International Journal of Computer Vision
https://doi.org/10.1007/s11263-021-01507-2

Context and Structure Mining Network for Video Object Detection

Liang Han[1] · Pichao Wang[2] · Zhaozheng Yin[1,3] · Fan Wang[4] · Hao Li[5]

Received: 17 December 2020 / Accepted: 13 July 2021

Abstract

Aggregating temporal features from other frames is verified to be very effective for video object detection to overcome the challenges in still images, such as occlusion, motion blur, and rare pose. Currently, proposal-level feature aggregation dominates this direction. However, there are two main problems for the holistic proposal-level feature aggregation. First, the object proposals generated by the region proposal network ignore the useful context information around the object which is proved to be helpful for object classification. Second, the traditional proposal-level feature aggregation regards the proposal as a whole without considering the important object structure information, which makes the similarity comparison between two proposals less effective when occlusion or pose misalignment occurs on proposal objects. To deal with these problems, we propose the *Context and Structure Mining Network* to better aggregate features for video object detection. In our method, we first encode the spatial-temporal context information into object features in a global manner, which can benefit the object classification. In addition, the holistic proposal is divided into several patches to capture the structure information of the object, and *cross patch matching* is conducted to alleviate the pose misalignment between objects in target and support proposals. Moreover, an importance weight is learned for each target proposal patch to indicate how informative this patch is for the final feature aggregation, by which the occluded patches can be neglected. This enables the aggregation module to leverage the most important and informative patches to obtain the final feature aggregation. The proposed framework outperforms all the latest state-of-the-art methods on the ImageNet VID dataset with a large margin. This project is publicly available https://github.com/LiangHann/Context-and-Structure-Mining-Network-for-Video-Object-Detection.

Keywords Video object detection · Spatial-temporal · Context and structure mining · Cross patch matching

1 Introduction

With the great success of deep neural networks, significant progress has been made on object detection in static images (Girshick 2015; Redmon et al. 2016; Ren et al. 2015; Liu et al. 2016; Dai et al. 2016; He et al. 2017). Nowadays, video-based analysis is becoming more and more popular as the rapid development of 5G and "We Media". However, directly applying those image-based object detectors on a video frame-by-frame often makes the performance unsatisfactory, due to the challenges posed in video capturing, e.g., object occlusion, motion blur, out-of-focus cameras, and rare poses (Fig. 1).

Communicated by Dong Xu.

Equal contribution: Liang Han and Pichao Wang. Liang Han and Zhaozheng Yin have been supported by NSF grants CMMI-1646162 and CMMI-1954548.

✉ Zhaozheng Yin
zyin@cs.stonybrook.edu

Liang Han
liahan@cs.stonybrook.edu

Pichao Wang
pichao.wang@alibaba-inc.com

Fan Wang
fan.w@alibaba-inc.com

Hao Li
lihao.lh@alibaba-inc.com

[1] Department of Computer Science, Stony Brook University, Stony Brook, New York, USA

[2] Alibaba Group, Bellevue, Washington, USA

[3] Department of Biomedical Informatics, Stony Brook University, Stony Brook, New York, USA

[4] Alibaba Group, Sunnyvale, California, USA

[5] Alibaba Group, Hangzhou, Zhejiang, China

Published online: 13 August 2021

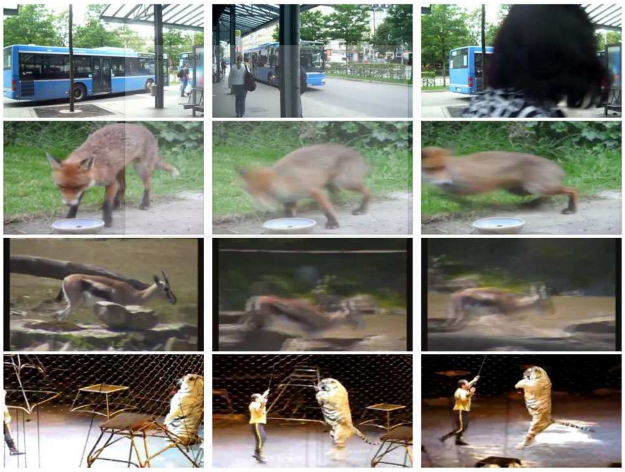


Fig. 1 Challenges in video object detection. First row: part occlusion; second row: motion blur; third row: out-of-focus camera; fourth row: rare poses

It is natural to explore the temporal information inherently encoded in videos to deal with the aforementioned video object detection (VOD) challenges, and this is usually done by building relationship between nearby frames in the video. For example, optical flow is adopted to build correspondences across frames in FGFA (Zhu et al. 2017a) and MANet (Wang et al. 2018a) to conduct feature aggregation, D&T (Feichtenhofer et al. 2017) applies correlation features between nearby frames, STSN (Bertasius et al. 2018) uses deformable convolutions across the temporal domain, and PSLA (Guo et al. 2019) explores the spatial correspondence between features across frames in a local region using progressive sparser strides. In those methods, only local temporal information is used and the long range relation exploration largely depends on some post-processing techniques (Han et al. 2016; Kang et al. 2016, 2017), which are usually not able to be jointly optimized with designed networks, making it sub-optimal. The lack of the capability for long-term temporal exploitation in training makes the performance of these methods degrade in the case of fast motion.

To take advantage of the long-term dependencies between frames, several relation-based feature aggregation models are proposed. Shvets et al. (2019) propose to leverage long-range temporal relationship (LLR) to encode the inter-frame dependencies between object proposals in a long video clip. Wu et al. (2019) introduce the Sequence Level Semantics Aggregation (SELSA) to further explore this long range relation in the whole video sequence level. Deng et al. (2019b) propose the Relation Distillation Networks (RDN) to progressively distill the long range relation. Han et al. (2020a) propose a class-constrained spatial-temporal relation network and a correlation-based feature alignment module for better feature aggregation. To encode both local and global range information, Chen et al. (2020) propose the memory enhanced global-local aggregation (MEGA). Similarly, Jiang et al. (2020) adopt the Learnable Spatial-Temporal Sampling (LSTS) to mine the local motion information, and Sparsely Recursive Feature Updating (SRFU) and Dense Feature Aggregation (DFA) modules to exploit the global temporal information. To exploit the inter-video proposal relations, Han et al. (2020a) introduce the Hierarchical Video Relation Network (HVR-Net), by integrating intra-video and inter-video proposal relations in a hierarchical fashion. However, all of these methods are focused on how to build the relationships across frames, and when it comes to feature aggregation step, each proposal is treated as a whole; instead, we believe that there is important spatial-temporal context information around the objects and the structure information inside the object, which have been ignored in VOD. The spatial context information has been proved to be helpful for static image detection (Kantorov et al. 2016; Chen et al. 2018b), and it is an auxiliary information that can assist suppressing the false positive detection in noisy backgrounds, and recognizing objects that have little distinctive appearances with each other. The structure information of objects is proved to be very important (Sharif Razavian et al. 2015; Gao et al. 2018) for object retrieval to deal with object variances, such as translation, scaling, rotation and occlusion.

To deal with these problems, we propose the *Context and Structure Mining Network (CSMN)* for video object detection. In our method, to explore the context information, each object pixel in the feature map is aggregated with its surrounding pixels in both the spatial and temporal dimension to encode the useful context information. To leverage the object structure information, each object proposal is divided into several non-overlapping patches (9 patches in our experiments). First, instead of directly comparing two holistic proposals, we use divide-and-match strategy to alleviate the pose misalignment between two object proposals, which gives us a better similarity measurement of these two proposals. Then, for each patch, an importance weight is learned from the feature of this patch to indicate its importance for the final feature aggregation. With these importance weights, different patches play different important roles in aggregating final features, and the occlusion problem can be mitigated by focusing more on those non-occluded patches when aggregating features. Through the divide-and-match process, the structure information of the object is captured to deal with the occlusion and pose misalignment challenge, which is demonstrated to be able to benefit the final regression and classification.

The main contributions of this work are summarized as follows:

- We exploit the non-local network (Wang et al. 2018b) to design a context information encoding module to encode the useful *context information* into the object features for more accurate object detection, in which we extend the

original non-local network into the spatial and temporal dimension and fix the position of where we should perform the context information encoding.
- The *structure information* of objects is exploited by using a divide-and-match strategy to deal with the object pose misalignment and occlusion problems, which is able to aggregate more informative and supportive features for target proposals.
- The proposed framework achieves much better results on ImageNet VID dataset.

2 Related Work

In this section, we briefly review the object detection from the image and video perspectives.

2.1 Object Detection in Static Images

Till the present, there are mainly two branches for static image object detection: one-stage object detector and two-stage object detector. In the one-stage detector, the bounding box of interest is directly predicted based on the extracted feature map from CNN, such as YOLO (Redmon et al. 2016), YOLO9000 (Redmon and Farhadi 2017), YOLOV3 (Redmon and Farhadi 2018), SSD (Liu et al. 2016), DSSD (Fu et al. 2017) and FCOS (Tian et al. 2019). Compared with two-stage object detectors, one-stage object detectors are with fast inference speed. However, one-stage object detectors are more likely to lead to foreground and background class imbalance problem, and affect the training process and accuracy (Lin et al. 2017b). Instead, two stage detectors usually generate region proposals first, with the majority of negative locations filtered out, and then the proposals are refined by the classification and regression through the Regions with Convolutional Neural Networks (R-CNN) stage (Girshick et al. 2014). Faster R-CNN (Ren et al. 2015) proposes Region Proposal Network (RPN) to generate region proposals. R-FCN (Dai et al. 2016) replaces the ROI pooling on the intermediate feature maps with position-sensitivity ROI pooling on the final score maps. Feature Pyramid Networks (FPN) (Lin et al. 2017a) brings an inherent multiscale, pyramidal hierarchy of deep convolution networks to build feature pyramids. Mask RCNN (He et al. 2017) proposes the ROI align operation to replace ROI pooling to further improve the detection accuracy. To explore the appearance and geometry relations among object proposals within a still image, relation networks (Hu et al. 2018) and non-local neural networks (Wang et al. 2018b) are proposed, which enable the detector to reason the topological relations of objects and improve the performance. Our work adopts the idea of the two-stage object detector and the relation networks to exploit the relations in both spatial-temporal domain. However, our work targets to encode the context information in the video frames and the structure information contained in the object proposals to improve the accuracy of detector in video object detection.

2.2 Video Object Detection

There are two mainstream approaches for video object detection. In the first approach, the redundancy in video frames is leveraged to improve the detection speed. For example, optical flow is adopted by (Zhu et al. 2017b, 2018) to propagate the key frame feature to other frames to save the expensive feature extraction cost. A time-scale lattice is designed by (Chen et al. 2018a) to improve the speed with an extra classifier to re-score the bounding boxes. Liu and Zhu (2018); Liu et al. (2019) adopt Bottleneck-LSTM with MobileNet (Howard et al. 2017; Sandler et al. 2018) as the backbone and use SSD as the detector to improve the speed on the mobile devices. Similarly, Jiang et al. (2019) adopt brain-inspired memory mechanism to propagate and update the memory feature from keyframes to keyframes, and propose the locally-weighted deformable neighbors to align the high-level features between keyframes and non-keyframes. Yao et al. (2020) adopt object tracker for temporal propagation, and using reinforcement learning for adaptive key-frame scheduling. Xu et al. (2020) propagate the previous reliable long-term detection in the form of heatmap to boost results of upcoming image for one-stage detector.

In the second approach, temporal information encoded in videos is explored to improve the performance of the detection, and our paper belongs to this approach. In the second approach, there are two major branches. The first branch is focused on post processing (Han et al. 2016; Kang et al. 2016, 2017). These methods usually take the spatial and temporal coherence into consideration, and explore bounding box association rules across nearby frames to refine the per-frame detection results. Those methods are sub-optimal because they are highly dependent on the quality of initial detector which is trained without any temporal information. In contrast, the other category of methods (Feichtenhofer et al. 2017; Zhu et al. 2017a, b; Chen et al. 2018a; Wang et al. 2018a; Xiao and Jae Lee 2018; Zhu et al. 2018; Bertasius et al. 2018; Deng et al. 2019a; Guo et al. 2019; Deng et al. 2019b; Shvets et al. 2019; Wu et al. 2019; Chen et al. 2020) directly exploits the temporal information in videos during the training stage. Among these methods, optical flow based feature warping (Dosovitskiy et al. 2015) is widely used to propagate the features across frames (Zhu et al. 2017a, b; Wang et al. 2018a). However, the optical flow module here significantly increases the overall model size of detectors, and it only exploits the temporal information between frames in short time range, and the warping does not works well in occlusion.

