

Sequence Level Semantics Aggregation for Video Object Detection

Haiping Wu¹ Yuntao Chen^{3,4} Naiyan Wang² Zhaoxiang Zhang^{3,4,5}

¹ McGill University ² TuSimple

³ University of Chinese Academy of Sciences

⁴ Center for Research on Intelligent Perception and Computing, CASIA

⁵Center for Excellence in Brain Science and Intelligence Technology, CAS

haiping.wu2@mail.mcgill.ca chenyuntao2016@ia.ac.cn winsty@gmail.com zhaoxiang.zhang@ia.ac.cn

Abstract

Video objection detection (VID) has been a rising research direction in recent years. A central issue of VID is the appearance degradation of video frames caused by fast motion. This problem is essentially ill-posed for a single frame. Therefore, aggregating useful features from other frames becomes a natural choice. Existing methods heavily rely on optical flow or recurrent neural networks for feature aggregation. However, these methods emphasize more on the temporal nearby frames. In this work, we argue that aggregating features in the whole sequence level will lead to more discriminative and robust features for video object detection. To achieve this goal, we devise a novel Sequence Level Semantics Aggregation (SELSA) module. We further demonstrate that the proposed method has a close relationship with the classical spectral clustering methods, thus providing a novel view to understand the VID problem. Lastly, we test our proposed method on the large-scale ImageNet VID dataset and EPIC KITCHENS dataset and archive new state-of-the-art results compared with previous works. Moreover, to achieve such superior performance, we do not need other complicated post-processing methods such as Seq-NMS or Tubelet rescoring as in previous works, which keeps our pipeline simple and clean.

1. Introduction

Recent years have witnessed fast progress in object detection using deep convolutional networks. Renewed detection paradigms [8, 25, 11], strong backbone [12, 34] and large scale datasets [18, 16] jointly push forward the limit of object detection.

Video Object Detection (VID) has now emerged as a new challenge beyond object detection in still images. Thanks to

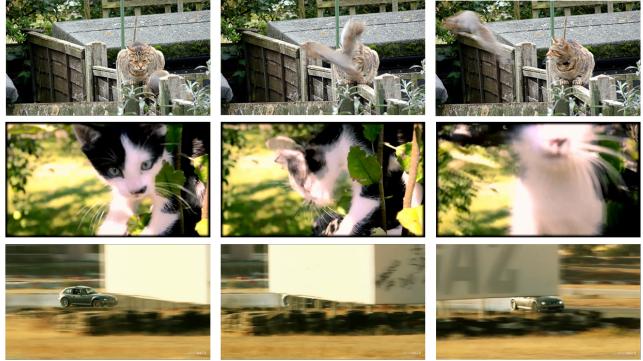


Figure 1. Challenges in video object detection. Motion blur, camera defocus and pose variation.

the fast progress in still image object detection, detectors' performance on slow-moving objects in video object detection has somewhat saturated [36]. The main challenge now lies in the scenario where objects or cameras are under fast motion.

Fast motion brings up appearance degradation unseen in still image setting like motion blur, camera defocus and large pose variation as shown in Figure 1. Still image detectors often fail in these cases. On the other hand, a video provides far richer visual information of a single object than a still image. Typically, each object of interest in the video sequence consists of hundreds of shots across different frames, containing rich visual information. When an object's appearance deteriorates drastically in a frame, it is natural to incorporate information from the video (e.g. nearby frames) to mitigate this degradation. The second and third columns in Figure 1 show the various difficult sequences in VID. Though in these hard cases, there are still some frames more salient than the others. A good video object detector should be able to identify the salient views

to refine its beliefs on those of degraded views if they are (semantically) similar, either to support its beliefs or deny them. Note that useful information is not necessarily from temporal nearby frames, any objects share high similarity with the object of interest in any frames (even within the same frame) could contribute.

Post-processing methods try to incorporate video-level information by designing sophisticated rule set for linking bounding boxes generated by still image detectors. These two-stage methods are not jointly optimized and may lead to sub-optimal results. Instead, end-to-end feature aggregation utilizes motion information estimated from optical flow [37] or instance tracking [29] for object feature calibration. Feature calibration methods heavily rely on accurate motion estimation, but this requirement is somewhat contradictory. In the circumstance of motion blur or fast motion, it is just the hardest case for optical flow estimation. Usually, the results of optical flow are also unsatisfactory in such cases, which makes it unusable and unhelpful for VID task.

To lift this limitation in a principled way, we need to take a deeper look at the video itself. Existing works generally take video as sequential frames, and thus mainly utilize the temporal information to enhance the performance of a detector. For example, Flow Guided Feature Aggregation (FGFA) [36] uses at most 21 frames during training and testing, which is less than 5% of average video length. Instead of taking a consecutive viewpoint, we propose to treat video as a bag of unordered frames and try to learn an invariant representation of each class on the whole-video level. This reinterprets video object detection from a sequential detection task to a multi-shot detection task.

In the multi-shot view, a video is consisted of clusters of objects, with each cluster containing hundreds even thousands of shots. The appearance degradation of an object is the manifestation of large intra-class feature variance. Thus reducing the feature variance lies in the core of addressing appearance changes. As mentioned before, temporal feature aggregation is a well-established way for feature variance reduction. However, it fails to utilize the rich information beyond a fixed time window.

