Video Recognition
Recap: Faster R-CNN

Faster R-CNN

Per-image computation

\[ f_1 = FCN(I) \]

Per-region computation for each \( r_i \in r(I) \)

- **RPN** \( f_1 \)
- **RoIPool**
- **MLP**
- Softmax clf.
- Box regressor

*Learned proposals Shares computation with whole-image network*

Credit: R. Girshick
Recap: Faster R-CNN, RPN

RPN: Region Proposal Network

\[ f_i = FCN(I) \]
Recap: Faster R-CNN, RPN

RPN: Region Proposal Network

\[ f_i = FCN(I) \]

3x3 “sliding window” Scans the feature map looking objects

Credit: R. Girshick
Recap: Faster R-CNN, RPN

RPN: Anchor Box

- Anchor box: predictions are w.r.t. this box, *not the 3x3 sliding window*
- 3x3 “sliding window”
  - Objectness classifier
  - Box regressor predicting \( (dx, dy, dh, dw) \)

\[ f_1 = FCN(I) \]
Recap: Faster R-CNN, RPN

RPN: Prediction (on object)

- Objectness score
- Anchor box: transformed by box regressor
- 3x3 “sliding window”
  - Objectness classifier
  - Box regressor predicting (dx, dy, dh, dw)

P(object) = 0.94

Credit: R. Girshick
Recap: Faster R-CNN, RPN

RPN: Prediction (off object)

- Objectness score
- Anchor box: transformed by box regressor
- 3x3 “sliding window”
  - Objectness classifier
  - Box regressor predicting (dx, dy, dh, dw)

P(object) = 0.02

Credit: R. Girshick
Recap: Faster R-CNN

RPN: Multiple Anchors

- Anchor boxes: $K$ anchors per location with different scales and aspect ratios
- $f_1 = FCN(I)$
- 3x3 “sliding window”
  - $K$ objectness classifiers
  - $K$ box regressors

Credit: R. Girshick
Recap: Faster R-CNN, RPN

Faster R-CNN

Per-image computation

\[ f_1 = FCN(I) \]

Per-region computation for each \( r_i \in r(I) \)

\[ \text{RPN}(f_1) \]

\[ \text{RolPool} \]

\[ \text{MLP} \]

\[ \text{Softmax clf.} \]

\[ \vdots \]

\[ \text{Box regressor} \]

Learned proposals

*Shares computation with whole-image network*

Credit: R. Girshick
Recap: Faster R-CNN, RoI Pooling

RoIPool (on each Proposal)

Transform arbitrary size proposal into a fixed-dimensional representation (e.g., 2x2)

$\mathcal{f}_i = \text{FCN}(I)$

Credit: R. Girshick
R-FCN is a variant of Faster R-CNN

R-FCN

Per-image computation

\[ f_1 = FCN(I) \]

Per-region computation for each \( r_i \in r(I) \)

\[ \text{AvgPool} \rightarrow \text{Softmax clf.} \]

\[ \text{AvgPool} \rightarrow \text{Box regressor} \]

Extremely light-weight per-region computation

Credit: R. Girshick
Object detection in the wild by Faster R-CNN + ResNet-101

Model pre-trained on ImageNet, fine-tuned on MS COCO that has 80 categories. Frame-by-frame detection, no temporal processing.

Credit: K. He
Increasing the difficulty of the task: ImageNet VID
Object Detection and Tracking in Video

Feichtenhofer, Pinz, Zisserman, ICCV 2017
Object Detection from Video: ImageNet VID Challenges

- View point change
- Illumination variation
- Motion blur
- Occlusion

Feichtenhofer, Pinz, Zisserman, ICCV 2017
Detect & Track Architecture

Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks
Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, In TPAMI, 2016

Can be seen as a multi-target version of a regression tracker
Can be seen as a multi-target version of a correlation tracker

R-FCN: Object Detection via Region-based Fully Convolutional Networks
Jifeng Dai, Y Li, Kaiming He, and Jian Sun, NIPS, 2016

Fully-convolutional networks for object tracking

Feichtenhofer, Pinz, Zisserman, ICCV 2017
Detect & Track Training: Forward pass

Video Frames → Convolutional Feature Maps → RPN → RoI Pooling → RoI Tracking

- Multi-task objective for frame-based object detection and across-frame track regression

\[
L(\{p_i\}, \{b_i\}, \{\Delta_i\}) = \frac{1}{N} \sum_{i=1}^{N} L_{cls}(p_i, c_i^*) + \lambda \frac{1}{N_{fg}} \sum_{i=1}^{N} [c_i^* > 0] L_{reg}(b_i, b_i^*) + \lambda \frac{1}{N_{tra}} \sum_{i=1}^{N_{tra}} L_{tra}(\Delta_{i}^{t+\tau}, \Delta_{i}^{*,t+\tau})
\]

Classification softmax scores → Frame Rols → Track Rols
Detect & Track Training: Backward pass
Detect & Track: Testing

Feichtenhofer, Pinz, Zisserman, ICCV 2017
Detect & Track: Testing

Feichtenhofer, Pinz, Zisserman, ICCV 2017
RoI Tracking

\[ \text{tracks} \quad \text{frame} t \rightarrow t + \tau \]

RoI Pooling

\[ \Delta_x, \Delta_y, \Delta_w, \Delta_h \]

RoI Tracking

"detections" frame t

"detections" frame t+\tau

Detect & Track: Testing

Feichtenhofer, Pinz, Zisserman, ICCV 2017
Detect & Track: Temporally-Strided Testing