To address these shortcomings, Guo et al. (2019) introduce PSLA to model the spatial correspondence between features across frames in a local region using the progressive sparser stride, and Jiang et al. (2020) proposes the Learnable Spatial-Temporal Sampling (LSTS) to mine the local motion information. To explore the long-range dependencies in the temporal domain, Xiao and Jae Lee (2018) propose a spatial-temporal memory networks (STMN) as the recurrent operation to model long-term temporal appearance and motion dynamics, with a MatchTrans module proposed to align the spatial-temporal memory. Shvets et al. (2019) propose to use the relation module (Vaswani et al. 2017) to model the inter-frame dependencies between the object proposals in a long video segment, Wu et al. (2019) further explore the temporal relation across the whole sequence, Deng et al. (2019b) propose the RDN to model the spatial-temporal relations for video object detection, Han et al. (2020a) adopt a class-constrained spatial-temporal relation network and a correlation-based feature alignment module for better feature aggregation, Chen et al. (2020) further exploit both the global and local relationships between object proposals, and Han et al. (2020b) propose the HVR-Net by integrating intra-video and inter-video proposal relations in a hierarchical fashion. The six works (Shvets et al. 2019; Wu et al. 2019; Deng et al. 2019b; Han et al. 2020a, b; Chen et al. 2020) achieved promising results on video object detection. However, they are all holistic proposal-based feature aggregation scheme, and the context and structure information contained in the objects are overlooked. Previous works in image detection (Kantorov et al. 2016; Chen et al. 2018b) and image retrieval (Sharif Razavian et al. 2015; Gao et al. 2018) verified that the context and structure information are very important to deal with object variances in noisy background, such as translation, scaling, rotation and occlusion.

To deal with these problems, we propose to exploit the spatial-temporal context contained in the video frames and structure information in the proposals for better video object detection.

3 Proposed Method

Fully encoding spatial-temporal context information and better aggregating temporal information of objects from neighboring frames are the keys for detecting object in the current video frame. In this section, we introduce the details of the proposed framework called Context and Structure Mining Network (CSMN), which consists of spatial-temporal context information encoding module and structure-based object feature aggregation module, for video object detection.

3.1 Overview

The pipeline of the proposed framework is depicted in Fig. 2. Target frame is the frame where final object detection is performed at the moment, and support frames are the frames in the same video and are selected to provide additional information for the target frame. First, a backbone network (ResNet-101 in most of our experiments) is adopted to extract features for the target frame and the support frames, and the extracted frame features are fed into two modules: a Region Proposal Network (RPN) which is used to generate target proposals and support proposals, and a spatial-temporal Context Information Encoding (stCIE) module to encode context information into the features of the objects in the target and support frames. Then, with the location and bounding box shape information of the generated target and support proposals, a ROI pooling operation is performed on the context information encoded frame features to extract feature for each proposal. After that, the target proposals and the support proposals are thrown into a Structure-based Proposal Feature Aggregation (SPFA) module, which aggregates target proposal features with the support proposal features. Finally, the aggregated target proposal features are used to perform the final detection (i.e., classification and location regression).

3.2 Spatial-temporal Context Information Encoding

The spatial context information in a frame is helpful for static image object detection (Kantorov et al. 2016; Chen et al. 2018b). It is an auxiliary information that can assist in recognizing and classifying objects that have little distinctive appearances from the background or from other kind of objects. Accordingly, in this subsection, we develop a spatial-temporal Context Information Encoding (stCIE) module to encode the spatial-temporal context information into the object feature for better object detection. We borrow the idea of the non-local network (Wang et al. 2018b) for our proposed stCIE with the following modifications: (a) we extend the original non-local network into the spatial and temporal dimension, (b) we fix the position of where we should use the stCIE module to perform the context information encoding, i.e., on the $conv5$ feature. Specifically, we first adopt a Region Proposal Network (RPN) on the original $conv4$ feature extracted by the backbone network without encoding the context information to generate ROIs. Then, we perform context information encoding on the $conv5$ feature maps extracted by the backbone network, and project the generated ROIs onto the context information encoded $conv5$ featue maps to extract features for each proposal. Under these modifications, the background pixel features in $conv4$ will not be contaminated by the object information, and thus it is easier for RPN to filter out the background proposals from

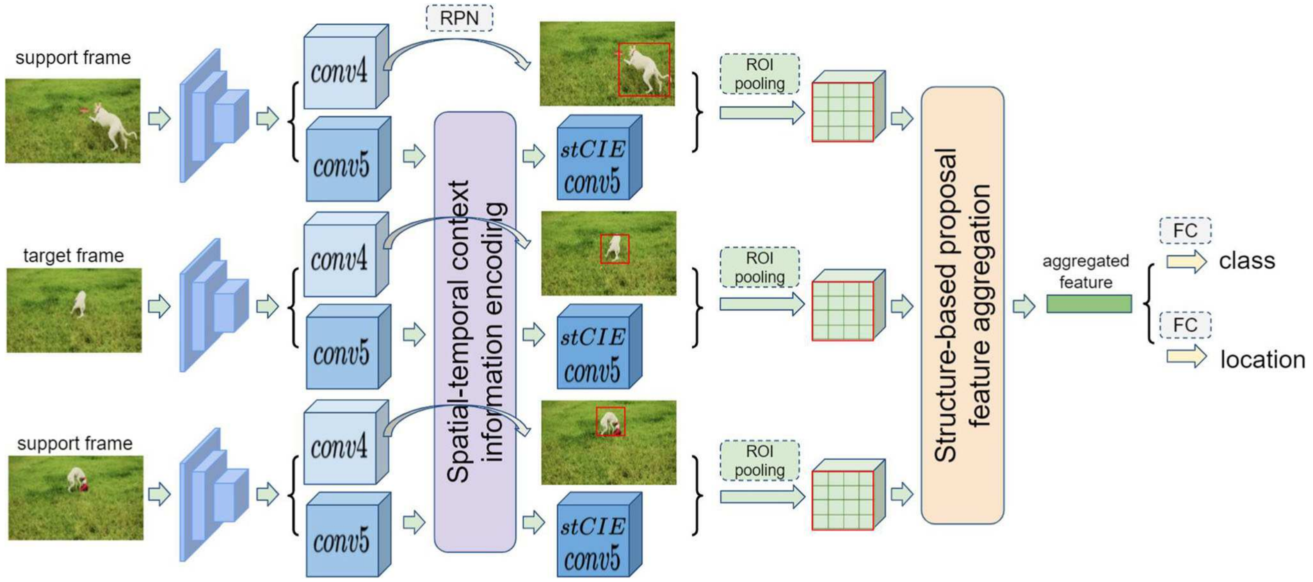


Fig. 2 Pipeline of the proposed framework. First, ResNet-101 is used to extract features for the target and support frames, and the extracted $conv4$ feature maps are fed into a RPN to generate target proposals and support proposals; while the extracted $conv5$ feature maps are fed into a spatial-temporal Context Information Encoding (stCIE) module to encode context information. Then, with the location and shape information of the generated proposals, a ROI pooling operation is performed on the context information encoded $conv5$ feature maps to extract features for each target and support proposals. After that, the target and support proposals are put into a Structure-based Proposal Feature Aggregation (SPFA) module to aggregate target proposal features with the support proposal features. Finally, the aggregated target proposal feature is used to perform object detection

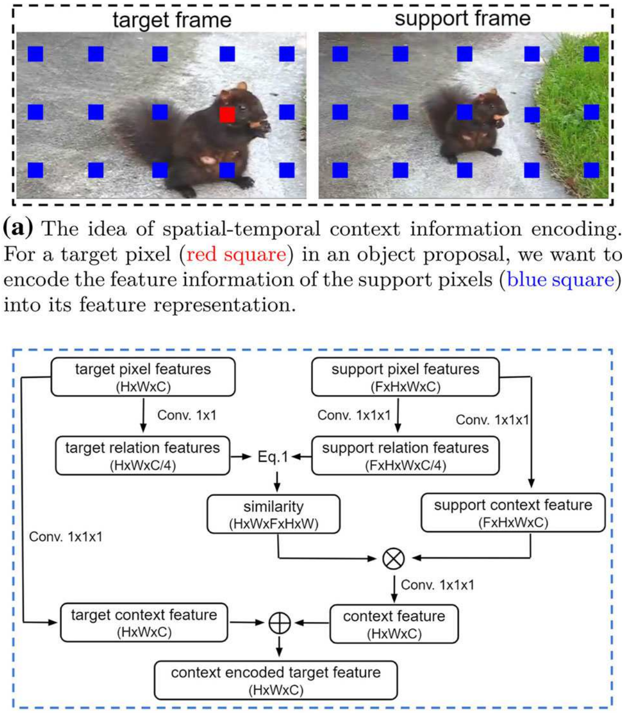
the object proposals, which will further benefit the target proposal feature aggregation by keeping the purity of proposals.

Figure 3a explains the basic idea of the spatial-temporal context information encoding. A pixel in an object proposal is called ***target object pixel*** (the red square), and all the pixels in the target frame and support frames except for the target object pixel are called ***support pixels*** (the blue squares). For a certain target object pixel, the support pixels can be object pixels and background pixels. Thus, the spatial-temporal context information is included in the support pixel features. We want to encode the spatial-temporal context information carried in the support pixel features into the target pixel feature by exploring the relation between the target object pixel and each of the support pixels. It is worth noting that a pixel in the extracted feature map corresponds to a patch in the original image. Therefore, when performing the context information encoding in the pixel level of the feature map, we actually encode the context information in the patch level of the original image.

The approach of exploring the relations between the target object pixel and the support pixels is motivated by the great success of the attention mechanism in natural language processing (Vaswani et al. 2017) and computer vision (Hu et al. 2018) community, which is able to well capture the complex relations between independent units (e.g., words, proposals, etc.). Figure 3b presents how to capture the relations between a target object pixel and its corresponding support pixels, and how to encode the support pixel information into the target object pixel based on the captured relations. More precisely, for a target object pixel p^t and a support pixel p^i, a 1×1 convolution is applied on them to generate the target content feature f_{ctnt}^t and the support content feature f_{ctnt}^i, respectively. Another 1×1 convolution is then applied on these two kinds of pixels to output the target relation feature f_{rela}^t and the support relation feature f_{rela}^i, respectively. After obtaining the relation features, the relation weight $r^{t,i}$ between the target object pixel p^t and the support pixel p^i is computed as

$$r^{t,i} = \frac{exp\Big(cos(f_{rela}^t, f_{rela}^i)\Big)}{\sum_{i=1}^{I} exp\Big(cos(f_{rela}^t, f_{rela}^i)\Big)}, \tag{1}$$

where $cos(\cdot, \cdot)$ is the cosine similarity of two vectors, and I represents the set of all support pixels for this target object pixel. The context information carried in the support pixels is then summarized together based on the calculated relation weights and added back onto the original target content feature to generate the context information encoded target pixel


target frame
support frame
target pixel features (HxWxC)
support pixel features (FxHxWxC)
Conv. 1x1
Conv. 1x1x1
Conv. 1x1x1
target relation features (HxWxC/4)
Eq.1
support relation features (FxHxWxC/4)
similarity (HxWxFxHxW)
support context feature (FxHxWxC)
Conv. 1x1x1
⊗
Conv. 1x1x1
target context feature (HxWxC)
⊕
context feature (HxWxC)
context encoded target feature (HxWxC)

(a) The idea of spatial-temporal context information encoding. For a target pixel (red square) in an object proposal, we want to encode the feature information of the support pixels (blue square) into its feature representation.

