Video Object Detection based on STMM

Barış Can Çam

Middle East Technical University
Department of Computer Engineering

April 27, 2018
Overview of the Presentation

1. Introduction
   - Object Detection Problem Explained
   - Object Detection in Videos
   - Past Efforts in Video Object Detection

2. Related Work
   - Still Image Object Detection
   - Video Object Detection
   - Modeling Sequential Data with RNNs

3. Proposed Method
   - Overview
   - STMM Module
   - Modified BatchNorm Function
   - MatchTrans Module

4. Experiments & Results
   - Implementation Details
   - Comparison with Other Methods
   - Qualitative Results
   - Ablation Studies

5. Conclusion
Object Detection Problem

Given an image find out:

1. A bounding box for each object (localization).
2. Class labels of objects with confidence scores.
Running Still Image Detector Independently in Each Frame:

- Oscillatory class predictions of objects with confidence scores.
- Lack of generalization performance under challenging conditions.
- Loosing rich temporal and motion information during training and test.
An example result of still image detector on video
Past Effort in Video Object Detection

The community put attention on using the temporal information in the video for video object detection.

- Static object detectors like R-FCN or FAST-RCNN are used, and results are linked across frames using post-processing methods.
- This effort only provide sub-optimal results, because temporal information is not used during training.
Past Effort in Video Object Detection

The community put attention on using the temporal information in the video for video object detection.

- More recent works tried to use the temporal information across neighboring frames during training.
- But these methods are not able to model variable and long-term temporal behavior of objects due to fixed-length temporal window sizes.
Past Effort in Video Object Detection: Tubelet Proposal Network [Kang et al.]

- Uses vectors to represent the memory which loses spatial information.
- In order to compensate loss of spatial information, they used region-level sampling of each tube (proposals).
Solution

1. Aligned Spatial-Temporal Memory Module:
   - Convolutional Gated Recurrent Computation Unit.
   - Model and align object’s long-term appearance and motion dynamics.
   - Has pre-trained RFCN detector on ImageNet-DET which has large intra-class object diversity.
   - A novel MatchTrans module to model displacement introduced by motion across frames.
We will cover the related works in three main sections as follows:

- Still image object detection.
- Video object detection.
- Modeling sequential data with RNNs.
Still Image Object Detection: RFCN

[Diagram showing the process of RFCN for still image object detection]
Related Work

Video Object Detection: Detect to Track & Track to Detect

[Diagram of Video Object Detection process]

Video Frames

Convolutional Feature Maps

Correlation

RoI Tracking

RoI Pooling

Loss

Cls

Reg

Frame t

Frame t+τ

RPN

Advanced Deep Learning Paper Presentation (METU) Video Object Detection with STMM
April 27, 2018 12 / 35
Video Object Detection: Detect to Track & Track to Detect

**Drawbacks:**
- Only uses consecutive frames.
- Only considers box displacements to predict movements.
- Alignment of features are not considered between video frames.
Review: RNN

Recall the RNN:

\[ h_t = f(W^{hh} h_{t-1} + W^{hx} x_t) \]
\[ y_t = \text{softmax}(W^S h_t) \]
\[ J_t(\Theta) = \sum_{i=1}^{|V|} (y_{ti} \log y_{ti}) \]
Review: GRU

\[
\begin{align*}
  z_t &= \sigma(W^{(z)}x_t + U^{(z)}h_{t-1}) \\
  r_t &= \text{softmax}(W^{(r)}x_t + U^{(r)}h_{t-1}) \\
  h_t' &= \tanh(Wx_t + r_t \odot Uh_{t-1}) \\
  h_t &= z_t \odot h_{t-1} + (1 - z_t) \odot h_t'
\end{align*}
\]
Modeling Sequential Data with RNNs: ConvGRU

- Different stability characteristics of features in temporal domain.
- As going from bottom to up temporal variances of spatial features decreases.
Modeling Sequential Data with RNNs: ConvGRU

\[
\begin{align*}
    z_t^L &= \sigma(W_z^L \odot x_t^L + U_z^L \odot h_{t-1}^L) \\
    r_t^L &= \text{softmax}(W_r^L \odot x_t^L + U_r^L \odot h_{t-1}^L) \\
    h_t'^L &= \tanh(W^L \odot x_t^L + U \odot (r_t^L \odot h_{t-1}^L)) \\
    h_t^L &= (1 - z_t^L)h_{t-1}^L + z_t^L h_t'^L
\end{align*}
\]
The proposed architecture is basically a modified version of ConvGRU with following modifications:

- Classification of bounding boxes rather than pixels or frames.
- Recurrent computation unit called STMM with static image detector which is pre-trained on ImageNet-DET dataset.
- Spatial-temporal memory alignment with Match-Trans module.
Proposed Method: Overview

\[ L(s, tx, y, w, h) = L_{cls}(s_c^* + \lambda[c^* > 0]L_{reg}(t, t^*) \]

- \[ L_{cls}(s_c^*) = -\log(s_c^*) \]
- \[ L_{reg} \] is the L1 loss.
Proposed Method: STMM Module

- \( z_t = \text{BN}^*(\text{ReLU}(W_z \odot F_t + U_z \odot M_{t-1})) \)
- \( r_t = \text{BN}^*(\text{ReLU}(W_r \odot F_t + U_r \odot M_{t-1})) \)
- \( M'_t = \text{ReLU}(W \odot F_t + U \odot (M_{t-1} \odot r_t)) \)
- \( M_t = (1 - z_t) \odot M_{t-1} + z_t \odot M'_t \)
Proposed Method: Modified BatchNorm Function

In order to keep the output of gates range in \([0, 1]\) following changes in BatchNorm function is done:

- Modified mean \(\mu(X)\) and the standard deviation \(\sigma(X)\) of each batch \(X\).
- Applying BatchNorm to each batch one by one.
Proposed Method: What is different from ConvGRU?