We take a further step by clustering and enhancing features in the entire sequence level. In this work, we present Sequence Level Semantics Aggregation (SELSA) method. We introduce a SELSA module which is inspired by spectral clustering. ROI features are extracted from frames sampled in the whole video, and then go through our clustering module and transformation module. The enhanced features are handed to the detection head to get final detection results. Our method is throughoutly tested on the large scale ImageNet VID dataset. We have designed thorough ablation experiments to demonstrate the effectiveness of proposed methods. We achieve 82.7 mAP with Faster-

RCNN detector and ResNet-101 backbone, 84.3 mAP with ResNeXt-101 backbone, improving the state-of-the-art results by a large margin. Additional experiments on EPIC KITCHENS [4] dataset show that our method generalize to complex scenes.

In summary, our contributions are three folds:

1. We first treat video detection as a sequence level multi-shot detection problem and then introduce a global clustering viewpoint of VID task for the first time.
2. To incorporate such view into current deep object detection pipeline, we introduce an end-to-end Sequence Level Semantics Aggregation (SELSA) module to fully utilize video information.
3. We test our proposed method on the large scale ImageNet VID and EPIC KITCHEN dataset and demonstrate significant improvement over state-of-the-art methods.

2. Related Work

In this section, we briefly review several works that closely related to our method.

2.1. Object Detection in Still Images

Thanks to the success of deep neural networks, state-of-the-art detection systems [25, 3] are based on deep convolution neural networks (CNNs). The typical two-stage detector R-CNN [9] first extracts regional features from backbone networks based on deep CNNs, and then classifies and refines the corresponding bounding boxes. Fast R-CNN [8] proposed ROI Pooling operation to speed up the regional feature extraction process. Traditionally, region proposals are generated through selective search [28]. The Regional Proposal Network (RPN) was proposed in Faster R-CNN [25] to generate region proposals through deep CNNs, using backbone networks shared with Fast R-CNN. R-FCN [3] introduced position-sensitive ROI Pooling operation, improving the detection efficiency by sharing the computation of regional features.

On the other hand, one-stage object detector directly predicts the bounding box of interest based on the extracted feature map from CNN. Without the extra stage, one-stage detector is usually faster than the two-stage counterpart. Representative works include YOLO [22] and its variants [23, 24], SSD [19] and its variants [7, 17]. Nevertheless, one-stage detector can hardly extend to more complicated tasks such as key point detection and instance segmentation. Similarly in our work, it can hardly be extended to extract proposal-level object semantic features. Thus we choose Faster R-CNN as our basic still image detector.

Recently, high-level relations among objects in object detection has been studied in [13, 30]. These works model

the appearance and geometry relations among object proposals within a single image. This enables joint reasoning of objects and improves the accuracy and it could also be used as a duplicate removal step instead of NMS since the geometry is embedded. Similarly, our work also captures relations among objects. However, we especially capture the relation measured by semantic similarity (objects of the same class across the video) instead of high-level interaction between objects (e.g person v.s glove in [13]). We use these similarities to guide our feature aggregation and alleviate problems introduced by videos (fast motion).

2.2. Object Detection in Videos

For object detection in videos, the main challenge lies in how to utilize the rich information of videos (e.g. temporal continuity) to improve the accuracy as well as the speed upon still image detectors.

Several previous works devised various post-processing techniques applied to the results of still image detectors by leveraging temporal information: Kang *et al.* [15, 14] proposed to suppress false positive detections via multi-context suppression (MCS) and propagate predicted bounding boxes across frames using the motion calculated by optical flow. Then a temporal convolution neural network is trained to rescore the tubelets generated using visual tracking. Feichtenhofer *et al.* [6] performed single-frame object detection and object movements regression across frames (tracking) in a multi-task fashion. Then it links the detections across frames to object tubelets using the predicted movements, and re-weights detection scores in tubelets. Han *et al.* [10] proposed Seq-NMS to form high score linkages using bounding box IoU across frames and then rescore the boxes associated with each linkage to the average or maximum scores of the linkage. These methods perform box-level post-processing upon still image detections, which could be sub-optimal since they are not optimized jointly. In contrast, our method manages to leverage video-level information at proposal-level by end-to-end optimization without post-processing steps.

Another line of work [14] focuses on utilizing optical flow to extract motion information to facilitate object detection. However, such pre-computed optical flow is neither efficient nor task related. Deep Feature Flow (DFF) [37] is the first work that adopts in-network fine-tuned optical flow computation. It utilizes the optical flow generated by FlowNet [5] to propagate and aligns the features of selected keyframes to nearby non-keyframes, thus reducing redundant calculation and speeding up the system. FGFA [36] is built on DFF [37]. However, its objective is to improve the accuracy by aligning and aggregating features from keyframes using optical flow. Based on DFF and FGFA, MANet [29] adds an instance-level feature calibration and aggregation module besides the pixel-level one in FGFA,

and then it combines these two levels through a motion pattern reasoning module. Furthermore, [35] and [1] design more advanced feature propagation and keyframe selection mechanisms to improve the accuracy as well as the speed.