(b) The implementation of spatial-temporal context information encoding. ⊗ is the matrix multiplication, and ⊕ is the element-wise summation.

Fig. 3 The idea and detailed implementation of the proposed spatial-temporal Context Information Encoding (stCIE) module (Color figure online)

feature $f^t_{ctnt+ctxt}$:

$$f^t_{ctnt+ctxt} = f^t_{ctnt} + \sum_{i=1}^{I} r^{t,i} \cdot f^i_{ctnt}. \tag{2}$$

3.3 Structure-based Proposal Feature Aggregation

Leveraging the temporal information of objects from neighboring frames to aggregate the target object feature is proven to be an effective strategy for more accurate video object detection (Chen et al. 2020; Wu et al. 2019; Deng et al. 2019b; Shvets et al. 2019; Wang et al. 2018a; Zhu et al. 2017a). Usually, the support proposal features are weighted aggregated onto the target proposal feature based on the calculated similarities between the target proposal and the support proposals. *Unfortunately, it is error-prone to directly measure the similarities of holistic proposals due to the challenges in videos such as occlusion and pose misalignment of objects.* For example, when the target object proposal is partially occluded (e.g., the first column in Fig. 4), though the support proposal in the second column of Fig. 4 is very informative and supportive (i.e., it can compensate the missing information of the target proposal), the similarity between this support proposal and the target proposal is small because the extracted feature of the target proposal is contaminated by the occlusion. As a result, the target proposal feature might be overwhelmed by the proposal features of objects in other class or the background. The same thing happens for object proposals that are not aligned (e.g., the third column and fourth column in Fig. 4).

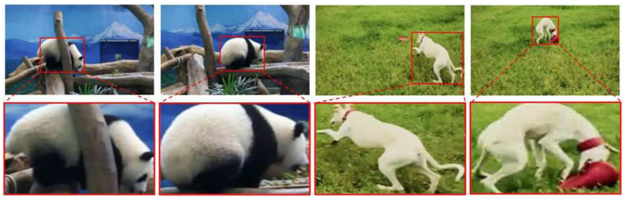


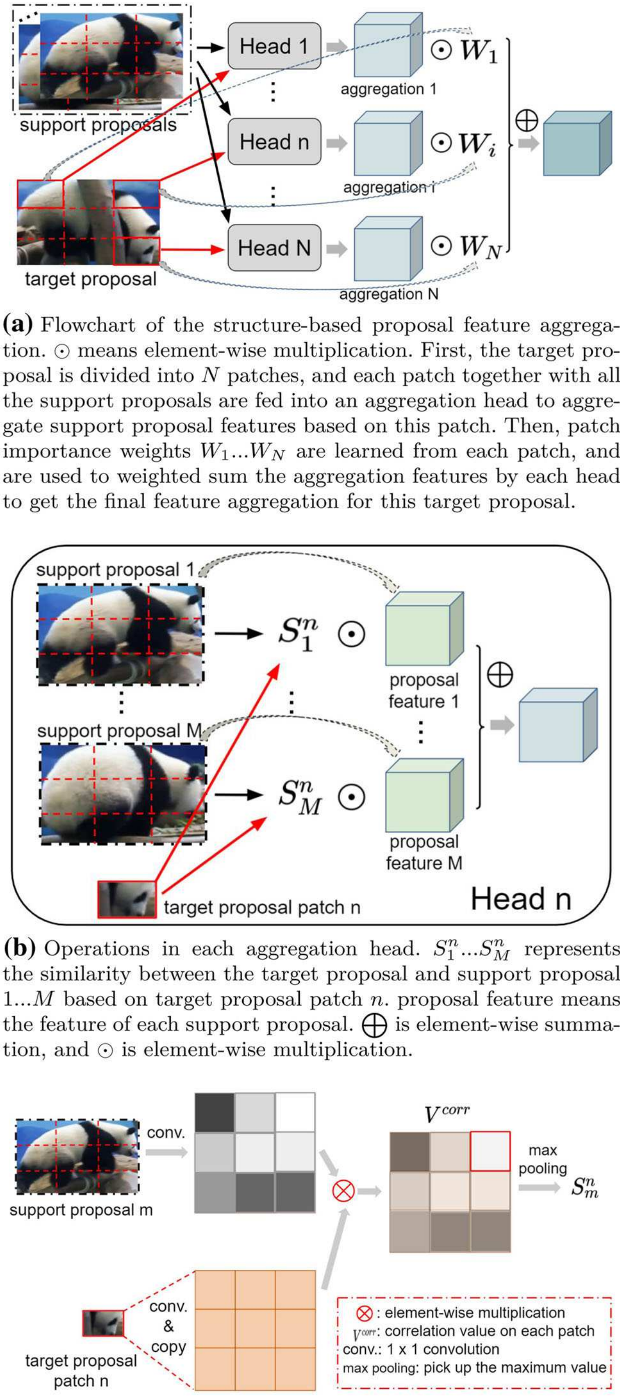
Fig. 4 Challenges in the video which harm the accurate measurement of the similarity between the target proposal and support proposal

To overcome these challenges, we propose a Structure-based Proposal Feature Aggregation (SPFA) module which can better aggregate the target proposal feature by exploiting its structure information in the target proposal. Figure 5a depicts the basic idea of the proposed SPFA. Let M denote the number of support proposals generated from the sampled support frames. First, we divide a target proposal into N ($N = 9$ in our experiments) non-overlapping proposal patches, and each target proposal patch and the M support proposals will go into an aggregation head (***Head*** 1 ... ***Head*** N in Fig. 5a), which aggregates the support proposal feature based on this target proposal patch. Figure 5b presents the detailed feature aggregation operation in each aggregation head. Let f^{prop}_m denote the feature of the m-th support proposal, the similarity S^n_m between the target proposal and the support proposal m based on the target proposal patch n is calculated (Fig. 5c shows how to calculate the similarity, which will be introduced in details later), and the feature f^n_{agg} aggregated by ***Head*** n is

$$f^n_{agg} = \sum_{m=1}^{M} S^n_m \cdot f^{prop}_m, \tag{3}$$

where $n \in [1 \ldots N]$ denotes the head index, and $m \in [1 \ldots M]$ is the support proposal index.

Note that for each patch in a target proposal, we can get an aggregated feature with each aggregation head, and in total N aggregated features are calculated for this target proposal based on its N patches ($aggregation$ 1 ... $aggregation$ N in Fig. 5a). As some proposal patches are informative to represent the object in this proposal (e.g., patches that are object body parts), while some are not (e.g., patches which are heavily occluded), we need some weights to select the most informative proposal patches and use the aggregated feature obtained with these patches to compensate the target proposal feature. Thus, different from the tradi-



(a) Flowchart of the structure-based proposal feature aggregation. $\odot$ means element-wise multiplication. First, the target proposal is divided into N patches, and each patch together with all the support proposals are fed into an aggregation head to aggregate support proposal features based on this patch. Then, patch importance weights $W_1 ... W_N$ are learned from each patch, and are used to weighted sum the aggregation features by each head to get the final feature aggregation for this target proposal.

(b) Operations in each aggregation head. $S_1^n ... S_M^n$ represents the similarity between the target proposal and support proposal $1...M$ based on target proposal patch n. proposal feature means the feature of each support proposal. $\bigoplus$ is element-wise summation, and $\odot$ is element-wise multiplication.

(c) Similarity computation. The similarity S_m^n is between the target proposal and a support proposal m based on target proposal patch n.

Fig. 5 Structure-based proposal feature aggregation. (**a**) shows the basic idea of the SPFA, (**b**) presents the detailed feature aggregation in each head, and (**c**) introduces how to better measure the similarity between two proposals with cross patch matching

tional feature aggregation module, in our proposed SPFA module, N ***patch importance weights*** (W_1 ... W_N in Fig. 5a) are learned from the corresponding N original target proposal patch features with a fully connected layer, followed by a $softmax$ operation to normalize these weights. These importance weights indicate how important each target proposal patch is for the final feature aggregation. The final aggregated feature of the target proposal is obtained by weighted adding up the features aggregated by the aggregation heads and with patch importance weights as the adding weights

$$f_{agg}^{final} = \sum_{n=1}^{N} W_n \cdot f_{agg}^{n} = \sum_{n=1}^{N} \Big(W_n \cdot \sum_{m=1}^{M} (S_m^n \cdot f_m^{prop}) \Big). \quad (4)$$

With the learned patch importance weights, the heavily occluded parts can be ignored when searching for support proposal features, which further alleviate the influence of the occlusion. These N learnable patch weights enable the SPFA module use the most informative target object parts to search for the compensatory object feature for target proposal feature aggregation.

Note that an alternative but straightforward way to obtain the patch weights is simply using equal importance weights for all the patches of the target proposal, i.e., if we divide the target proposal into N patches, the importance weight of each patch is $\frac{1}{N}$. We treat this simple equal importance weight way to aggregate patch feature as a baseline, and compare it with our proposed learnable patch weights in the experiments.

To calculate the similarities S_1^n ... S_M^n between the target proposal and support proposals 1 ... M based on target proposal patch n in Fig. 5b, the cross patch matching strategy is adopted. Figure 5c gives a detailed illustration of how to calculate these similarities (S_m^n is taken as an example to show the computation process). For a certain target proposal patch, the correlations between this target proposal patch and each of the patch in support proposal m are calculated. Calculating the correlation between proposal patches instead of the whole object proposal can mitigate the influence of the occlusion in the maximum extent. After that, the maximum correlation value is picked out of the correlation matrix as the similarity between this target proposal and the support proposal based on this target proposal patch. By selecting the maximum correlation value, the most related object parts in the target proposal and the support proposal can be found and compared to calculate the similarity between these two proposals, which means that the misalignment problem can be alleviated. Mathematically, the similarity score S_m^n between the target proposal and support proposals m based on target proposal patch n is calculated as

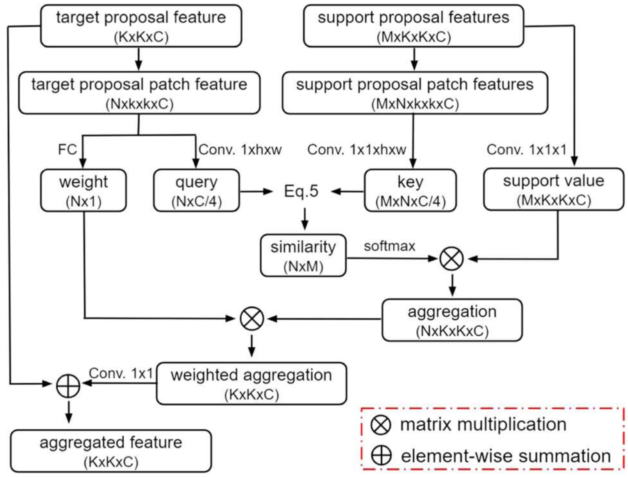


Fig. 6 Detailed implementation of the structure-based proposal feature aggregation. N is the number of non-overlapping patches generated from each proposal, M denotes the number of support proposals, K and C are the spatial size and feature channel dimension of the object proposals after RoIAlign pooling

$$\begin{aligned} S_m^n &= \max_{j=1...N} S_m^{n,j} \\ &= \max_{j=1...N} corr(p_t^n, p_m^j) \\ &= \max_{j=1...N} \frac{\sum_{d=1}^{D}(p_t^n(d) - \overline{p_t^n})(p_m^j(d) - \overline{p_m^j})}{var(p_t^n)\cdot var(p_m^j)} \end{aligned} \tag{5}$$

where p_t^n denotes the feature of the target proposal patch n, p_m^j denotes the feature of the patch j in support proposal m, $p_t^n(d)$ denotes the d-th dimension of feature p_t^n, $\overline{p_t^n}$ and $var(p_t^n)$ denote the mean and variance of feature p_t^n, respectively. The similarity scores $\{S_m^n\}$ $(m \in [1, ...M])$ between the target proposal and support proposal m based on target proposal patch n are calculated with Eq. 5, and a $softmax$ operation is performed to normalize the similarity scores before using them as summation weights in Eq. 3.

