Video Object Detection by Aggregating Features across Adjacent Frames

Ruyi Zhang*, Zhenjiang Miao, Qiang Zhang, Shanshan Hao and Shengbo Wang

Beijing Jiaotong University, Institute of Information Science, School of Computer and Information Technology, No.3 Shangyuancun, Haidian District, Beijing, China, 100044

*Email: 17120331@bjtu.eud.cn

Abstract. Video object detection is of great significance for video analysis. In contrast to object detection in still image, video object detection is more challenging which suffers from motion blur, varying view-points/pose, and occlusion. Existing methods utilized temporal information during detection in videos and show improvement over static-image detector. In this paper, we propose a novel method for video object detection that can aggregate features across adjacent frames adaptively as well as capture more global cues so as to be more robust to drastic appearance changes. Initially, current frame feature and warped feature from adjacent frames can be obtained via feature extraction network and optical flow network. Next, a coherence contribution module is designed for adaptively feature aggregation of the two kinds of features obtained from the first step. Finally, the still-image detector which included an extra instance-level module that aggregates features from adjacent frames for capturing more global feature is adopted to get the final result. The experimental results evaluated on our method shows leading performance on the ImageNet VID dataset.

1. Introduction

Image object detectors like SSD [1], Faster R-CNN [2] and R-FCN [3] have achieved great success in the past decade. Recently, the topic of object detection in videos has gained more attention and is important in many real applications such as video surveillance, automatic driving, robot navigation and so on. The simplest way to detect object in video is to apply these image-level detectors on each frame of videos due to the movement of objects or cameras. Directly using image-level detectors for detecting objects in videos achieves far more unsatisfactory performance.

In contrast to images, there exists much extra information among video frames can be exploited for object detection in videos. For instance, it is of high probability that class labels and locations of objects in adjacent frames are similar or even the same. Based on this idea, researchers have designed several video object detectors [4, 5, 6, 7, 8, 9, 10] and they can be divided into box-level detectors [4, 5, 6, 7] and feature-level detectors [8, 9, 10]. [4, 5, 6, 7] employ manually-designed association rules to link bounding-boxes across frames to form sequences of bounding boxes. Then by using the information of tubelet proposals obtained before, scores of each frame can be refined. This kind of post-processing method only applies during testing, hence in theory it is not optimal. By contrast, feature-level detectors [8, 9, 10] aim to improve per-frame feature quality through fusing useful information from adjacent frames. Therefore, the enhanced features can produce high quality boxes and thus get better detection results. [9] can aggregate features adaptively from adjacent frames but unable to handle the problem of
dramatic changes, [10] can solve the problem of dramatic changes of objects but ignores that frames from different location contributes differently to current frames.

In this paper, we propose a novel model can avoid the issues mentioned above to a certain extent. To be more specific, current frame feature and warped features from adjacent frames are adopted through the backbone architecture as well as the flow network for further aggregation. In order to combine the two kinds of features adaptively, we adopt a coherence contribution module to compute the contribution of each frame to current frame. Moreover, an extra instance-level aggregation module that aggregates instance feature from adjacent is introduced to the detection network for estimating instance-level movement so as to be more robust to sudden appearance changes.

![Figure 1](image)

**Figure 1.** An illustration of the common challenges associated with object detection in video. These are motion blur, unusual view-points/pose, occlusion, video defocus.

2. Related Work

2.1. Object detection in static images

Object Detection is one of the core issues in the field of computer vision, for which academic community has studied nearly 20 years. State-of-the-art methods for image object detection [1, 2, 3] are mainly based on deep neural networks [11, 12, 13]. There are two main streams for image object detection based on deep neural networks: one is the region-based object detectors like Faster R-CNN [1] and R-FCN [2]. The other is the one-stage detectors like SSD [3]. The main difference between the two methods is that the two stage algorithm needs to generate proposals first, and then perform fine-grained object classification and regression based on the proposals. The one stage algorithm extracts features from the feature extraction network, then directly utilizes the features to predict the class labels and the corresponding location of the objects.