- Remember the initialization of Convolutional modules with pre-trained weights.
- Main issue is that; how to integrate these weights to RFCN architecture.
Proposed Method: Remember the Overview

Diagram:
- CONV ➔ STMM ➔ CONV ➔ STMM ➔ CONV ➔ STMM ➔ CONV
- ROIs ➔ Position Sensitive RoIPooling
- Position Sensitive RoIPooling ➔ Class ➔ Boxreg
Proposed Method: Modifications to entail STMM module.

- As shown in equations they replaced non-linearity functions from sigmoid and \textit{Tanh} to \textit{ReLU}. Because of the compatibility of recurrent unit with its input and output modules.

- $W_z$, $W_r$ and $W$ activations are replaced with pre-trained weights rather than initializing them with random values.
Proposed Method: MatchTrans Module

- How to handle mis-aligned features in the memory, .
- Which cause false negative detections in individual frames for a single object.
Proposed Method: Alignment Problem in the Memory
Proposed Method: MatchTrans Module

\[ \Gamma_{x,y}(i,j) = \frac{F_t(x,y) \cdot F_{t-1}(x+i,y+j)}{\sum_{i,j \in (-k,k)} F_t(x,y) \cdot F_{t-1}(x+i,y+j)} \]

\[ M'_{t-1} = \sum_{i,j \in (-k,k)} \Gamma_{x,y}(i,j) \cdot M_{t-1}(x+i, y+j) \]
Proposed Method: After Transformation
ImageNet-VID Dataset

1. Large size dataset for video object detection task with following properties;
   - 3862 videos for training.
   - 555 videos for validation.
   - 937 videos for test.
   - 30 categories with provided bounding box annotations.
The following implementation details are provided by the authors:

- DeepMask for object proposals.
- Pre-trained weight in ImageNet-DET.
- $T = 7$ for training, $T = 11$ for test.
- $1 \times 1 \times 512$ Feature Dimensionality.
- $1/10$ sampling through frames.
- Learning Rate: From $10^{-3}$ to $10^{-4}$.
- Left-to-right flipping data augmentation.
Comparison with Other Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Base network</th>
<th>Base detector</th>
<th>Test</th>
<th>Val</th>
</tr>
</thead>
<tbody>
<tr>
<td>STMN (Ours)</td>
<td>ResNet-101</td>
<td>R-FCN</td>
<td>-</td>
<td>80.5</td>
</tr>
<tr>
<td>Zhu et al. [50]</td>
<td>ResNet-101+DCN</td>
<td>R-FCN</td>
<td>-</td>
<td>78.6</td>
</tr>
<tr>
<td>FGFA [51]</td>
<td>ResNet-101</td>
<td>R-FCN</td>
<td>-</td>
<td>78.4</td>
</tr>
<tr>
<td>T-CNN [23]</td>
<td>DeepID+Craft [33 48]</td>
<td>RCNN</td>
<td>67.8</td>
<td>73.8</td>
</tr>
<tr>
<td>R-FCN [8]</td>
<td>ResNet-101</td>
<td>R-FCN</td>
<td>-</td>
<td>73.4</td>
</tr>
<tr>
<td>TPN [22]</td>
<td>GoogLeNet</td>
<td>TPN</td>
<td>-</td>
<td>68.4</td>
</tr>
<tr>
<td>STMN (Ours)</td>
<td>VGG-16</td>
<td>Fast-RCNN</td>
<td>56.5</td>
<td>61.7</td>
</tr>
<tr>
<td>Faster-RCNN [118]</td>
<td>VGG-16</td>
<td>Faster-RCNN</td>
<td>48.2</td>
<td>52.2</td>
</tr>
<tr>
<td>ITLab VID - Inha [1]</td>
<td>VGG-16</td>
<td>Fast-RCNN</td>
<td>51.5</td>
<td>-</td>
</tr>
</tbody>
</table>
Qualitative Results
Ablation Studies

<table>
<thead>
<tr>
<th></th>
<th>STMN</th>
<th>STMN No-MatchTrans</th>
<th>ConvGRU Pretrain</th>
<th>ConvGRU FreshFC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test mAP</td>
<td>50.7</td>
<td>49.0</td>
<td>48.0</td>
<td>44.8</td>
</tr>
</tbody>
</table>

- MatchTrans module ablation
- Replacement of STMM with ConvGRU, randomly initialized fully connected layers.
- Finally, ConvGRU with pre-trained weights.
Conclusion

1. In conclusion proposed architecture gives the state of the art results in ImageNet-VID dataset.
2. Their proposed architecture provides improvement because it can model spatial-temporal behavior of videos successfully.
3. There are still open issues in this work which are;
4. Their method comes with a computational overhead of 0.028 seconds/frame compared with R-FCN. This overhead is due to STMM and MatchTrans modules. (On a single TitanX GPU)
5. There are still open issues about this work;
   - Is it possible to find a method for warping features instead of MatchTrans module, which will reduce the dependency to the hyper-parameters such as local region size $k$. (Optical Flow based methods are available, but computational cost is high)
   - They did not show results for different temporal window sizes used in training stage.
   - Is it possible to find the optimal number of channels for the activations in spatial memory module?
   - Although they include idea of Why they need an extra alignment module
Thank you for listening.