The detailed implementation of the proposed SPFA is shown in Fig. 6, where N is the number of non-overlapping patches generated from each proposal, M denotes the number of support proposals, H, W and C are the height, width and feature channels of the object proposals after RoIAlign pooling.

4 Experiments

We implement our method based on the source codes of SELSA (Wu et al. 2019). In the following, we briefly describe the details of the backbone network, region feature extraction network, dataset, evaluation metric and training&testing settings. After that, we evaluate the effectiveness of the proposed context information encoding module and the structure-based proposal feature aggregation module. Finally, the comparison with state of the art is performed.

4.1 Network Implementation

Similar to most previous VOD works, we select the ResNet-101 (He et al. 2016) as the backbone to perform feature extraction for each video frame, and the Region Proposal Network (RPN) (Ren et al. 2015) is applied on the $conv4$ frame feature to generate object proposals for the target and support frames. During training and inference, anchors are set with 3 different scales and 3 different aspect ratios, and in total 9 different kinds of anchors are used in RPN to first generate 6000 proposals with the highest objectness scores for each video frame. After that, the Non-Maximum Suppression (NMS) is performed on these 6000 proposals to finally keep 300 object proposals for each frame. Finally, RoI pooling is performed on the context information encoded frame feature, instead of the original $conv5$ frame feature, to extract feature for each of the 300 object proposals.

4.2 Dataset and Evaluation Metric

We select the ImageNet DET and VID datasets (Russakovsky et al. 2015), which are the most widely-used datasets for the VOD task, to train and evaluate our proposed framework. Specifically, we first get the intersection of these two datasets by picking out the 30 object classes they shared in common to train our proposed CSMN model. The validation set of ImageNet VID is used to evaluate the performance of the proposed model. We set the training/validation split as in (Zhu et al. 2017a). Thus, the training and evaluation are conducted on the 3,862 video snippets from the training set and the 555 snippets from the validation set, respectively. The snippets are fully annotated, and are at frame rates of 25 or 30 fps in general. For better analysis, following (Zhu et al. 2017a), according to the motion speed, the ground truth objects are categorized to slow, medium and fast motion. The object speed is measured by its averaged intersection-over-union (IoU) scores with its corresponding instances in the nearby frames (±10 frames), and we denote it as ”motion IoU”. The lower the motion IoU is, the faster the object moves. According to the score, the objects are divided into slow (score $\in$ (0.9, 1.0]), medium (score $\in$ [0.7, 0.9]), and fast (score $\in$ [0.0, 0.7)) groups, respectively. In evaluation, besides the standard mean average-precision (mAP)(@IoU=0.5) scores, we also report the mAP scores over the slow, medium, and fast groups, respectively, denoted as mAP(slow), mAP(medium), and mAP(fast). This provides us a more detailed analysis and in-depth understanding.

Table 1 Ablation study on proposed spatial-temporal Context Information Encoding (stCIE) module and Structure-based Proposal Feature Aggregation (SPFA) module

method	Baseline	Baseline + stCIE	Baseline + SPFA	Baseline + stCIE + SPFA
mAP(%)	82.7	83.7	84.3	85.2
mAP(%) slow	88.9	89.7	90.2	90.8
mAP(%) medium	81.2	82.5	83.2	84.2
mAP(%) fast	65.4	67.9	69.3	70.5

We take SELSA (Wu et al. 2019) as the baseline. mAP slow/medium/fast are the detection precision for objects with slow motion/medium motion/fast motion

4.3 Training and Inference

We train the proposed framework on 8 V100 GPUs for a total of 10 epochs with a SGD optimizer. The backbone network is first initialized with the weights pre-trained on ImageNet classification task, then all modules in the framework (backbone, stCIE, SPFA and final detection layers) are trained and optimized simultaneously. We set the batch size as 8 with each minibatch is allocated to one GPU. An initial learning rate of $2.5e^{-4}$ is established, which is 10 times less after 4 epochs, and decreases again after another 4 epochs. During inference, for every target frame, we establish the support frames by randomly sampling T frames from the same video sequence. All of the video frames are resized to be with shorter dimension of 600 pixels both for training and inference. As our proposed CSMN is implemented based on SELSA (Wu et al. 2019), the same training protocol from SELSA is applied, i.e., the same data augmentation strategy in (Wu et al. 2019) is adopted to train our CSMN model.

4.4 Quantitative Ablation Study on Proposed Modules

In this subsection, we perform some ablation experiments to show the effectiveness of the proposed ***spatial-temporal context information encoding (stCIE)*** module and the ***structure-based proposal feature aggregation (SPFA)*** module. As the proposed method is implemented over the source conde of SELSA (Wu et al. 2019), we take the SELSA network as a baseline. First, the video object detection performance of the baseline network is evaluated. Then, we add the stCIE module into the baseline network (Baseline+stCIE) and evaluate the object detection performance. After that, the SPFA module is inserted into the baseline network (Baseline+SPFA) and perform the detection. Finally, both the stCIE module and the SPFA module are put into the baseline network (Baseline+stCIE+SPFA) to detect the objects in videos. The mAP results of each experiment are reported in Table 1, from which we can get the following observations: (1) *The spatial-temporal context information encoding brings us a +1.0% mAP improvement* compared with the baseline network (i.e., SELSA). This is because some spatial-temporal context information is aggregated into the object proposal feature, which enables the classifier distinguish some objects with confusing feature representations more easily. Moreover, the stCIE module can enhance the object features in pixel level, which makes the object features more distinctive from each other. (2) *The structure-based proposal feature aggregation strategy makes a contribution of +1.6% mAP* to the detection improvement. The reason is that the SPFA module is able to better aggregate target proposal feature by exploiting the object structure in the target proposal. By comparing the similarity between object parts instead of two whole objects, the influence of the occlusion and misalignment on similarity computation is greatly alleviated, and the most informative and supportive proposals are searched to aggregate the target proposal feature. It is worth noting that the baseline, SELSA, also adopts the proposal level feature aggregation, but without considering the structure information. In this work, we improve the proposal level feature aggregation by exploiting the structure information of the objects, which is able to better aggregate target proposal features, especially for object proposals with occlusions. Thus, the improvement of +1.6 % mAP brought by the SPFA module is actually the contribution of exploiting the structure information in the proposal feature aggregation. When there is no proposal level feature aggregation in the detection model, i.e., only Faster-RCNN + stCIE, the detection precision is only about 77 % mAP, which is much worse than the result of the proposed CSMN. (3) *These two modules (stCIE and SPFA) together make a +2.5% mAP improvement*, which also demonstrates that these two modules are not conflicting on improving object detection performance. (4) *Compared with the stCIE module, the SPFA is more effective for detecting objects with fast motion*. The reason is that compared with slow motion objects, objects with fast motion are more easily occluded. Also, fast motion objects have more various poses, which increases the misalignment between objects in different frames. The stCIE encodes the context information at the pixel level, which can benefit the classification of the proposal by leveraging the surrounding context information. However, the stCIE can not effectively

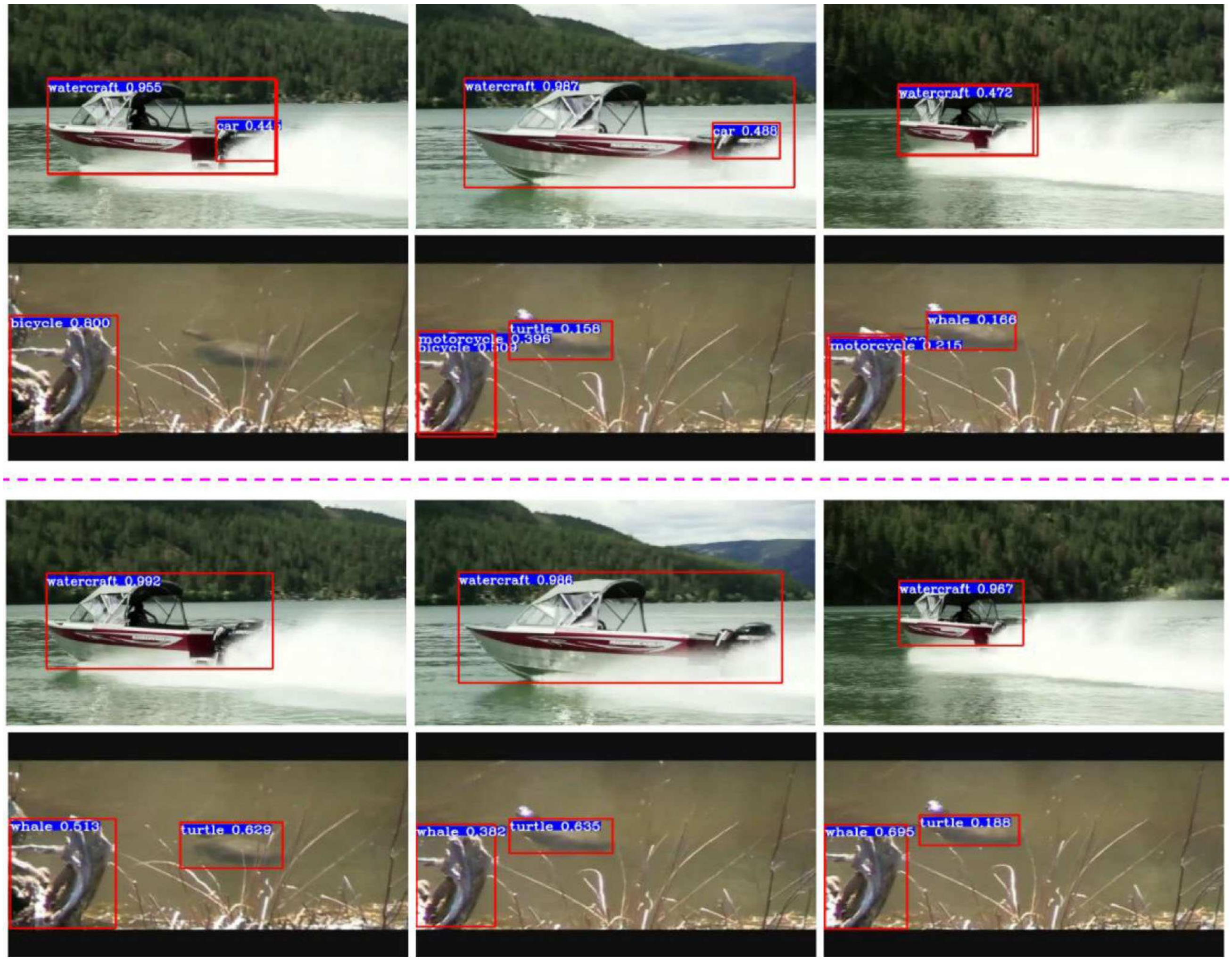


Fig. 7 Qualitative ablation study on the proposed stCIE module. First two rows: results of the proposed detection model without the stCIE module. Last two rows: results of the proposed detection model with the stCIE module

overcome the challenges such as occlusion and misalignment, which are fairly common in the video, especially for fast-moving objects. While the SPFA module is specifically designed to deal with these problems, which makes it more effective for fast motion object detection.

4.5 Qualitative Ablation Study on Proposed Modules

Figure 7 shows the ablation study on the proposed stCIE module. The top two rows and bottom two rows shown the results of the proposed detection model without the stCIE module and with stCIE module, respectively. In the first example (1st row vs. 3rd row), when context information encoding is performed, the detection model successfully eliminates the label of "car" with the aid of the context information (the surrounding water). In the second example (2nd row vs. 4th row), the context information (the water) helps the detection model to consistently detect the turtle correctly. Besides, even though the model falsely detects a piece of wood, the classification is more reasonable by assigning a label of "whale" with the help of the context information, compared with the label of "bicycle" or "motorcycle" in the results without stCIE. From these two examples we can see that the proposed stCIE module improves the detection model by using the surrounding context information.