While the methods mentioned above are specially designed for image detection. They process each image independently without taking the temporal information between video frames into account. Our method focuses on object detection in videos. Temporal information will be joined to our method to get high-quality feature maps per-frame. As a consequence, high-quality proposals are obtained to get better result. Our method can easily benefit from the improvement of still-image detectors.

2.2. Object detection in videos

There was less research on video object detection compared to image object detection before the introduction of the ImageNet VID challenge in 2015. The introduction of this video dataset greatly promotes the research on video object detection.
As far as we know, there exists two main methods for object detection in videos: one of the solutions is to explore mapping strategies to link still image detection results across frames. [4] links bounding-box from different frames if their IOU is higher than a certain threshold, then it selects the sequences with the highest score and re-ranking the boxes through the method called “Seq-NMS”. Kang et al. [5] exploits a new rule to generate high-quality tubelet proposals efficiently, then the Short-term Memory (LSTM) network makes use of the temporal information from the tubelet proposals in order to rescore the boxes of each frame. [6] also adopts a different strategy for tubelet classification and rescoring. By contrast, [7, 8, 9, 10] consider temporal information at feature level. [7] exploits warped features to current frame adapted from key frame by predicting motions through optical flow. By this way, non-key frames of the video need not to pass through the feature extraction network. So this method is of great efficiency. However, the performance is even poor than the baseline method by applying image detectors to each frame of videos. This is because we only the information of adjacent frames are used and it is of high possibility that it can’t reflect the changes of current frame. Later, [8, 9, 10] utilizes an aggregation strategy to fuse both current and neighbour frames so as to enhance the feature of current frames. Our method also adopt the idea of feature aggregation to enhance current frame features.

### Table 1. Notations we used in our proposed method

| Notation | Description |
|----------|-------------|
| \( p \) | 2D location |
| \( I_{t-k}, I_t, I_{t+k} \) | video frames |
| \( f_{t-k}, f_t, f_{t+k} \) | intermediate features generated by feature extraction network |
| \( f_{t-k\rightarrow t}, f_{t+k\rightarrow t} \) | warp features |
| \( y_t \) | detection results of frame \( t \) |
| \( N_{feat} \) | sub-network for feature extraction |
| \( s_{insta}^l \) | the feature of roi |
| \( N_{task+} \) | sub-network for detection result |
| \( F \) | flow estimation function |

3. Proposed Method

In this section we show our method that incorporate temporal information at feature level for video object detection.

The main symbols we use are shown in Table 1. Our method can be summarized in three steps. First, the convolutional neural network [13], is applied on each frame of the video clip to get the intermediate feature. Then, our temporal coherence contribution module combines the intermediate feature of current frame and warped features from nearby frames. Note that the warped features from nearby are adopted by the flow network [14]. Finally, the aggregated feature is provided as input to the detection network to get the final detection result. But compared with standard detection network an extra instance-level aggregation module was introduced to improve the feature of the proposals through fusing adjacent features of proposals so as to be more robust to large appearance changes. Our network is integrated in a single framework and can be trained end-to-end. The proposed framework is shown in Figure 2.

3.1. Feature Extraction.

We use the pre-trained ResNet-101 model as our backbone network \( N_{feat} \). The last fully connected layer is abandoned and the stride is taken to half of the original. Other changes made to the original network are the same as [9].
Figure 2. Our proposed framework. The overall framework contains three modules: feature extraction module $N_{feat}$, feature aggregation module $C$ and detection module.

3.2. Feature Aggregation.

The current frame $I_t$ and the adjacent frame $I_{t-k}$ are provided as the input of FlowNet to estimate a flow field $F(I_{t-k}, I_t)$. Then the warped feature from adjacent frame is computed as follows:

$$f_{t-k} = N_{feat}(I_{t-k})$$ (1)

$$f_{t-k} \rightarrow t = W(f_{t-k}, F(I_{t-k}, I_t))$$ (2)

Where $f_{t-k}$ is the intermediate feature extracted from the backbone network $N_{feat}$. $f_{t-k} \rightarrow t$ represents the warped feature maps from frame $t-k$ to frame $t$. $W(\cdot)$ is a bilinear interpolation function that exploited on each location of the maps $f_{t-k}$.