Using optical flow to calibrate features across frames could be error-prone since object location, appearance and pose could change dramatically, where optical flow estimation becomes unreliable. Unlike these methods, our method does not intend to align features across frames by temporal information. We aggregate features on the proposal level, which makes our method more robust and superior.

Tripathi *et al.* [27] trained a recurrent neural network to refine its initial detection results. Lu *et al.* [20] used association LSTM to address the object association between consecutive frames. STMN [33] used a Spatial-Temporal Memory module as the recurrent operation to pass the information through a video. Unlike [33], our method does not need to pass information using memory modules in temporal order. We form clusters and aggregate features in a multi-shot view to capture the rich information of videos instead. Also, our clustering and feature aggregation are performed on instance-level features, where redundant pixel-level calculation is unnecessary. Moreover, it focuses more on subjects of interest.

3. Method

In this section, we first describe the motivation of our SEquence Level Semantics Aggregation (SELSA) method in Sec. 3.1. We then elaborate the details of our SELSA module in Sec. 3.2. We further interpret our method from the clustering view in Sec. 3.3. Finally, we discuss the relation between our method and existing works in Sec. 3.4.

3.1. Motivation

Feature aggregation is an efficient way to mitigate appearance degradation in video detection. The vital part of this method is to choose proper features in the video for aggregation. Previous methods [29, 36] generally aggregate features from a temporally local window. But appearance deterioration could also span a wide time window, thus makes temporally local methods less effective. Moreover, in nearby frames, the appearance of objects may be highly redundant, consequently weaken the advantage of feature aggregation. To address this problem, we propose to aggregate feature within the semantic neighborhood, which is not susceptible to the appearance degradation lasting in time.

3.2. SEQUENCE LEVEL SEMANTICS AGGREGATION

The best way for feature aggregation is to aggregate within the ground truth tracklet. But the golden association for proposals in different frames is not available during test phase. Inspired by the ReID-based association which is popular in multi-object tracking system [32], we propose to

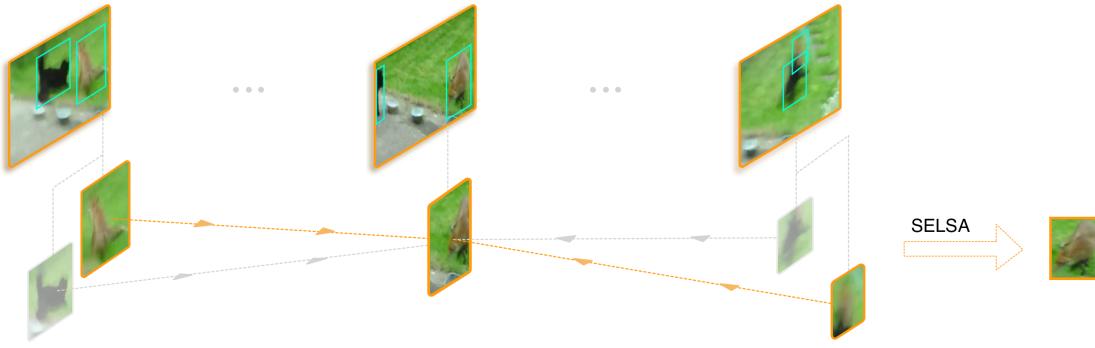


Figure 2. The overall architecture of the proposed model. We first extract proposals in different frames from the video, then the semantic similarities of proposals are computed across frames. At last, we aggregate the features from other proposals based on these similarities to obtain a more discriminative and robust features for object detection.

link the proposals cross space-time with their semantic similarities. This semantic feature based association approach is well known for its robustness to appearance change.

Semantic Guidance Suppose for each frame f , let $\mathbf{X}^f = \{\mathbf{x}_1^f, \mathbf{x}_2^f, \dots\}$ be the proposals generated by the RPN network from Faster-RCNN. For a specific pair of proposals $(\mathbf{x}_i^k, \mathbf{x}_j^l)$, we measure the pairwise semantic similarity between them with the generalized cosine similarity:

$$w_{ij}^{kl} = \phi(\mathbf{x}_i^k)^T \psi(\mathbf{x}_j^l), \quad (1)$$

where $\phi(\cdot)$ and $\psi(\cdot)$ represent some general transformation functions. Higher similarity indicates a higher chance of proposals being the same category.

Feature Aggregation After defining the similarity between proposals, the semantic similarity now serves as a guide for the reference proposal to aggregate features from other proposals. By aggregating across multiple proposals, the new proposal feature contains much richer information and should be robust against appearance variation like pose change, fast camera motion, motion blur, and object deformation. Moreover, since the similarities are built on the proposal level, they are more robust compared with the optical flow which is computed on each position in feature maps.