The ablation study on the proposed SPFA module is visualized Fig. 8. The top two rows and bottom two rows are the results of the proposed detection model without the SPFA module and with the SPFA module, respectively. From the first example (1st row vs. 3rd row) we can see that with an object partially occluded, the proposed SPFA module helps the detection model to correctly detect this occluded object with higher confidence scores. The second example (2nd row vs. 4th row) shows that the SPFA helps eliminating misclassification of the partially occluded objects. These two examples demonstrate that the proposed SPFA module can better deal

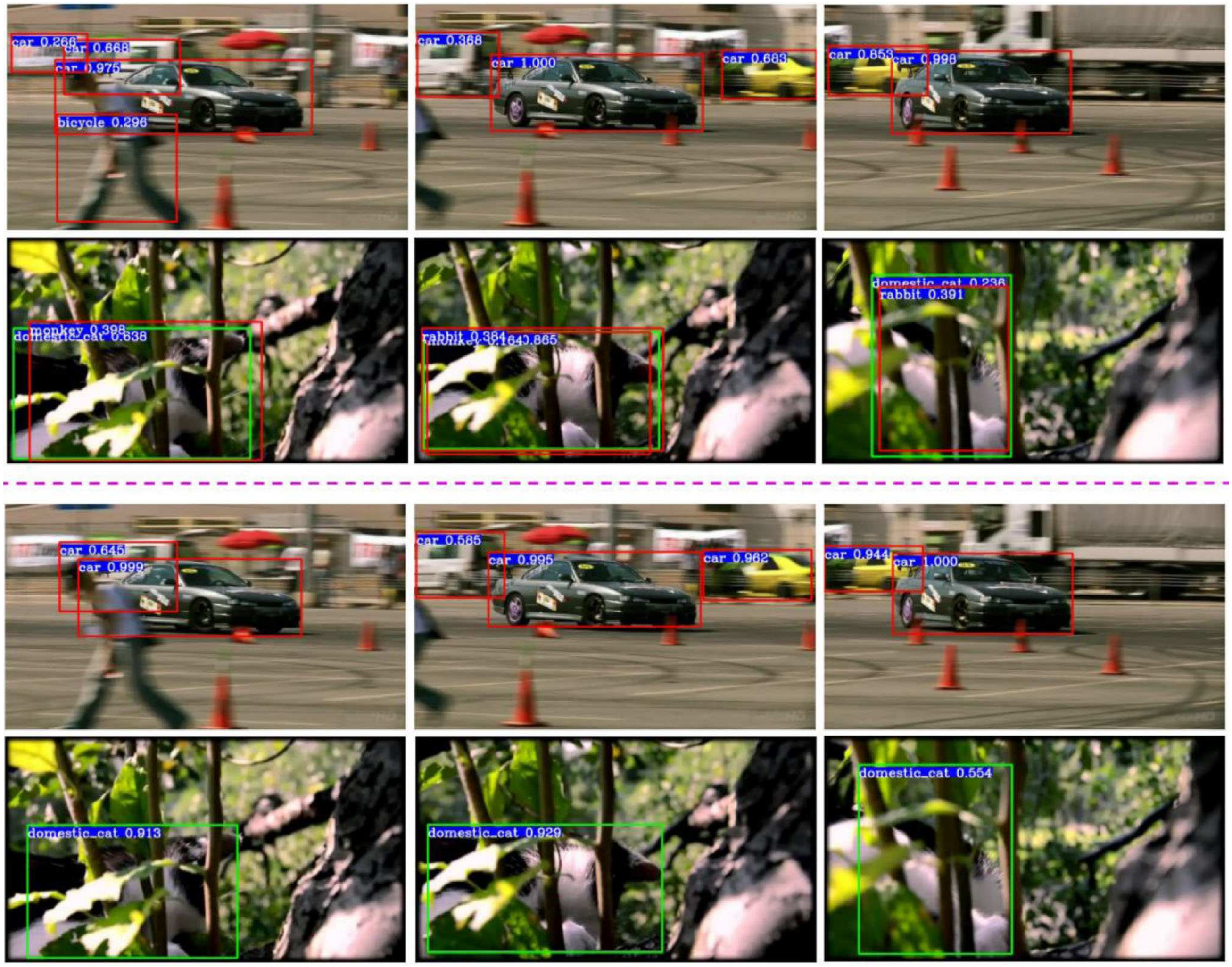


Fig. 8 Qualitative ablation study on the proposed SPFA module. First two rows: results of the proposed detection model without the SPFA module. Last two rows: results of the proposed detection model with the SPFA module

with the occlusion challenge in the video object detection by performing feature aggregation with the object structure information.

4.6 Ablation Study on stCIE

We then dive deeper into the proposed stCIE module by separating the context information encoding along the spatial and temporal dimensions, i.e., only the spatial context information ('sCIE' in Table 2) is encoded in the current target frame, to check the efficiency of the spatial context and temporal context in the proposed stCIE. From the comparison results in Table 2 it can be concluded that compared with the baseline [SELSA Wu et al. (2019)], performing the context information encoding along the spatial dimension can improve the detection accuracy (i.e., 'SELSA + sCIE' in Table 2, context information encoded for the objects with the context information in this single frame). This is because the context information in this frame can help the detection model better classify the objects (some examples can be found in Fig. 7). When we encode both the spatial and temporal context information for the object pixels (i.e., 'SELSA + stCIE' in Table 2), the final detection accuracy is further improved compared with that of only encoding the spatial context information. The reason is that for some video frames with motion blur or out-of-focus scene, the spatial context information in the current frame may be obscure and not very helpful for object classification. By encoding the temporal context feature, more robust and informative context information will be aggregated to the object pixels, which then assists the classification of confusing objects. Objects with fast motion gains more detection precision improvement from the temporal context information encoding than objects with slow motion, which also confirms the point, because frames with fast-motion objects are more likely to have motion blur.

Table 2 Ablation study on the spatial context and temporal context in the proposed spatial-temporal Context Information Encoding (stCIE) module

# method	SELSA (baseline)	SELSA + sCIE	SELSA + stCIE
mAP(%)	82.7	83.2	83.7
mAP(%) slow	88.9	89.4	89.7
mAP(%) medium	81.2	81.8	82.5
mAP(%) fast	65.4	66.6	67.9

'sCIE' means encoding context information only along the spatial dimension

Table 3 Quantitative comparison between different importance weight types

# importance weight type 	
mAP(%)
	Fixed
	84.7
		Learnable
		85.2

mAP(%) slow	90.5	90.8
mAP(%) medium	83.5	84.2
mAP(%) fast	69.4	70.5

'Fixed' means that we use the same value for all the importance weights, i.e., importance weights are 1/9 for the case of 9 patches in each proposal, and 'learnable' means that the importance weights of the target proposal patches are learnt from the patch feature with our proposed SPFA

4.7 Evaluation of SPFA

As we discussed in the methodology section, an alternative but straightforward way to obtain the patch weights is simply using equal importance weights for all the patches of the target proposal. We treat this simple equal importance weight way to aggregate patch feature as a baseline, and compare it with the learnable patch weights in the proposed SPFA. The comparison results are summarized in Table 3, which shows that the detection accuracy decreases when we replace the learnable patch importance weights with fixed ones (i.e., the importance weights of all proposal patches are equal).

We further study the proposed SPFA module by visualizing some examples in Fig. 9. Note that we use 18 support frames for inference, and here 3 support frames are randomly selected out of the 18 frames for visualization. In each row of this figure, the first column shows the target proposal and the other columns are the selected support proposals to provide feature aggregation to the target proposal. The yellow scores in the dashed green boxes under the target proposals are the corresponding normalized importance weights W_i of the patches cropped by the green boxes, and the green scores in the dashed green boxes under the support proposals are the normalized similarity scores S_m^n between the target proposal and the support proposals based on the target proposal patch cropped by the green boxes. From these examples we can see that when an object is partially occluded, the importance weights of the occluded patches are much smaller than that of the non-occluded patches when performing feature aggregation with the proposed Structure-based Proposal Feature Aggregation (SPFA) module, which demonstrates that the proposed SPFA works as we expect.

Combining the comparison results in Table 3 with the visual examples in Fig. 9, we can conclude that the proposed SPFA can automatically learn the corresponding importance weight for each target proposal patch, with which the heavily occluded patches can be underrated or even ignored when searching for support proposal features.

4.8 Position of Context Information Encoding

An alternative way to encode context information into object feature is directly using the non-local network (Wang et al. 2018b) on the $conv4$ feature map in the pixel level. There are two ablation factors, RPN with or without context encoding, and aggregation position ($conv4$ or $conv5$ feature). Based on these two ablation factors, we design the corresponding experiments to evaluate the effectiveness of the proposed stCIE.

For the first experiment, we perform the context information encoding on $conv4$ feature map. The context information encoded $conv4$ feature map is input to a RPN to generate ROIs, and the backbone network to further obtain the $conv5$ feature. The $conv5$ feature is then used to extract the object proposal feature by a RoI pooling operation. We denote this experiment as 'Non-local ($conv4$ and RPN)'. In the second experiment, the original $conv4$ is used to generate the ROIs with the RPN. We still perform the context information encoding on $conv4$ feature map, but the context information encoded $conv4$ feature map is only input to the backbone network to obtain the $conv5$ feature, which is then used to extract the object proposal feature by a RoI pooling operation. We denote this experiment as 'Non-local ($conv4$)'. In the last experiment, we use the proposed stCIE to perform context information encoding, and we denote this experiment as 'stCIE'. Note that the structure-based feature aggregation (SPFA) is always included in each experiment. To get rid of the influence of randomness, each experiment has been run for 3 times to get the mean and standard deviation of the detection accuracy. The results are presented in Table 4.

From Table 4 it can be concluded that compared to encode context information on $conv4$ feature with a non-local network ('non-local ($conv4$+RPN) + SPFA' and 'non-local ($conv4$) + SPFA' in Table 4), mining context information

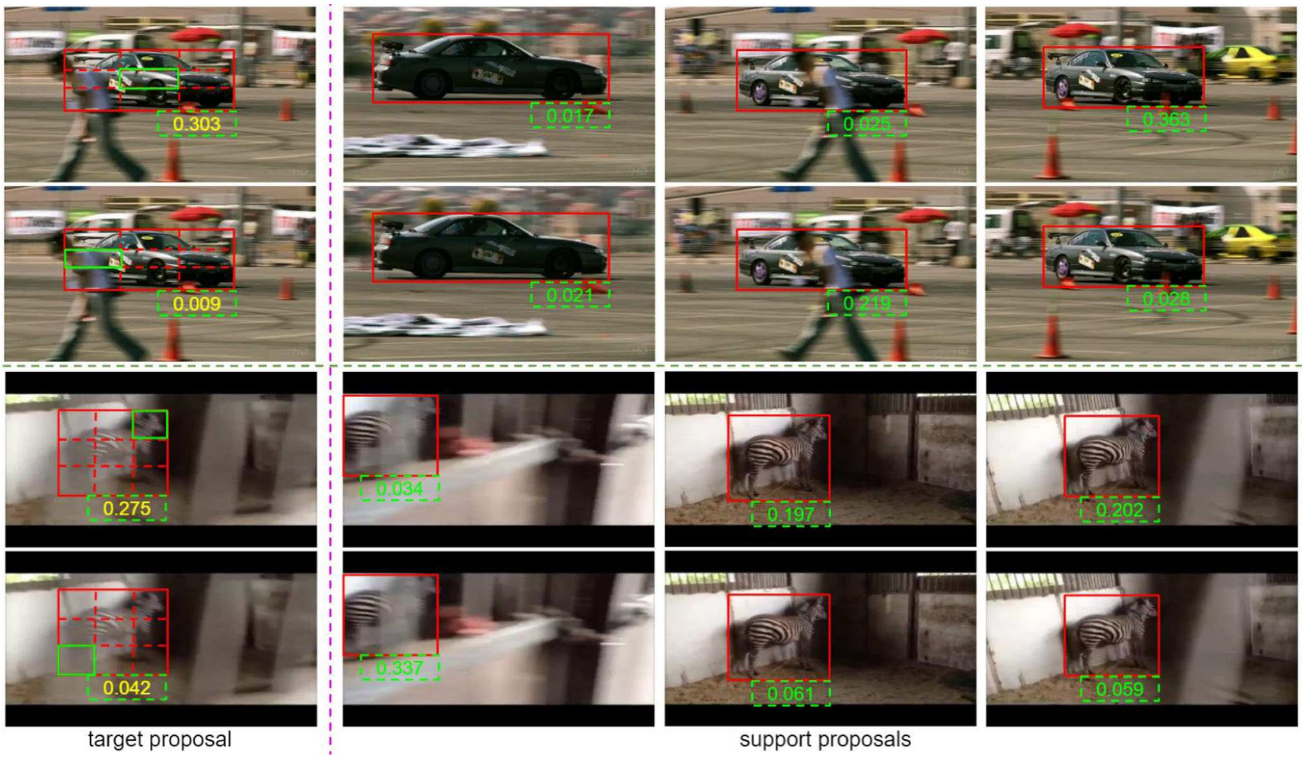


Fig. 9 Visualization of the calculated patch importance weights and the similarity scores between the target proposal and the support proposals. The yellow scores in the dashed green boxes under the target proposals are the corresponding normalized importance weights W_i of the patches cropped by the green boxes, and the green scores in the dashed green boxes under the support proposals are the normalized similarity scores S_m^n between the target proposal and each support proposal based on the target proposal patch cropped by the green boxes

