</table>

Summary

- Learning person trajectory representations for group activity analysis
- Using deep neural network models for learning trajectory features
- Experiments shows our model is capable of capturing:
  - Complex spatial-temporal dependencies
  - Distinctive group dynamics
Methods for handling *structures* in deep networks

**Label structure**: message passing algorithms for multi-level image/video labeling; purely from image data or with partial labels

**Temporal structure**: action detection in time; efficient glimpsing of video frames

**Group structure**: network structures to connect related people, gating functions or modules for reasoning about relations
Thank you!
Example: Rally in a Volleyball Game
Challenge:
- high level description
- aggregate information over whole scene
- focus on relevant people
Intuitive fix: use person-centric representation
Person Tracks

- Extract trajectories by tracking each person forward/backward in time
Stage 1: Learning Individual Action Features

Walking → LSTM → CNN (AlexNet) → Walking
Walking → LSTM → CNN (AlexNet) → Walking
Walking → LSTM → CNN (AlexNet) → Walking
Stage 1: Learning Individual Action Features

Person 1
Person 2
Person 3
Person N
Stage 2: Learning Frame Representations

Person 1  \[\rightarrow\text{LSTM} \rightarrow \text{Person 1 feature Representation}\]

Person 2  \[\rightarrow\text{LSTM} \rightarrow \text{Person 2 feature Representation}\]

Person 3  \[\rightarrow\text{LSTM} \rightarrow \text{Person 3 feature Representation}\]

Person N  \[\rightarrow\text{LSTM} \rightarrow \text{Person n feature Representation}\]

Aggregate  \[\rightarrow \text{LSTM}\]
Summary
Collective Activity Dataset

- Same label set for people and group activities
- 1925 video clips for training, 638 video clips for testing

Choi et al., VSWS 2009
## Collective Activity Dataset

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image Classification</td>
<td>63.0</td>
</tr>
<tr>
<td>Person Classification</td>
<td>61.8</td>
</tr>
<tr>
<td>Person - Fine tuned</td>
<td>66.3</td>
</tr>
<tr>
<td>Temp Model - Person</td>
<td>62.2</td>
</tr>
<tr>
<td>Temp Model - Image</td>
<td>64.2</td>
</tr>
<tr>
<td>Our Model w/o LSTM1</td>
<td>70.1</td>
</tr>
<tr>
<td>Our Model w/o LSTM2</td>
<td>76.8</td>
</tr>
<tr>
<td>Our Model</td>
<td>81.5</td>
</tr>
</tbody>
</table>
## Collective Activity Dataset

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contextual Model [Lan et al. NIPS’10]</td>
<td>79.1</td>
</tr>
<tr>
<td>Deep Structured Model [Deng et al. BMVC’15]</td>
<td>80.6</td>
</tr>
<tr>
<td><strong>Our Model</strong></td>
<td><strong>81.5</strong></td>
</tr>
<tr>
<td>Cardinality Kernel [Hajimirsadeghi &amp; Mori CVPR’15]</td>
<td><strong>83.4</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image Classification</td>
<td>63.0</td>
</tr>
<tr>
<td>Person Classification</td>
<td>61.8</td>
</tr>
<tr>
<td>Person - Fine tuned</td>
<td>66.3</td>
</tr>
<tr>
<td>Temp Model - Person</td>
<td>62.2</td>
</tr>
<tr>
<td>Temp Model - Image</td>
<td>64.2</td>
</tr>
<tr>
<td>Our Model w/o LSTM1</td>
<td>70.1</td>
</tr>
<tr>
<td>Our Model w/o LSTM2</td>
<td>76.8</td>
</tr>
<tr>
<td><strong>Our Model</strong></td>
<td><strong>81.5</strong></td>
</tr>
</tbody>
</table>
Volleyball Dataset – Frame Labels

- 4830 frames annotated from 55 volleyball videos
- 2/3 videos for training, 1/3 testing
- 9 player action labels
- 4 scene labels

Left/right team variants
Volleyball Dataset – People Labels

- Waiting
- Digging
- Setting
- Spiking
- Falling
- Jumping
- Moving
- Standing
- Blocking
## Experimental results on Volleyball Dataset

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image Classification</td>
<td>66.7</td>
</tr>
<tr>
<td>Person Classification</td>
<td>64.5</td>
</tr>
<tr>
<td>Person - Fine tuned</td>
<td>66.8</td>
</tr>
<tr>
<td>Temp Model - Person</td>
<td>67.5</td>
</tr>
<tr>
<td>Temp Model - Image</td>
<td>63.1</td>
</tr>
<tr>
<td>Our Model w/o LSTM1</td>
<td>73.3</td>
</tr>
<tr>
<td>Our Model w/o LSTM2</td>
<td>80.9</td>
</tr>
<tr>
<td>Our Model</td>
<td>81.6</td>
</tr>
</tbody>
</table>

**Dense trajectories:** 73.4-78.7
## Visualization of results

<table>
<thead>
<tr>
<th>Left set</th>
<th>Right pass</th>
<th>Right Spike</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Left set image" /></td>
<td><img src="image2.png" alt="Right pass image" /></td>
<td><img src="image3.png" alt="Right Spike image" /></td>
</tr>
<tr>
<td><img src="image4.png" alt="Left pass image" /></td>
<td><img src="image5.png" alt="Left spike (Left pass)" /></td>
<td><img src="image6.png" alt="Right spike (Left spike)" /></td>
</tr>
</tbody>
</table>
Summary

- A two stage hierarchical model for group activity recognition

- LSTMs as a highly effective temporal model and temporal feature source

- People-relation modeling with simple pooling