In order to preserve the magnitude of the features after aggregation, we normalize the similarities with softmax function across all proposals. Formally, suppose that we are aggregating from randomly picked F frames in the video with N proposals produced in each frame, the aggregated feature for reference proposal is defined as:

$$\bar{\mathbf{x}}_i^k = \sum_{j \in \Omega} \sum_{l=1}^N w_{ij}^{kl} \mathbf{x}_i^k, \quad (2)$$

where Ω is the set of frame indexes randomly selected for the aggregation. The whole SELSA module is differentiable and can be optimized end-to-end with standard SGD. After the aggregation, the enhanced proposal features are further fed into the detection header network for classification and bounding box regression as in the original Faster-RCNN. An intuitive illustration of our proposed method is depicted in Figure 2.

3.3. A Spectral Clustering Viewpoint

Besides the simple and intuitive formulation of our method, we further reveal its deep connection with the classical spectral clustering algorithm.

With proposals \mathbf{X} as nodes and similarity \mathbf{W} as edges, we can define a semantic similarity graph $G = (\mathbf{X}, \mathbf{W})$ on proposals. From a probabilistic viewpoint, the random walk on graph G is controlled by the stochastic matrix \mathbf{T} which is obtained by normalizing each row in \mathbf{W} to sum 1. \mathbf{T}_{ij} describes the transition probability from proposal i to proposal j during a random walk. Proposals belong to the same class should form a subgraph $\mathbf{A} \subset \mathbf{X}$. For feature aggregation, we are especially interested in minimizing the risk of incorrectly aggregating the features of a proposal which does not belong to the reference class. This risk can be measured by the transition probability $P_{\bar{\mathbf{A}}\mathbf{A}}$ from the subgraph $\bar{\mathbf{A}} = \mathbf{X} - \mathbf{A}$ to the subgraph \mathbf{A} .

The transition probability between subgraphs is formally defined as,

$$P_{\bar{\mathbf{A}}\mathbf{A}} = \frac{\sum_{i \in \bar{\mathbf{A}}, j \in \mathbf{A}} \pi_i \mathbf{T}_{ij}}{\sum_{i \in \bar{\mathbf{A}}} \pi_i}, \quad (3)$$

where $\pi_i = \sum_k \mathbf{W}_{jk} / \sum_{j,k} \mathbf{W}_{jk}$ denotes the stationary distribution of the graph. π_i represents the connection strength between a proposal and the rest proposals in a graph.

As proved in [21], the transition probability is equivalent

to the normalized minimum cut,

$$\text{NCut}(\mathbf{A}, \bar{\mathbf{A}}) = P_{\mathbf{A}\bar{\mathbf{A}}} + P_{\bar{\mathbf{A}}\mathbf{A}}. \quad (4)$$

From the traditional spectral clustering view, the stochastic matrix \mathbf{T} is fixed, and the transition probability is minimized by finding the optimal partition $\mathbf{A}, \bar{\mathbf{A}}$. However, from the supervised deep learning view, the stochastic matrix \mathbf{T} derived from proposal features is the variable to optimize, and the optimal partition $\mathbf{A}, \bar{\mathbf{A}}$ is given. The optimization of \mathbf{T} is further propagated to the proposal features and backbone network for discriminative feature learning. Furthermore, [21] gives the desired form of \mathbf{T} , a block-wise diagonal matrix w.r.t $\mathbf{A}, \bar{\mathbf{A}}$, which is exactly the desired guide for proposal feature aggregation.

3.4. Connection to Graph Convolution Network

Recently, Wang *et al.* [31] have applied GCN for video classification task. They built a space-time graph with a similar affinity measurement to us. In their work, they took the edges of a graph as a general relation in space-time and mainly focus on modeling the high order interaction of objects in a video. However, in our work, we design the SELSA module to refine the features of a reference proposal by the relationship between them, which leads to a different motivation and optimization objective.

4. Experiments on ImageNet VID

In this section, we first present the dataset and evaluation metric used in Sec. 4.1, then followed by the implementation details of our method in Sec. 4.2. We next justify the design choice of our SELSA module in Sec. 4.3 by ablation studies. We also investigate the effects of existing post-processing techniques on our method. Finally, we compare our method with other state-of-the-art methods.

4.1. Dataset and Evaluation Setup

We evaluate our proposed methods on ImageNet VID dataset [26]. We follow the protocol widely used in [36, 37] and report the mAP and motion-specific mAP on the validation set for evaluation and comparison.

4.2. Implementation Details

Feature Network. We use ResNet-101 [12] as the backbone feature network for ablation studies. ResNext-101 (32 x 4d) [34] is also utilized for the final results. Following the practice in [36], the stride of *conv5* block is changed from 32 to 16 using dilated convolutions.

Detection Network. RPN is applied on the feature maps of *conv4* stage. Totally 9 anchors are used, corresponding to 3 scales and 3 aspect ratios. Then Fast R-CNN is applied on the feature maps of *conv5* stage following [36]. We apply

Method	(a)	(b)	(c)
Semantics Aggregation		✓	✓
Multi-frame			✓
mAP (%)	73.62	75.26 _{↑1.64}	80.25 _{↑6.63}
mAP (%) (slow)	82.12	83.59 _{↑1.47}	86.91 _{↑4.79}
mAP (%) (medium)	70.96	72.88 _{↑1.92}	78.94 _{↑7.98}
mAP (%) (fast)	51.53	51.43 _{↓0.10}	61.38 _{↑9.85}

Table 1. Detection results on ImageNet VID validation. For multi-frame methods, 21 frames are used when testing. No post-processing techniques are used. The relative gains (drops) compared to the baseline are shown in the subscript.

two fully connected (FC) layers upon the RoI pooled features followed by classification and bounding box regression.