on $conv5$ feature with the proposed stCIE module ('stCIE + SPFA' in Table 4) improves the detection accuracy. Besides, the standard variation of detection accuracy of mining context information on $conv5$ feature map with the proposed stCIE is much smaller than that of the non-local on $conv4$, which means that the stCIE based context information mining on $conv5$ feature map generates more stable detections. The possible reason is that the context information encoded $conv4$ feature, will further go through some convolutional layers in the backbone to generate the $conv5$ feature, in which the object feature will inevitably include some background information. Thus, when extracting proposal feature with the $conv5$ feature map, the extracted proposal feature may contain some background information, which will affect the final detection.

Further, when performing context information encoding on the $conv4$ feature and generating the ROIs with this context information encoded $conv4$ feature ('non-local ($conv4$+RPN) + SPFA' in Table 4), the detection accuracy is worse than that of performing context information encoding on the $conv4$ feature but generating the ROIs with the original $conv4$ feature ('non-local ($conv4$) + SPFA' in Table 4). The reason is that when performing a pixel level feature aggregation on the $conv4$ feature maps with a non-local network, some object information will also be encoded into the background pixel features. When a RPN is adopted on the feature aggregated $conv4$ maps to generate proposals, some background proposals might be labeled as object proposals by the RPN, which will then pollute the target proposal features during feature aggregation.

We then visualize some detection results in Fig. 10. The first two columns are some detection results of performing the context information encoding on the $conv4$ feature map and using the encoded $conv4$ feature map to generate proposals (i.e., $conv4$+RPN) + SPFA' in Table 4), and the last two columns are the corresponding detection results of performing the context information encoding with the proposed stCIE (i.e., 'stCIE + SPFA' in Table 4). From the examples in the first two columns we can see that some background proposals which have similar appearance features with object proposals are more likely to be classified as the corresponding objects. This is because when we perform the context information encoding on the $conv4$ feature map and use the encoded $conv4$ feature map to generate proposals, the background proposals will be encoded with some object information if these background proposals have similar appearance feature

Table 4 Ablation studies on the position of perform context information encoding

Method	Metric	Experiment 1	Experiment 2	Experiment 3	Mean ± Std.
Non-local (*conv*4+RPN) + SPFA	mAP(%) overall	84.77	84.45	84.60	84.61 ± 0.160
	mAP(%) slow	90.44	90.28	90.33	90.35 ± 0.082
	mAP(%) medium	83.62	83.15	83.36	83.38 ± 0.235
	mAP(%) fast	69.88	68.97	69.29	69.38 ± 0.462
Non-local (*conv*4) + SPFA	mAP(%) overall	84.89	84.95	85.01	84.95 ± 0.060
	mAP(%) slow	90.52	90.57	90.61	90.57 ± 0.045
	mAP(%) medium	83.80	83.88	83.96	83.88 ± 0.080
	mAP(%) fast	70.00	70.10	70.19	70.10 ± 0.095
Non-local (*conv*5) (i.e., stCIE) + SPFA	mAP(%) overall	85.18	85.21	85.10	85.16 ± 0.057
	mAP(%) slow	90.73	90.75	90.67	90.72 ± 0.042
	mAP(%) medium	84.18	84.23	84.07	84.16 ± 0.082
	mAP(%) fast	70.46	70.55	70.30	70.44 ± 0.127

‘std.’ denotes the standard deviation

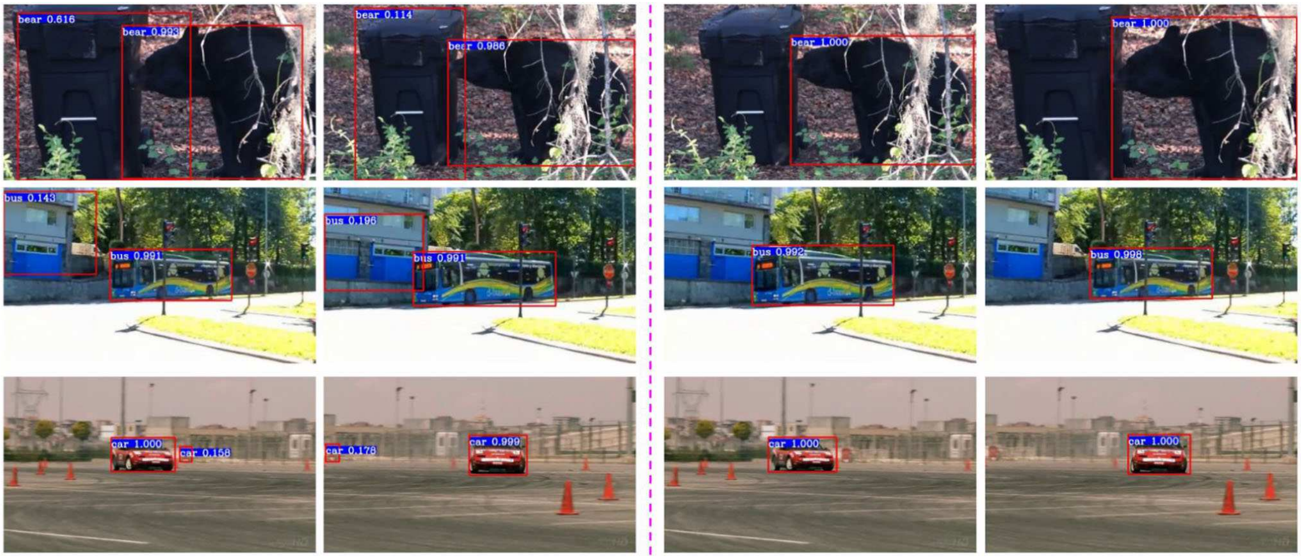


Fig. 10 Visualization of some detection results. First two columns: detection results of performing the context information encoding on the *conv*4 feature map and using the encoded *conv*4 feature map to generate proposals (i.e., *conv*4+RPN) + SPFA’ in Table 4). Last two columns: the corresponding detection results of performing the context information encoding with the proposed stCIE (i.e., ‘stCIE + SPFA’ in Table 4

with the object, and as a result, the RPN might mistakenly regard these background proposals as object proposals. However, in the last two columns, these background proposals can be classified as background correctly because in our proposed stCIE module, we perform the context information encoding on the *conv*5 feature map, and when the RPN classifies the proposals, the object information will not be encoded into the background proposals, and thus it is easier for RPN to distinguish the background proposals. This comparison verifies our statement that with the modifications in our proposed stCIE, the background pixel features in *conv*4 will not be contaminated by the object information, and thus it is easier for RPN to filter out the background proposals from the object proposals.

4.9 Analysis on Number of Support Frames

Intuitively, sampling more support frames will yield better detection results during inference, because with more support frames sampled, more temporal information can be extracted from the support frames and then leveraged to aggregate features for target proposals. Unfortunately, more support frames means more computation cost (GPU memory, inference time). Accordingly, establishing a suitable support

Table 5 Influence of support frame number F on detection precision

# frames	0	2	6	10	14	18
mAP(%)	74.4	83.1	84.3	84.7	85.0	85.2
mAP(%) slow	82.4	89.3	90.2	90.6	90.7	90.8
mAP(%) medium	71.6	81.5	82.9	83.6	84.1	84.2
mAP(%) fast	52.4	66.6	69.4	70.1	70.3	70.5

frame number is very important. Considering the promising performance of the random sampling strategy (Wu et al. 2019), we adopt it for our support frame sampling. Table 5 summarizes the influence of the support frame number on detection performance. When the number of support frames is 0, i.e., no support frame is sampled, the detection result is very bad (only 74.4%mAP), because no feature aggregation is performed to deal with the challenges such as occlusion, motion blur, and rare pose in target frames without sampling any support frame. When two support frames are sampled in the experiment for feature aggregation, we get a much better detection performance (+8.7%mAP), which demonstrates the effectiveness of the feature aggregation operation. With the number of support frames increasing, the detection performance improves consistently. This is because with more support frames sampled, more temporal information such as object appearance, pose, shape, etc. can be mined by the feature aggregation module to enhance the target proposal features. Moreover, more support frames will provide more informative context, which will then be encoded into the object proposal features and benefit the final detection. One interesting thing is that objects with faster motion gain more improvement than objects with slower motion when using more support frames for feature aggregation. The reason is that fast-moving objects usually have much more shape and pose variations, which cause the pose misalignment. Also, fast-moving objects are more likely to be occluded, or with motion blur. Sampling more support frames can provide various supportive information for the target proposal objects with deteriorated appearance. Unfortunately, the improvement is saturated when support frames are up to a certain number. This is reasonable, because when the number of support frames is large enough, adding more support frames only can provide very limited extra supportive information. Thus, to balance the detection accuracy and computation cost, we set the number of support frames as 18 for our experiments.

4.10 Analysis on number of patches

Detection models do benefit from the parallel multi-head attention module (Hu et al. 2018), which runs through an attention mechanism in each head in parallel, and the independent attention outputs are then concatenated and linearly transformed into the expected dimension. Intuitively, multiple attention heads are expected to capture different relations between the input features. We also adopt the multiple head strategy in our proposed structure-based proposal feature aggregation (SPFA) module. However, the biggest difference between the SPFA and the traditional multi-head attention module is that the input to each head of the traditional multi-head attention module is the same, and each head is expected to learn different relations between the input, while the inputs to each head of our SPFA are different, i.e., each head in our SPFA takes as input a certain patch of the target proposal and all the patches of the support proposals. Different heads in the SPFA exploit the relation between the target proposal and support proposals based on different object parts, and finally the individual relation outputs are concatenated together. By doing this, the SPFA measures the relation between different proposals by exploiting the object structure information.

The number of segmented proposal patches N is another important hyper-parameter in our experiment settings, and can have significant influence on our final detection precision. In this subsection, we conduct some experiments to analyze the influence of the patch number.

In the experiments, the proposal patches are divided in equal stride and with equal size. The proposal features generated with RoIAlign pooling are with the size of $K \times K \times C$ ($K = 8$ in our experiments) where C is the feature channel dimension. We keep the proposal size invariant in all the experiments. When dividing each proposal into 2×2 (i.e., 4 patches) non-overlapping patches, each patch is with the size of $4 \times 4 \times C$. When dividing each proposal into 3×3 (i.e., 9 patches) non-overlapping patches, we first increase the proposal size to $9 \times 9 \times C$ by replicatively padding the proposal feature by 1 on the top and on the left, and then equally divide the proposal into patches. In this case, each patch is with the spatial size of $3 \times 3 \times C$. When dividing each proposal into 4×4 (i.e., 16 patches) non-overlapping patches, each patch is with the size of $2 \times 2 \times C$.