Here we introduce a temporal coherence contribution module. We all know that the closer the two frames in the same video, the more similar they are. So frames from different locations shouldn’t be treated equally. So we design a module that can calculate the coherence contribution of different adjacent frames to the current frame. The temporal coherence contribution reflect the degree of correlation between adjacent frames and current frame. In order not to introduce extra parameters, we don’t add any convolution or activation layers. Specially, the aggregated features mentioned above is first normalized through L2Normalization so as to restrict them to a fixed range. Then the cosine similarity metric is applied to measure the similarity between warped feature $f_{t-k} \rightarrow t$ and current feature $f_t$. Therefore, we can get the contribution of different frames at each location to current frame instead of equal contribution. So, current frame can select useful information from neighbouring frames adaptively according to the real changes of the video. The coherence contribution is computed by the following formula:

$$c_{t-k} = \exp \left( \frac{-f_{t-k}(p) f_t(p)}{||f_{t-k}(p)|| ||f_t(p)||} \right)$$ (3)

where $p$ denotes spatial location of the intermediate feature maps. And we let different channels share the same spatial weight. The contribution for every spatial location are normalized by the following formula:

$$\sum_{j=t-k}^{t+k} c_{j-t} c_{j-t}(p) = 1$$ (4)

Then, the coherence contribution is adopted to get the aggregated feature by formula (5):

$$f_a = \sum_{j=t-l}^{t+t} c_{j-t} f_{j-t}$$ (5)

We believe this coherence contribution module can be exploited to other methods or tasks.
3.3. Detection network.

Our basic detection network is R-FCN, R-FCN is a fully convolutional detector. It achieves excellent performance on both speed and accuracy. Different to standard R-FCN, we add an instance-level aggregation module to the standard detector in order to further enhance the feature of the proposals on the score maps. So $N_{task+}$ is utilized to represent the detection network. This aggregation step is similar to the previous feature aggregation step. Position-sensitive score maps $s_{t-k}, s_t, s_{t+k}$ are produced through convolutional layers. Correspondingly, $s^i_{t-k}, s^i_t, s^i_{t+k}$ represent the $i$-th proposal of these score maps. Then a regression network is introduced to predict the movement between $s^i_{t-k}$ and $s^i_t$, through which $(x^i_{t-k}, y^i_{t-k}, w^i_{t-k}, h^i_{t-k})$ is obtained. Then the feature from $s^i_{t-k}$ can be aggregated to $s^i_t$. Thus the proposal feature of current frame is enhanced. Note that the conduction on $s^i_{t+k}$ is the same as $s^i_{t-k}$.

The regression formula and fusion formulas are (6), (7) respectively.

$$\left(\Delta x^i_{t-k}, \Delta y^i_{t-k}, \Delta w^i_{t-k}, \Delta h^i_{t-k}\right) = R(m^i_{t-k})$$

$$s^i_{insta} = \frac{\sum_{k=t-k}^{t+k} \phi(s_j(x^i_j, y^i_j, w^i_j, h^i_j))}{2k+1}$$

$m^i_{t-k}$ is the warped feature from current frame $t$. It is obtained as follows:

$$m^i_{t-k} = \emptyset \left( F(l_{t-k}, I_t), (x^i_t, y^i_t, w^i_t, h^i_t) \right)$$

Note that $\emptyset(\cdot)$ is the RoI pooling operation. $F(l_{t-k}, I_t)$ is the flow field generated by FlowNet as described in the feature aggregation step. The output of $N_{task+}$ is the final detection result.

4. Experiments

Video datasets is more difficult to collect and annotate compared to image datasets. In our experiment, we exploit two datasets for video object detection: ImageNet VID and ImageNet DET.

4.1 Experiment setup

Our method is evaluated on the ImageNet VID [15] dataset which contains 30 classes. It is split into 3862 training videos (1122397 frames), 555 validation videos (176126 frames) and 937 test videos. We use standard mean average precision (mAP) as our evaluation criteria. The annotations for the test set is not released, therefore we report our final results on the validation set following the standard practice in [7, 8, 9, 10].