SELSA Module. We insert two SELSA modules into our network. Each one is inserted after one fully-connected layer in Faster R-CNN (FC → SELSA → FC → SELSA). The general transformation functions in Eq. 1 are realized by one fully-connected layer.

Training and Testing Details. The backbone networks are initialized by weights obtained from models pre-trained on ImageNet classification task [26]. The models are trained on a mixture of ImageNet VID and DET datasets. A total of 220k iterations of SGD training is performed on 4 GPUs. The initial learning rate is 2.5×10^{-4} and is divided by 10 at the 110k and the 165k iterations. For training, one training frame is sampled along with two random frames from the same video (identical frames if are sampled from DET dataset). For inference, K frames from the same video are sampled along with the inference frame. In both training and inference, the images are resized to a shorter side of 600 pixels.

4.3. Ablation study

In this subsection, we study the impact of each design choice and parameter settings.

Effectiveness of SELSA. Table 1 compares our proposed methods with the single-frame baseline.

Column (a) shows the results of our single-frame baseline. It uses ResNet-101 as the backbone and achieves a reasonable mAP of 73.62 as in [36].

Column (b) performs semantics aggregation (SA) within a single frame, a degenerated variant of SELSA. More specifically, only proposals obtained from the same frame

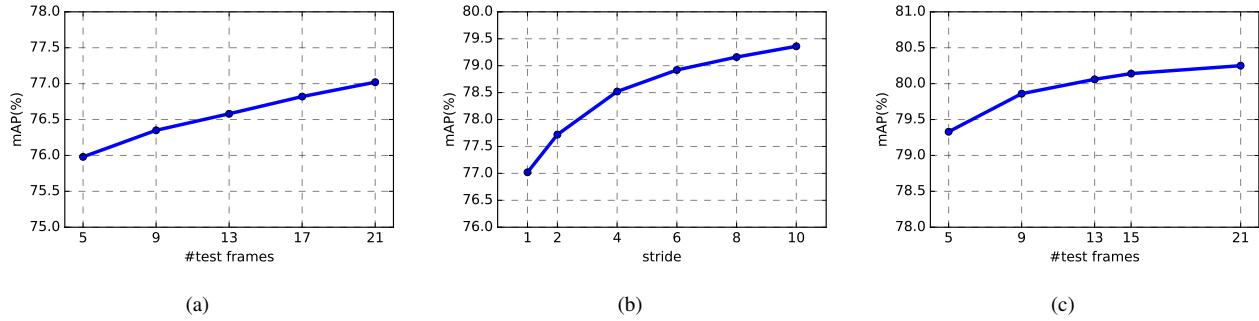


Figure 3. Ablation analyses of different test settings. (a) The effect of different number of frames on sequential test performance. (b) The effect of different sequence sampling time stride on sequential test performance. (c) The effect of different number of frames on shuffled test performance.

are considered as possible semantic neighbors for aggregation. This leads to an mAP of 75.26, a gain of 1.64 mAP compared to the baseline. When multiple objects with the same semantics or multiple proposals corresponding to the same object appear in the same frame, the semantic aggregated proposal features are hence enhanced with contextual information like in [13, 2], thus leading to the performance improvement. Note that for objects under fast motion, the mAP (fast) receives no improvement over baseline. This indicates that appearance degradation induced by fast motion could not be remedied by the contextual or object interaction information.

Column (c) is the proposed SELSA method. It utilizes the SELSA module to enhance proposal features by sampling semantic neighbors from the whole video sequence. It gives an mAP of 80.25, a large 6.63 mAP improvement compared to the baseline method. Note that it enhances the motion-specific performance in fast motion to 61.38 mAP, which is a huge improvement of 9.95 mAP compared with the baseline. Compared with column (b) and (c), it is easy to see that our method directly harvests high-quality features from aggregating sequence level features other than high order interaction information on the graph, as previously stated in 3.4.

Sampling Strategies for Feature Aggregation Frame sampling strategy matters for video detection. As previous works [33, 36] pointed out, using more frames in testing usually yields better results. Besides, [33] samples frames with a stride in testing to improve the performance. Specifically, by using a sampling stride of S , one frame in every S frames is used for testing instead of consecutive frames.

We examine the influence of the number of frames used and sampling strides when testing our method. First, we use no stride and vary the number of frames used. As seen in Figure 3(a), with more frames used for testing, the performance increases consistently. For example, using 21 frames instead of 5 contributes 1.04 mAP improvement. Then we

fix the number of frames to 21 for aggregation and examine the impact of sampling stride. Figure 3(b) shows the performance variation when using different sample strides. Increasing the sampling stride from 1 to 10 further improves the performance from 77.02 to 79.36 mAP (a gain of 2.34 mAP). Notice that the sampling stride demonstrates a larger influence on the performance than the number of testing frames in general, which coincides with our assumption that our sequence level method could benefit more from sample diversity. Other feature aggregation methods which use optical flow or RNN cannot benefit from the larger stride since it violates the temporal continuity assumption of these methods.