Feichtenhofer, Pinz, Zisserman, ICCV 2017
Qualitative Results

Code & Models: github.com/feichtenhafer/Detect-Track/

Results: robots.ox.ac.uk/~vgg/research/Detect-Track/

Feichtenhofer, Pinz, Zisserman, ICCV 2017
Detect & Track: Qualitative Results
Difficult cases

Feichtenhofer, Pinz, Zisserman, ICCV 2017
Image and Video Understanding

- **Representations for Video Classification**
  - Video object detection
    - Feichtenofer, Pinz, Zisserman, Detect to Track and Track to Detect, ICCV 2017
  - Hand-designed features
    - Wang et al., Action Recognition by Dense Trajectories, CVPR 2011.
  - Spatiotemporal ConvNets
    - Karpathy et al., Large-scale Video Classification with Convolutional Neural Networks, CVPR 2014
  - Two-stream ConvNets
Action classification

I. Laptev and P. Pérez, Retrieving actions in movies, ICCV'07
Datasets and evaluation protocols

Datasets

- **UCF101**: 101 classes, 13K videos, ~180 frames in a video
- **HMDB51**: 51 classes, 6.8K videos

Evaluation protocol: average classification accuracy over 3 train and test splits
Related work (implicit motion)

Karpathy et al., Large-scale Video Classification with Convolutional Neural Networks, CVPR 2014

Tran et al., Learning Spatiotemporal Features with 3D Convolutional Networks, ICCV 2015

- C3D Architecture
  - 8 convolution, 5 pool, 2 fully-connected layers
  - 3x3x3 convolution kernels
  - 2x2x2 pooling kernels
Large-scale Video Classification

Sports-1M dataset

- 1 million YouTube videos in 487 classes of sports
Amazing what a human observer can do without spatial information
Motivation: Separate visual pathways for perception and action

The Human Visual Cortex has two hierarchical pathways

- Ventral stream performs object recognition
- Dorsal stream recognizes motion and locates objects

Spatial ConvNet?

Temporal ConvNet?

David C. Van Essen, Jack L. Gallant, Neural mechanisms of form and motion processing in the primate visual system, Neuron, Volume 13, Issue 1, July 1994, Pages 1-10, ISSN 0896-6273
Two-Stream Convolutional Networks for Action Recognition in Videos

Individual processing of spatial and temporal information
- Using a separate ConvNet recognition stream for each
- Late fusion via softmax score averaging

Figure 1: Two-stream architecture for video classification.

[Simonyan and Zisserman NIPS'14]
Spatial stream ConvNet

<table>
<thead>
<tr>
<th></th>
<th>conv1</th>
<th>conv2</th>
<th>conv3</th>
<th>conv4</th>
<th>conv5</th>
<th>full6</th>
<th>full7</th>
<th>softmax</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>7x7x96</td>
<td>5x5x256</td>
<td>3x3x512</td>
<td>3x3x512</td>
<td>3x3x512</td>
<td>4096</td>
<td>2048</td>
<td></td>
</tr>
<tr>
<td></td>
<td>stride 2</td>
<td>stride 2</td>
<td>stride 1</td>
<td>stride 1</td>
<td>stride 1</td>
<td>dropout</td>
<td>dropout</td>
<td>dropout</td>
</tr>
<tr>
<td></td>
<td>norm.</td>
<td>norm.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>pool 2x2</td>
<td>pool 2x2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Performs image classification on single RGB frames

Training:
- Supervised pre-training on ILSVRC (1.2M images in 1000 classes)
- Fine tuning of the softmax layer using the video frames

Testing
- ConvNet processes every 25th frame of a video
- Data augmentation: 10 ConvNet inputs for each frame (crops & flips)
- Results for all ConvNets are averaged

[Simonyan and Zisserman NIPS’14]
Same model as in the spatial net except:

- Optical flow over several frames acts as input.

**How to optimally combine optical flow for ConvNet input?**

Figure 2: **Optical flow.** (a),(b): a pair of consecutive video frames with the area around a moving hand outlined with a cyan rectangle. (c): a close-up of dense optical flow in the outlined area; (d): horizontal component $d^x$ of the displacement vector field (higher intensity corresponds to positive values, lower intensity to negative values). (e): vertical component $d^y$. Note how (d) and (e) highlight the moving hand and bow. The input to a ConvNet contains multiple flows (Sect. 3.1).

Horizontal and vertical flow is rescaled to [0, 255] for ConvNet input.
Optical flow stacking

Stack horizontal and vertical displacement fields $d$

- Optical flow, $d$, over several frames, $\tau$, acts as input $I_\tau$ to the network

Input $I_\tau$ at $p_1$ represents motion at $p_1$ across multiple frames

[Simonyan and Zisserman NIPS'14]
Motivation: Separate visual pathways in nature

Dorsal stream (‘where/how’) recognizes motion and locates objects

Ventral (‘what’) stream performs object recognition

https://en.wikipedia.org/wiki/Two-streams_hypothesis
We study a number of ways of **fusing** two-stream ConvNets [Simonyan & Zisserman, NIPS’14]

- Sum fusion works surprisingly well

\[ f \text{ can be initialized as a sum kernel + feature identity mapping} \]

Feichtenhofer, Pinz, Zisserman, CVPR 2016
Where to fuse the network streams?

Two examples of where fusion layers can be placed:

Feichtenhofer, Pinz, Zisserman, CVPR 2016