4.2 Results and Analysis

To verify the effectiveness of our method we compared our method to several other methods. In our experiment, FlowNet is the default optical flow network and ResNet-101 is the default feature network. We show our results in Table 2.

| Table 2. Accuracy of different methods on ImageNet VID validation |
|-----------------|-----------------|
| methods         | mAP (%)         |
|-----------------|-----------------|
| TPN+LSTM [5]    | 68.4            |
| TCNN [6]        | 73.8            |
| R-FCN [2]       | 70.9            |
| DFF [16]        | 69.9            |
| FGFA [9]        | 76.3            |
| Our method      | 76.5            |

Both TPN+LSTM [5] and TCNN [6] are box-level methods. Since box-level method doesn’t apply during training and can’t be trained end to end, the results of box-level methods we list in the paper is a little bit unsatisfactory. R-FCN is method that applies the still-image detector to each frame of the video. This method doesn’t utilize temporal information exists between video frames. It achieves a mAP of...
Both DFF [16] and FGFA [9] are box-level methods. DFF propagates sparse key frame features to non-key frames. Since it runs feature extraction network only on sparse key frames, it achieves high speed. Correspondingly, the sparse propagated feature cannot reflect the actual changes of current frame so the performance is even poor than R-FCN. Compared with DFF, FGFA applies an aggregation strategy which greatly improve the result. Our method can compute the contribution without introducing any extra parameters, and an instance-level aggregation module is added to the initial detector to capture more global clues. The results from the Table demonstrate the effectiveness of our method.

5. Conclusion and Future Work
In this paper, we introduce an accurate framework that aggregates features across time. The coherence contribution module can adaptively compute the contribution of each adjacent frame to current frame. The combination of the adaptively feature aggregation module and the instance-level aggregation module capture useful information across time. The benefits of our model are proved by the experimental results. In the future work we will attempt to explore a method that can select the number of frames for aggregation can further improve the performance for video object detection.

References
[1] Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. In: NIPS (2015).
[2] Dai, J., Li, Y., He, K., Sun, J.: R-fcn: Object detection via region-based fully convolutional networks. In: NIPS (2016).
[3] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In ECCV, 2016.
[4] W. Han, P. Khorrami, T. Le Paine, P. Ramachandran, M. Babaeizadeh, H. Shi, J. Li, S. Yan, and T. S. Huang. Seq-nms for video object detection. arXiv preprint arXiv:1602.08465, 2016.
[5] Kang, K., Li, H., Xiao, T., Ouyang, W., Yan, J., Liu, X., Wang, X.: Object detection in videos with tubelet proposal networks. In: CVPR (2017).
[6] Kang, K., Li, H., Yan, J., Zeng, X., Yang, B., Xiao, T., Zhang, C., Wang, Z., Wang,R., Wang, X., et al.: T-cnn: Tubelets with convolutional neural networks for object detection from videos. TCSVT (2017).
[7] Feichtenhofer, C., Pinz, A., Zisserman, A.: Detect to track and track to detect. In: ICCV (2017).
[8] Hetang, C., Qin, H., Liu, S., Yan, J.: Impression Network for Video Object Detection. arXiv:1712.05896
[9] Zhu, X., Wang, Y., Dai, J., Yuan, L., Wei, Y.: Flow-guided feature aggregation for video object detection. ICCV (2017)
[10] Wang, S., Zhou, Y., Yan, J., Deng, Z.: Fully Motion-Aware Network for Video Object Detection.ECCV(2018)
[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In NIPS, 2012.
[12] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In ICLR, 2015.
[13] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In Proc. CVPR, 2016
[14] Dosovitskiy, A., Fischer, P., Ilg, E., Hausser, P., Hazirbas, C., Golkov, V., van der Smagt, P., Cremers, D., Brox, T.: Flownet: Learning optical flow with convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2758–2766 (2015).
[15] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International Journal of Computer Vision 115(3), 211–252 (2015)
[16] X. Zhu, Y. Xiong, J. Dai, L. Yuan, and Y. Wei. Deep feature flow for video recognition. In CVPR, 2017.