Semantics Aggregation in Sequence Level. As discussed earlier, good features for aggregation in VID should be more diverse in terms of appearance and poses. This observation motivates the use of semantic neighbors instead of temporal neighbors. Thus, taking a step further, we sample semantic neighbors uniformly from the whole video sequence regardless of the temporal orders (shuffled test setting). This is feasible since our method does not rely on any temporal information (e.g. optical flow), and also no feature alignment operation across frames is performed. Our method is exempt from possible inaccurate predictions of temporal information (e.g. optical flow estimation [36], bounding box shifting prediction [6]) and feature alignment process [37, 29], which is important when the motion is large. In fact, performance drops have been shown in optical flow based method [29] as the number of frames increase when exceeding a certain threshold (12 frames in [29]). Our method, on the contrary, shows its power of performing feature aggregation in the whole video sequence level in Figure 3(c). As we have seen, using only 5 frames in shuffled test already achieves the same level of performance as 21 frames in strided testing. And using 21 frames along with shuffled testing gives an mAP of 80.25. This introduces an improvement of 0.89 mAP against to the strong result of

79.36 mAP where a sampling stride of 10 and in total 21 frames are used. This gain comes from sampling more diverse features in semantic neighbors rather than temporal neighbors, which further shows the effectiveness of SELSA for capturing whole video sequence level information for feature aggregation. This is the default test setting in the following experiments.

Data augmentation. Existing VID datasets usually suffer from lacking of semantics diversity. Frames in a video are highly similar to each other and thus lead to potential overfitting. Thus we adopt data augmentation to alleviate this problem. Photometric distortion, random expand and random crop as in [19] are used besides the original random flipping operation. This gives us an improvement of 2.44 mAP, leading to 82.69 mAP when using ResNet-101 as backbone.

4.4. Video-level post-processing techniques.

One advantage of our method is that it does not rely on post-processing methods (e.g Seq-NMS) to incorporate whole-video level information. Nearly all the state-of-the-art video detection systems [36, 29, 6, 1, 33] adopted post-processing methods which give high gains in performance. To illustrate that our method has already captured the whole sequence level information, we further apply Seq-NMS upon our method. Table 2 shows the results of how Seq-NMS affects our methods when using different backbone networks. As easily seen, adding Seq-NMS only has a minor impact on the results. In particular, adding Seq-NMS to ResNet-101/ResNext-101 backbone network yields 0.21/0.57 mAP drop.

Referring to Table 3, post processing methods have introduced large performance improvement upon existing state-of-the-art methods: 2.1 mAP for FGFA [36] and 2.2 mAP for MANet [29] with Seq-NMS and 4 mAP for D (& T loss) [6] with tubelet rescore. In contrast, almost no gain from Seq-NMS upon our method with ResNet-101 as backbone network shows that our method has already largely captured the whole-video level information through our SELSA module without any post-processing techniques. Moreover, different from post-processing methods like Seq-NMS which involves two separate stages, our method could be trained end-to-end with sequence level information. As the backbone feature network becomes stronger, our method could even better utilize such sequence level information, thus shows a better result than that with Seq-NMS, in which the separate post-processing steps might lead to sub-optimal results.

4.5. Comparison with the state-of-the-art methods.

Table 3 summarizes the performance of our methods and other state-of-the-art methods on the ImageNet VID valida-

Backbone	ResNet-101	ResNext-101
Seq-NMS	✓	✓
mAP (%)	82.69	82.48 _{±0.21}

Table 2. Results of our method on ImageNet VID validation w/o Seq-NMS with different backbone feature networks. The relative gains (drops) compared with the method without Seq-NMS are shown in the subscript.

Methods	Backbone	mAP (%)
FGFA [36]	ResNet-101	76.3
D (& T loss) [6]		75.8
MANet [29]		78.1
Ours		80.25
FGFA* [36]	ResNet-101	78.4
MANet* [29]		80.3
ST-Lattice* [1]		79.6
D&T* [6]		79.8
STMN*+ [33]		80.5
Ours*		80.54
Ours [△]		82.69
D&T* [6]	ResNext-101	81.6
D&T* [6]	Inception-v4	82.1
Ours	ResNext-101	83.11
Ours [△]	ResNext-101	84.30

Table 3. Performance comparison with state-of-the-art systems on ImageNet VID validation set. * indicates use of video-level post-processing methods (e.g Seq-NMS, Tube Rescoring). + indicates use of model emsembling. [△] indicates use of data augmentation.

tion set. Our method achieves the best performance among various testing settings.

With no video-level post-processing techniques, compared with FGFA [36] (76.3 mAP) and MANet [29] (78.1 mAP) which are both built on flow-based feature aggregation, our method is remarkably better (80.25 mAP), outperforming these two methods by 3.95 and 2.15 mAP, respectively. It also outperforms D (& T loss) [6] by a large margin of 4.45 mAP.