Table 6 presents the influence of the patch number N on detection precision. $N = 1$ means that the object proposals (both the target proposal and the support proposals) are not divided into patches, instead, the proposals themselves are used to calculate the similarities between proposals and perform feature aggregation without computing the patch importance weights [i.e., proposal feature aggregation is performed in the way as in SELSA (Wu et al. 2019)]. When the patch number is too small (e.g., $N = 4$), the detection precision is not good enough, because the structure information can not be fully captured. When the patch number is too big (e.g., $N = 16$), the detection precision also drops, since small patches (patches with small spatial size) capture limited object feature, and the object structure information can not be mined well either. Besides, this also shows that more heads in the SPFA do not always give us higher accuracy.

Table 7 Ablation studies on attention modules

# module	Multi-head attention (8 heads)	Multi-head attention (16 heads)	SPFA
mAP(%)	83.9	83.9	85.2
mAP(%) slow	89.8	89.8	90.8
mAP(%) medium	82.7	82.8	84.2
mAP(%) fast	68.2	68.4	70.5

'Multi-head attention' denotes the traditional multi-head attention module Hu et al. (2018)

Table 6 Influence of patch number N on detection precision

# patches	1	4	9	16
mAP(%)	83.7	85.0	85.2	84.8
mAP(%) slow	89.7	90.6	90.8	90.5
mAP(%) medium	82.5	84.0	84.2	83.9
mAP(%) fast	67.9	70.4	70.5	70.1

To further verify that the detection model benefits from the cross-patch feature compensation design instead of the multi-head model capacity, we design another experiment, in which we replace the proposed SPFA with a traditional multi-head attention module (Hu et al. 2018). The multi-head attention module does not divide the proposal into patches, instead, it takes as input the extract target and support proposal features for each head to exploit the relation between the target and support proposals, just as many state of the art (Deng et al. 2019a; Wu et al. 2019; Chen et al. 2020) do for aggregating target proposal feature. We use 8 (8 is used here instead of 9 because the input feature channel is usually an exponent of 2 and is divisible by 8) and 16 heads in the multi-head attention module, and the results are summarized in Table 7. When adopting the traditional multi-head attention module (Hu et al. 2018) for proposal feature aggregation, the performance is worse than that of using SPFA, even when more heads (16 heads) are used in the traditional multi-head attention module. This demonstrates that the proposed SPFA module benefits from the cross-patch feature compensation design instead of the multi-head model capacity.

4.11 Comparison with State of the Art

In this subsection, we compare the proposed framework with 13 baseline methods to show its effectiveness. The compared methods are:

- D & T (Feichtenhofer et al. 2017) is a ConvNet architecture for simultaneous detection and tracking. It uses correlation features that represent object co-occurrences across time to aid the ConvNet during tracking, and link the frame level detections based on the tracklets.
- FGFA (Zhu et al. 2017a) adopts optical flow to guide the feature aggregation. It improves the per-frame features by aggregation of nearby features along the motion paths to improve the video object detection.
- MANet (Wang et al. 2018a) jointly calibrates the features of objects on both pixel-level and instance-level. The pixel-level calibration targets to modeling detailed motion, while the instance-level calibration aims to capture more global motion cues in order to be robust to occlusion.
- ST-Lattice (Chen et al. 2018a) performs expensive detection sparsely and propagates the results across both scales and time with substantially cheaper networks, by exploiting the strong correlations among them.
- STSN (Bertasius et al. 2018) uses deformable convolutions across space and time to leverage temporal information for object detection in video. It learns to spatially sample useful feature points from nearby video frames.
- STMN (Xiao and Jae Lee 2018) adopts recurrent computation unit to model long-term temporal appearance and motion dynamics. It also proposes a MatchTrans module to align the spatial-temporal memory from frame to frame to tackle object motion in videos.
- PSLA (Guo et al. 2019) establishes the spatial correspondence between features across frames in a local region with progressively sparser stride and uses the correspondence to propagate features. It can be used to replace the expensive optical flow models.
- LWDN (Jiang et al. 2019) designs the locally-weighted deformable neighbors to latently aligh the high-level features between keyframes and keyframes or non-keyframes without utilizing time-consuming optical flow extraction.
- LRTRN (Shvets et al. 2019) leverages the temporal relation module by operating on object proposals to learn the similarities between proposals from different frames, and selects proposals from past and/or future to support current proposals.
- RDN (Deng et al. 2019a) uses object guided hard attention to improve storage-efficiency and propagate temporal information through external memory to address long-term dependency in video object detection.
- SELSA (Wu et al. 2019) aggregates semantic features across frames on the proposal level in the full-sequence level using relation networks.

Table 8 Comparison with state of the art on ImageNet VID validation set

Method	Backbone	RT (FPS)	mAP (%)
D & T (Feichtenhofer et al. 2017)	ResNet-101	7.8	75.8
D & T+TR (Feichtenhofer et al. 2017)	ResNet-101	–	79.8
D & T+TR (Feichtenhofer et al. 2017)	ResNeXt-101	–	81.6
FGFA (Zhu et al. 2017a)	ResNet-101	1.4	76.3
FGFA+Seq-NMS (Zhu et al. 2017a)	ResNet-101	–	78.4
MANet (Wang et al. 2018a)	ResNet-101	5.0	78.1
MANet+Seq-NMS (Wang et al. 2018a)	ResNet-101	–	80.3
ST-Lattice+TR (Chen et al. 2018a)	ResNet-101	20.0	79.6
STSN+Seq-NMS (Bertasius et al. 2018)	ResNet-101	–	80.4
STMN+Seq-NMS (Xiao and Jae Lee 2018)	ResNet-101	1.2	80.5
PSLA (Guo et al. 2019)	ResNet-101	–	80.0
PSLA+Seq-NMS (Guo et al. 2019)	ResNet-101	–	81.4
LWDN (Jiang et al. 2019)	ResNet-101	20.0	76.3
LRTRN (Shvets et al. 2019)	ResNet-101	10.0	81.0
RDN (Deng et al. 2019a)	ResNet-101	10.6 (V)	81.8
RDN (Deng et al. 2019a)	ResNeXt-101	–	83.2
RDN+BLR (Deng et al. 2019a)	ResNet-101	–	83.8
SELSA (Wu et al. 2019)	ResNet-101	1.2	82.7
SELSA (Wu et al. 2019)	ResNeXt-101	–	84.3
CenterNet (Zhou et al. 2019)	ResNet-101	47.0	73.6
CenterNet+Seq-NMS (Zhou et al. 2019)	ResNet-101	43.0	75.9
MEGA (Chen et al. 2020)	ResNet-101	4.2	82.9
MEGA (Chen et al. 2020)	ResNeXt-101	–	84.1
MEGA+BLR (Chen et al. 2020)	ResNet-101	–	84.5
CenterNetHP (Xu et al. 2020)	ResNet-101	37.0	76.7
CenterNetHP+Seq-NMS (Xu et al. 2020)	ResNet-101	34.0	78.4
KCF-RL(CenterNet-LSTM) (Yao et al. 2020)	ResNet-101	95.2 (C)	53.3
KCF-RL(YOLOV3-LSTM) (Yao et al. 2020)	ResNet-101	71.5 (C)	59.9
LSTS (Jiang et al. 2020)	ResNet-101	23.0	77.2
LSTS (Jiang et al. 2020)	ResNet-101+DCN	21.2	80.1
LSTS+Seq-NMS (Jiang et al. 2020)	ResNet-101+DCN	4.6	82.1
HVR (Han et al. 2020b)	ResNet-101	–	83.2
HVR+Seq-NMS (Han et al. 2020b)	ResNet-101	–	83.8
HVR+Seq-NMS (Han et al. 2020b)	ResNeXt-101	–	85.5
CSMN (Ours)	ResNet-101	1.1	85.2
CSMN (Ours)	ResNeXt-101	1.0	86.2
CSMN w/o data augmentation (Ours)	ResNet-101	1.1	83.1
CSMN w/o data augmentation (Ours)	ResNeXt-101	1.0	84.3

‘RT’ repsents ‘Runtime’ ‘X+Y’ means post-processing strategy Y is employed on method X. ‘A-B’ means technique A is combined with technique B. ‘TR’ is tubelet rescoring, ‘BLR’ means box linking with relations. V means that the speed is tested on TITAN V GPU, and C means that the speed is tested on Intel Xeon E5 CPU

- CenterNet (Zhou et al. 2019) is a center point based approach, which uses keypoint estimation to estimate a heatmap to identify the locations of the object center points, and regresses to the bounding box sizes.
- MEGA (Chen et al. 2020) takes full consideration of both global and local information for proposal level feature aggregation using relation networks, and designs a long range memory to get access to more contents.
- CenterNetHP (Xu et al. 2020) is a video object detector based on CenterNet. It propagates the previous reliable long-term detection in the form of heatmap to boost results of upcoming image with a heatmap propagation.

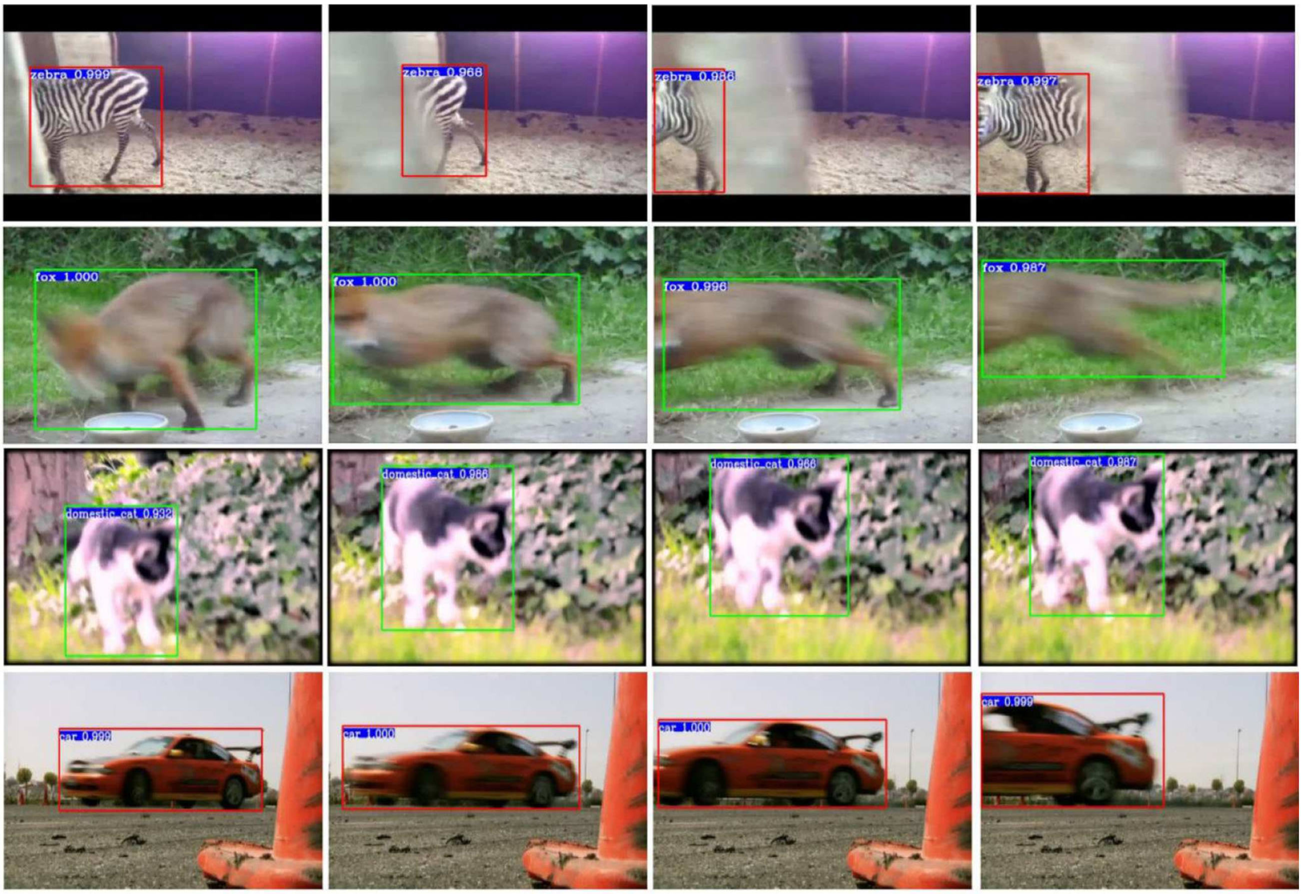


Fig. 11 Qualitative results on some challenging cases of the VID dataset

- KCP-RL (Yao et al. 2020) adopt object tracker for temporal propagation, and using reinforcement learning for adaptive key-frame scheduling.
- LSTS (Jiang et al. 2020) learns the semantic-level correspondences among adjacent frame features using learnable spatial-temporal sampling. Temporal relations are enhanced by sparsely recursive feature updating and dense feature aggregation.
- HVR (Han et al. 2020b) tries to learn effective object representations via modeling relations of hard proposals among different videos, based on a multi-level triplet selection scheme.

