

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

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

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

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 features from other frames becomes a natural choice. Existing methods rely heavily on optical flow or recurrent neural networks for feature aggregation. However, these methods emphasize more on the temporally nearby frames. In this work, we argue that aggregating features in the full-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 the close relationship between the proposed method and the classic spectral clustering method, providing a novel view for understanding the VID problem. We test the proposed method on the ImageNet VID and the EPIC KITCHENS dataset and achieve new state-of-the-art results. Our method does not need complicated post-processing methods such as Seq-NMS or Tubelet rescoring, which keeps the 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 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

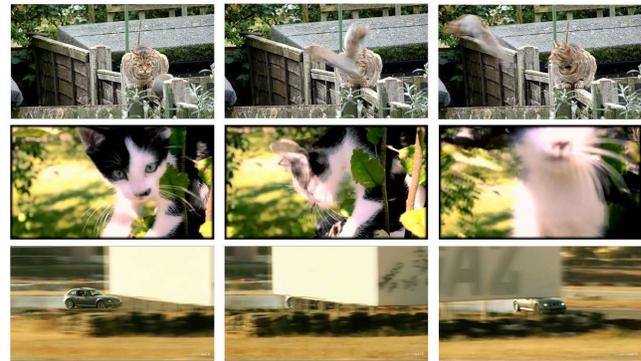


Figure 1. Challenges in video object detection. Motion blur, camera defocus and pose variation.

motion.

Fast motion brings up image degradation unseen in the 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 than a still image. When the appearance of an object deteriorates in a frame, it is natural to include information from the video (e.g. nearby frames) to mitigate this degradation. The second and third columns in Figure 1 show 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 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, which is somewhat contradictory. In the circumstance of fast motion, the appearance of objects degrades drastically. Thus the results of optical flow are usually unsatisfactory in such cases, which makes it less helpful 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 full sequence level. *This reinterprets video object detection from a sequential detection task to a multi-shot detection task.*

In the multi-shot view, a video consists 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 SELSA module which is inspired by spectral clustering. Features of Region of Interests (ROI) are extracted from frames sampled from 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 thoroughly tested on the large scale ImageNet VID and EPIC KITCHEN datasets. We also design ablation experiments to demonstrate the effectiveness of proposed methods. We achieve 82.7 mAP with Faster-RCNN detector and ResNet-101 backbone and 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 more 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 a simple but effective 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 datasets and demonstrate significant improvement over previous methods.

2. Related Work

In this section, we briefly review several works that are 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 RoIPooling 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 RoIPooling 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. It could also be used as a duplicate removal step instead of NMS since the geometry relations 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 align 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 effective way to mitigate the appearance degradation in video detection. The vital part of this method is to choose proper features for aggregation. Previous methods [29, 36] generally utilize features from a short temporal window. But appearance deterioration could span a wide time window and thus makes temporal-based methods less effective. Moreover, the frames may be highly redundant in a short time window and consequently weaken the advantage of feature aggregation. To address this problem, we propose to aggregate feature from the semantic neighborhood, which is not susceptible to the appearance degradation lasting in time.

3.2. Sequence Level Semantics Aggregation

The ideal way for feature aggregation is to aggregate within the ground truth tracklet. But the golden association for proposals across 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 link proposals across space-time with their semantic similarities. This semantic feature based association approach is well known for its robustness to appearance change.

Semantic Guidance 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 semantic similarity between them with the generalized cosine similarity:

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

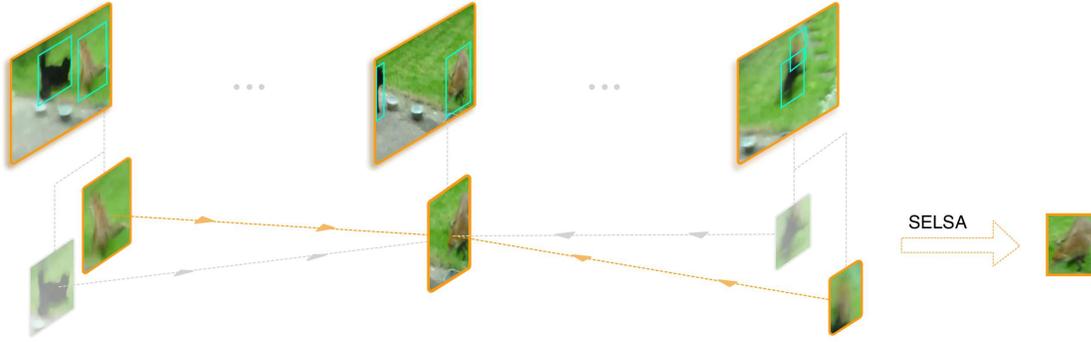


Figure 2. The overall architecture of the proposed model. We first extract proposals in different frames from the video, then the semantic similarities of proposals are computed across frames. At last, we aggregate the features from other proposals based on these similarities to obtain a more discriminative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