The middle part of Table 3 shows the comparison with methods that utilize sequence level post-processing techniques. FGFA*, MANet* and STMN*+ [33] use Seq-NMS, while D&T* [6], ST-Lattice* [1] utilize Tubelet rescoring. Our method, by using Seq-NMS as the post-processing method, achieves 80.54 mAP, which is slightly better than the previous state-of-the-art method STMN*+.

Furthermore, by replacing the backbone feature network from ResNet-101 to ResNext-101, our method achieves performance of 83.11 mAP **without** any post-processing techniques (e.g Seq-NMS), which surpasses the D&T with ResNext-101 backbone and tubelet rescoreing by a large

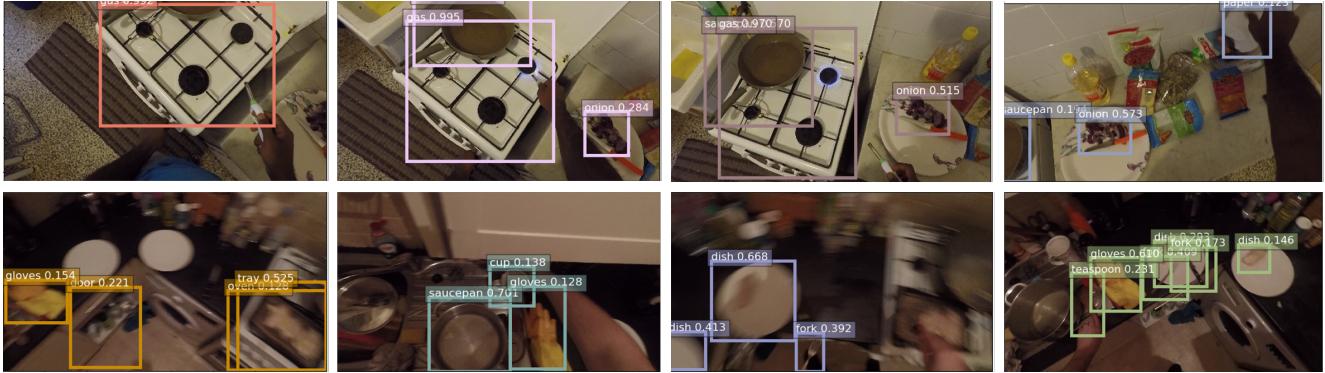


Figure 4. Visual results of our method on EPIC KITCHENS.

margin (1.15 mAP). Our method benefits from the stronger representation power introduced by better backbone networks. When equipped with training data augmentation, our methods show a significant gain of 2.44/1.19 mAP for ResNet-101/ResNext-101. This indicates SELSA can benefit from the diversity of proposal features during aggregation. These results reveal the potential of our proposed method.

5. Additional Experiments on Epic Kitchen

ImageNet VID dataset falls short of the lack of diversity. Here we evaluate SELSA on the EPIC KITCHENS dataset [4], in which frames usually contain multiple objects from different classes, which is far more complex and challenging.

5.1. Dataset and Evaluation Setup

EPIC KITCHENS [4] is a large scale egocentric dataset, capturing daily activities happened in the kitchens. The video object detection tasks consists of 32 different kitchens with 454,255 object bounding boxes spanning 290 classes. 272 video sequences captured in 28 kitchens are used for training. 106 sequences collected in the same 28 kitchens (S1) and 54 sequences collected in other 4 unseen kitchens (S2) are used for evaluation. Videos are annotated in 1s interval.

5.2. Implementation Details

Mostly, we adopt the same network setting as on ImageNet VID dataset. We adopt no data augmentation except random horizontal flip. A total of 600k iterations of SGD training is performed on 4 GPUs. The initial learning rate is 2.5×10^{-4} and is divided by 10 at the 300k iterations. For both training and inference, we sample frames within a ± 10 s window for the SELSA module.

	S1		
Methods	mAP@.05	mAP@.5	mAP@.75
EPIC [4]	45.99	34.18	8.49
Faster R-CNN	53.12	36.57	9.97
Ours	54.67	37.97	9.81
	S2		
Methods	mAP@.05	mAP@.5	mAP@.75
EPIC [4]	44.95	32.01	7.87
Faster R-CNN	48.91	31.86	7.36
Ours	50.25	34.80	8.10

Table 4. Performance comparison on EPIC KITCHENS test set. S1 and S2 indicate Seen and Unseen splits.

5.3. Results and Analysis

Here we present some preliminary results on the EPIC KITCHENS dataset. As shown in Table 4, SELSA improves over Faster R-CNN baseline by 1.4/2.94 mAP for Seen/Unseen splits. Although the training scheme and the hyper parameter selection are far from optimal, our method still achieves promising results. This shows that SELSA is applicable to more complex video detection tasks. Figure 4 shows some results of our method.

6. Conclusion

In this work, we have proposed a novel view of VID problem by taking the whole sequence level feature aggregation. Instead of using methods such as optical flow or RNN, we propose a simple yet effective SELSA module for aggregating semantic features across frames. Since the aggregation is conducted on the proposal level rather than feature map or even pixel level, our method is more robust to motion blur and large pose variation. Furthermore, we have derived the connection between our method and the classical spectral clustering method, providing a novel clustering view of our method. Extensive ablation analyses

demonstrate the effectiveness of the proposed SELSA module. When compared with previous methods, our method achieves superior performance without sophisticated post-processing methods.