The comparison results are summarized in Table 8. For fair comparison, we first compare our model with state of the art by adopting the same backbone network (ResNet-101) in all of the models. From the results we can see that our model beats D & T (Feichtenhofer et al. 2017) (75.8% mAP), a single-frame object detection method without feature aggregation, by a large margin (+9.4%). When compared with some optical flow based feature aggregation methods, our model is significantly better than FGFA (Zhu et al. 2017a) (76.3% mAP) and MANet (Wang et al. 2018a) (78.1% mAP), and the detection improvements are +8.9% mAP and +7.1% mAP, respectively. Besides, our method also shows its superior over some relation based feature aggregation methods (e.g., LRTRN (Shvets et al. 2019) (81.0% mAP), RDN (Deng et al. 2019a) (81.8% mAP), SELSA (Wu et al. 2019) (82.7% mAP), MEGA (Chen et al. 2020) (82.9% mAP), LSTS (Jiang et al. 2020) (82.1% mAP), HVR (Han et al. 2020b) (83.2% mAP)), which are considered as the most advanced VOD algorithms, on the detection precision. Like many previous state of the art that could gain further improvement by adopting a more powerful backbone network (ResNeXt-101), it could also benefit our model, and with no doubt, our model still performs the best.

To get rid of the affect of data augmentation during training and make fair comparison with the state of the art, we also train our CSMN model without data augmentation, and the detection results are summarized in the last two rows of Table 8. Training without data augmentation degrades the detection accuracy by $\sim 2\%$ mAP, while the performance is still competitive to the state of the art such as MEGA Chen et al. (2020) and HVR Han et al. (2020b).

Our proposed CSMN is built on SELSA Wu et al. (2019), and it improves the detection accuracy of SELSA by more than 2% mAP. This demonstrates the effectiveness of the proposed CSMN.

Post-processing benefits most of these methods more or less on detection precision (mAP). For example, the D & T algorithm achieves a +4.0% mAP improvement with the tubelet rescoring post-processing. FGFA, MANet, and PLSA get an improve of +1 ∼ 2% mAP by using the Seq-NMS post-processing strategy. In Deng et al. (2019a), they also design a novel post-processing technique called Box Linking with Relations (BLR) for their proposed RDN algorithm, which gives better refined detection performance (83.8% mAP). Nonetheless, our proposed framework still achieves the best performance.

For runtime, our methods achieves 1.1 FPS, comparable to the relation networks based methods, SELSA (Wu et al. 2019) and MEGA (Chen et al. 2020), with ∼ +2 mAP improvement. It is worth noting that some state of the art (Zhou et al. (2019); Yao et al. (2020); Xu et al. (2020); Jiang et al. (2019, 2020), etc.) aim at accelerating the video object detection, while our proposed CSMN targets on improving detection accuracy. Thus, the detection algorithms proposed in these three references achieve high detection speed, but the detection accuracy of these algorithms are much worse compared with the proposed CSMN.

Figure 11 presents the qualitative evaluation of the proposed method on some challenging VOD cases (e.g., part occlusion, motion blur, out-of-focus camera), which demonstrates that the proposed framework can handle these challenges well.

5 Conclusion

In this work, we propose a context and structure mining network for video object detection, which includes a spatial-temporal context information encoding module to encode the spatial-temporal context information in video frames into object features, and a structure-based proposal feature aggregation module to better aggregate target proposal features with temporal information in support frames. By encoding the spatial-temporal context information, more accurate object classification is achieved. Moreover, the object structure information enables us to find the most informative and supportive features to aggregate target proposal features even when some VOD challenges such as occlusion, pose misalignment, etc. exits on video objects. Experiments show the effectiveness of the the proposed framework, which achieves state-of-the-art video object detection performance.

References

Bertasius, G., Torresani, L., & Shi, J. (2018). Object detection in video with spatiotemporal sampling networks. In: *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 331–346).

Chen, K., Wang, J., Yang, S., Zhang, X., Xiong, Y., Change Loy, C., & Lin, D. (2018a). Optimizing video object detection via a scale-time lattice. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7814–7823).

Chen, Y., Cao, Y., Hu, H., & Wang, L. (2020). Memory enhanced global-local aggregation for video object detection. In: *CVPR*.

Chen, Z., Huang, S., & Tao, D. (2018b). Context refinement for object detection. In: *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 71–86).

Dai, J., Li, Y., He, K., & Sun, J. (2016). R-FCN: Object detection via region-based fully convolutional networks. In: *Advances in neural information processing systems* (pp. 379–387).

Deng, H., Hua, Y., Song, T., Zhang, Z., Xue, Z., Ma, R., Robertson, N., & Guan, H. (2019a). Object guided external memory network for video object detection. In: *Proceedings of the IEEE international conference on computer vision* (pp. 6678–6687).

Deng, J., Pan, Y., Yao, T., Zhou, W., Li, H., & Mei, T. (2019b). Relation distillation networks for video object detection. In: *Proceedings of the IEEE International Conference on Computer Vision* (pp. 7023–7032).

Dosovitskiy, A., Fischer, P., Ilg, E., Hausser, P., Hazirbas, C., Golkov, V., Van Der Smagt, P., Cremers, D., Brox, T . (2015). Flownet: Learning optical flow with convolutional networks. In: *Proceedings of the IEEE international conference on computer vision* (pp. 2758–2766).

Feichtenhofer, C., Pinz, A., & Zisserman, A. (2017). Detect to track and track to detect. In: *Proceedings of the IEEE international conference on computer vision* (pp. 3038–3046).

Fu, C. Y., Liu, W., Ranga, A., Tyagi, A., & Berg, A. C. (2017). DSSD: Deconvolutional single shot detector. arXiv preprint arXiv:1701.06659

Gao, Z., Wang, L., & Zhou, L. (2018). A probabilistic approach to cross-region matching-based image retrieval. *IEEE Transactions on Image Processing*, *28*(3), 1191–1204.

Girshick, R. (2015). Fast r-cnn. In: *Proceedings of the IEEE international conference on computer vision* (pp. 1440–1448).

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 580–587).

Guo, C., Fan, B., Gu, J., Zhang, Q., Xiang, S., Prinet, V., & Pan, C. (2019). Progressive sparse local attention for video object detection. In: *Proceedings of the IEEE international conference on computer vision*.

Han, L,, Wang, P., Yin, Z., Wang, F., & Li, H. (2020a). Exploiting better feature aggregation for video object detection. In: *ACM MM*.

Han, M., Wang, Y., Chang, X., & Qiao, Y. (2020b). Mining inter-video proposal relations for video object detection. In: *European conference on computer vision* (pp. 431–446). Springer.

Han, W., Khorrami, P., Paine, T. L., Ramachandran, P., Babaeizadeh, M., Shi, H., Li, J., Yan, S., & Huang, T. S. (2016). Seq-NMS for video object detection. arXiv preprint arXiv:1602.08465

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).

He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask R-CNN. In: *Proceedings of the IEEE international conference on computer vision* (pp. 2961–2969).

Howard, AG., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861

Hu, H., Gu, J., Zhang, Z., Dai, J., & Wei, Y. (2018). Relation networks for object detection. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3588–3597).

Jiang, Z., Gao, P., Guo, C., Zhang, Q., Xiang, S., & Pan, C. (2019). Video object detection with locally-weighted deformable neighbors. *Proceedings of the AAAI Conference on Artificial Intelligence*, *33*, 8529–8536.

Jiang, Z., Liu, Y., Yang, C., Liu, J., Gao, P., Zhang, Q., Xiang, S., & Pan, C. (2020). Learning where to focus for efficient video object detection. In: *European conference on computer vision* (pp. 18–34). Springer.

Kang, K., Ouyang, W., Li, H., & Wang, X. (2016). Object detection from video tubelets with convolutional neural networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 817–825).

Kang, K., Li, H., Yan, J., Zeng, X., Yang, B., Xiao, T., et al. (2017). T-CNN: Tubelets with convolutional neural networks for object detection from videos. *IEEE Transactions on Circuits and Systems for Video Technology*, *28*(10), 2896–2907.

Kantorov, V., Oquab, M., Cho, M., & Laptev, I. (2016). Contextlocnet: Context-aware deep network models for weakly supervised localization. In: *European Conference on Computer Vision* (pp 350–365). Springer.

Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017a). Feature pyramid networks for object detection. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2117–2125).

Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017b). Focal loss for dense object detection. In: *Proceedings of the IEEE international conference on computer vision* (pp. 2980–2988).

Liu, M., & Zhu, M. (2018). Mobile video object detection with temporally-aware feature maps. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5686–5695).

Liu, M., Zhu, M., White, M., Li, Y., & Kalenichenko, D. (2019). Looking fast and slow: Memory-guided mobile video object detection. arXiv preprint arXiv:1903.10172

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, AC. (2016). SSD: Single shot multibox detector. In: *European conference on computer vision* (pp. 21–37). Springer.

Redmon, J., Farhadi, A .(2017) . Yolo9000: better, faster, stronger. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7263–7271).

Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779–788).

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. In: *Advances in neural information processing systems*, vol 28 (pp. 91–99).

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., et al. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, *115*(3), 211–252.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4510–4520).

Sharif Razavian A, Sullivan J, Maki A, Carlsson S (2015) A baseline for visual instance retrieval with deep convolutional networks. In: *International conference on learning representations*, 7–9 May 2015. San Diego. ICLR: CA.

Shvets, M., Liu, W., & Berg, A. C. (2019). Leveraging long-range temporal relationships between proposals for video object detection. In: *Proceedings of the IEEE international conference on computer vision* (pp. 9756–9764).

Tian, Z., Shen, C., Chen, H., & He, T. (2019). FCOS: Fully convolutional one-stage object detection. In: *ICCV*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In: *Advances in neural information processing systems* pp. 5998–6008.

Wang, S., Zhou, Y., Yan, J., & Deng, Z. (2018a). Fully motion-aware network for video object detection. In: *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 542–557).

Wang, X., Girshick, R., Gupta, A., & He, K. (2018b). Non-local neural networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7794–7803).

Wu, H., Chen, Y., Wang, N., Zhang, Z. (2019). Sequence level semantics aggregation for video object detection. In: *Proceedings of the IEEE international conference on computer vision* (pp. 9217–9225).

Xiao, F., & Jae Lee, Y. (2018). Video object detection with an aligned spatial-temporal memory. In: *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 485–501).

Xu, Z., Hrustic, E., & Vivet, D. (2020). Centernet heatmap propagation for real-time video object detection. In: *European conference on computer vision* (pp. 220–234).

Yao, CH., Fang, C., Shen, X., Wan, Y., & Yang, MH. (2020). Video object detection via object-level temporal aggregation. In: *European conference on computer vision* (pp. 160–177).

Zhou, X., Wang, D., & Krähenbühl, P. (2019). Objects as points. arXiv preprint arXiv:1904.07850

Zhu, X., Wang, Y., Dai, J., Yuan, L., & Wei, Y. (2017a). Flow-guided feature aggregation for video object detection. In: *Proceedings of the IEEE international conference on computer vision* (pp. 408–417).

Zhu, X., Xiong, Y., Dai, J., Yuan, L., & Wei, Y. (2017b). Deep feature flow for video recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Zhu, X., Dai, J., Yuan, L., & Wei, Y. (2018). Towards high performance video object detection. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7210–7218).

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.