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

Feature Aggregation After defining the similarity between proposals, the semantic similarity now serves as guidance 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, 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:

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

where Ω is the set of frame indexes randomly selected for the aggregation. The SELSA module is fully 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. Figure 2 shows how the proposed SELSA module work.

3.3. A Spectral Clustering Viewpoint

Besides the simple and intuitive formulation of our method, we further reveal its close connection with the clas-

sic spectral clustering algorithm. This sheds light on how SELSA work from an intra-class variance reduction viewpoint.

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 the 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,

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

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 introduce the datasets and evaluation metrics used for VID 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 train our model with a mixture of ImageNet VID and DET datasets with the split provided in FGFA [36]. We evaluate our proposed method on ImageNet VID dataset [26]. We report the mAP@IoU=0.5 and motion-specific mAP on the validation set.

4.2. Implementation Details

Feature Network We use ResNet-101 [12] as the backbone network for ablation studies. ResNeXt-101-32 \times 4d [34] is also used for the final results. The total stride of *conv5* block is changed from 32 to 16 with dilated convolutions.

Detection Network RPN is applied on the output of *conv4*. Anchors of 3 scales and 3 aspect ratios are used. Then Fast R-CNN is applied on the output of *conv5*. We apply 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 \rightarrow SELSA \rightarrow FC \rightarrow SELSA). The general transformation functions in Eq. 1 are instantiated as one fully-connected layer.

Training and Testing Details The backbone networks are initialized with ImageNet pre-trained weights. A total of 220k iterations of SGD training is performed with a total batch size of 4 on 4 GPUs. The initial learning rate is 2.5×10^{-4} and is divided by 10 at the 110k and the

Component	(a)	(b)	(c)
Semantics Aggregation		✓	✓
Sequence-level Info			✓
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 the ImageNet VID validation set. For sequence-level methods, 21 frames are used when testing. No post-processing techniques are used. The absolute gains compared with the baseline are shown in the subscript.

165k iterations. For training, one training frame is sampled along with two random frames from the same video (identical frames for the 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 are considered as possible semantic neighbors for aggregation. This leads to a gain of 1.64 mAP compared with the baseline. When multiple objects with the same semantics or multiple proposals corresponding to the same object appear in the same frame, the semantically 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 full video sequence. It gives an mAP of 80.25, a large 6.63 mAP improvement compared with 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

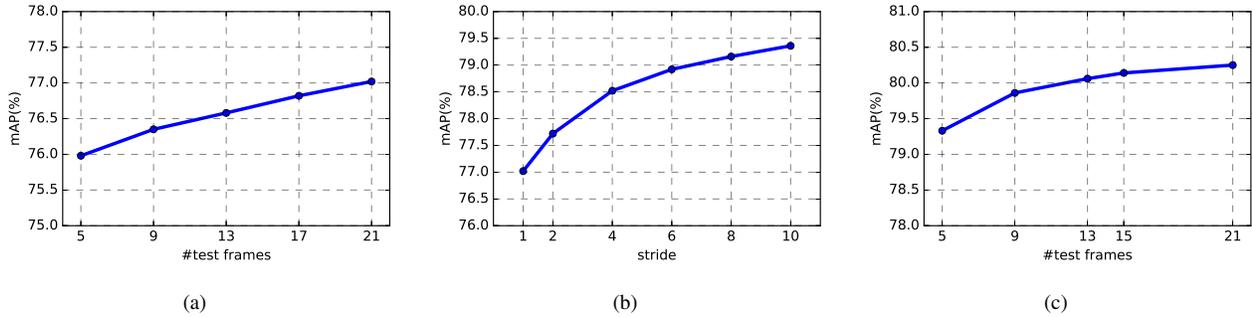


Figure 3. Ablation analyses of different test settings. (a) The effect of different number of frames on sequential test performance. (b) The effect of different sampling stride on sequential test performance. (c) The effect of different number of frames on shuffled test performance.