References

- [1] K. Chen, J. Wang, S. Yang, X. Zhang, Y. Xiong, C. C. Loy, and D. Lin. Optimizing video object detection via a scale-time lattice. In *CVPR*, 2018. [3](#), [7](#)
- [2] Z. Chen, S. Huang, and D. Tao. Context refinement for object detection. In *ECCV*, 2018. [6](#)
- [3] J. Dai, Y. Li, K. He, and J. Sun. R-FCN: Object detection via region-based fully convolutional networks. In *NIPS*, 2016. [2](#)
- [4] D. Damen, H. Doughty, G. Maria Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, et al. Scaling egocentric vision: The epic-kitchens dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 720–736, 2018. [2](#), [8](#)
- [5] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox. FlowNet: Learning optical flow with convolutional networks. In *CVPR*, 2015. [3](#)
- [6] C. Feichtenhofer, A. Pinz, and A. Zisserman. Detect to track and track to detect. In *ICCV*, 2017. [3](#), [6](#), [7](#)
- [7] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg. DSSD: Deconvolutional single shot detector. *arXiv:1701.06659*, 2017. [2](#)
- [8] R. Girshick. Fast R-CNN. In *CVPR*, 2015. [1](#), [2](#)
- [9] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. [2](#)
- [10] W. Han, P. Khorrami, T. L. Paine, P. Ramachandran, M. Babaieizadeh, H. Shi, J. Li, S. Yan, and T. S. Huang. Seq-NMS for video object detection. *arXiv:1602.08465*, 2016. [3](#)
- [11] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN. In *ICCV*, 2017. [1](#)
- [12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. [1](#), [5](#)
- [13] H. Hu, J. Gu, Z. Zhang, J. Dai, and Y. Wei. Relation networks for object detection. In *CVPR*, 2018. [2](#), [3](#), [6](#)
- [14] K. Kang, H. Li, J. Yan, X. Zeng, B. Yang, T. Xiao, C. Zhang, Z. Wang, R. Wang, X. Wang, et al. T-CNN: Tubelets with convolutional neural networks for object detection from videos. *TCSVT*, 2017. [3](#)
- [15] K. Kang, W. Ouyang, H. Li, and X. Wang. Object detection from video tubelets with convolutional neural networks. In *CVPR*, 2016. [3](#)
- [16] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, T. Duerig, and V. Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *arXiv:1811.00982*, 2018. [1](#)
- [17] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. *TPAMI*, 2018. [2](#)
- [18] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014. [1](#)
- [19] 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. [2](#), [7](#)
- [20] Y. Lu, C. Lu, and C.-K. Tang. Online video object detection using association LSTM. In *ICCV*, 2017. [3](#)
- [21] M. Meila and J. Shi. A random walks view of spectral segmentation. In *AISTATS*, 2001. [4](#), [5](#)
- [22] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, 2016. [2](#)
- [23] J. Redmon and A. Farhadi. YOLO9000: better, faster, stronger. *ICCV*, 2017. [2](#)
- [24] J. Redmon and A. Farhadi. YOLOv3: An incremental improvement. *arXiv:1804.02767*, 2018. [2](#)
- [25] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. [1](#), [2](#)
- [26] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *IJCV*, 2015. [5](#)
- [27] S. Tripathi, Z. C. Lipton, S. Belongie, and T. Nguyen. Context matters: Refining object detection in video with recurrent neural networks. *arXiv:1607.04648*, 2016. [3](#)
- [28] J. R. Uijlings, K. E. Van De Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. *International journal of computer vision*, 104(2):154–171, 2013. [2](#)
- [29] S. Wang, Y. Zhou, J. Yan, and Z. Deng. Fully motion-aware network for video object detection. In *ECCV*, 2018. [2](#), [3](#), [6](#), [7](#)
- [30] X. Wang, R. Girshick, A. Gupta, and K. He. Non-local neural networks. In *CVPR*, 2018. [2](#)
- [31] X. Wang and A. Gupta. Videos as space-time region graphs. *ECCV*, 2018. [5](#)
- [32] N. Wojke, A. Bewley, and D. Paulus. Simple online and realtime tracking with a deep association metric. In *ICIP*, 2017. [3](#)
- [33] F. Xiao and Y. J. Lee. Video object detection with an aligned spatial-temporal memory. In *ECCV*, 2018. [3](#), [6](#), [7](#)
- [34] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017. [1](#), [5](#)
- [35] X. Zhu, J. Dai, L. Yuan, and Y. Wei. Towards high performance video object detection. In *CVPR*, 2018. [3](#)
- [36] X. Zhu, Y. Wang, J. Dai, L. Yuan, and W. Yichen. Flow-guided feature aggregation for video object detection. In *ICCV*, 2017. [1](#), [2](#), [3](#), [5](#), [6](#), [7](#)
- [37] X. Zhu, Y. Xiong, J. Dai, L. Yuan, and Y. Wei. Deep feature flow for video recognition. In *CVPR*, 2017. [2](#), [3](#), [5](#), [6](#)