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 Sec. 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 feature aggregation during testing yields better results. Besides, [33] samples frames with a uniform stride during testing to improve the performance.

We examine the influence of the number of frames used and sampling strides when testing our method. Specifically, by using a sampling stride of S , one frame in every S frames is used for testing instead of consecutive frames.

First, we use sampling stride one and vary the number of frames used in aggregation. As seen in Figure 3(a), with more frames used for testing, the performance increases consistently. For example, using 21 frames for aggregation instead of 5 contributes a 1.04 mAP improvement.

We then fix the number of frames for aggregation to 21 and examine the impact of sampling stride. Figure 3(b) shows the performance with 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 may not 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 ob-

servation motivates the use of semantic neighbors instead of temporal neighbors. Thus, taking a step further, we sample semantic neighbors uniformly from the full 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 the full 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 semantic diversity. Frames in a video are high 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 backbone.

Backbone	ResNet-101		ResNeXt-101	
Seq-NMS		✓		✓
mAP (%)	82.69	82.48 _{↓0.21}	84.30	83.73 _{↓0.57}

Table 2. The effects of post-processing on our method. The absolute gains compared with the method without Seq-NMS are shown in the subscript.

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 the full-sequence level information. Nearly all the state-of-the-art video detection systems [36, 29, 6, 1, 33] adopted post-processing methods which gives huge gains in performance. To illustrate that our method has already captured the full-sequence level information, we further apply the Seq-NMS post-processing 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 full-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 validation 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.

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 the ImageNet VID validation set. * indicates use of video-level post-processing methods (e.g Seq-NMS, tubelet rescoring). + indicates use of model ensembling. [△] indicates using data augmentation.

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 plugging in the stronger 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 the same backbone and tubelet rescoring by a large 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 in the density and diversity of objects. Here we evaluate SELSA on the EPIC KITCHENS dataset [4].

5.1. Dataset and Evaluation Setup

EPIC KITCHENS [4] is a large scale egocentric dataset, capturing daily activities happened in the kitchens. In EPIC KITCHENS dataset, each frame contains avg/max 1.7/9 ob-

- [2] Zhe Chen, Shaoli Huang, and Dacheng Tao. Context refinement for object detection. In *ECCV*, 2018. 5
- [3] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-FCN: Object detection via region-based fully convolutional networks. In *NIPS*, 2016. 2
- [4] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will 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, 7, 8
- [5] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. FlowNet: Learning optical flow with convolutional networks. In *CVPR*, 2015. 3
- [6] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Detect to track and track to detect. In *ICCV*, 2017. 3, 6, 7
- [7] Cheng-Yang Fu, Wei Liu, Ananth Ranga, Amrith Tyagi, and Alexander C Berg. DSSD: Deconvolutional single shot detector. *arXiv:1701.06659*, 2017. 2
- [8] Ross Girshick. Fast R-CNN. In *CVPR*, 2015. 1, 2
- [9] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 2
- [10] Wei Han, Pooya Khorrami, Tom Le Paine, Prajit Ramachandran, Mohammad Babaeizadeh, Honghui Shi, Jianan Li, Shuicheng Yan, and Thomas S Huang. Seq-NMS for video object detection. *arXiv:1602.08465*, 2016. 3
- [11] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *ICCV*, 2017. 1
- [12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1, 5
- [13] Han Hu, Jiayuan Gu, Zheng Zhang, Jifeng Dai, and Yichen Wei. Relation networks for object detection. In *CVPR*, 2018. 2, 5
- [14] Kai Kang, Hongsheng Li, Junjie Yan, Xingyu Zeng, Bin Yang, Tong Xiao, Cong Zhang, Zhe Wang, Ruohui Wang, Xiaogang Wang, et al. T-CNN: Tubelets with convolutional neural networks for object detection from videos. *TCSVT*, 2017. 3
- [15] Kai Kang, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. Object detection from video tubelets with convolutional neural networks. In *CVPR*, 2016. 3
- [16] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Mallocci, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *arXiv:1811.00982*, 2018. 1
- [17] Tsung-Yi Lin, Priyal Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *TPAMI*, 2018. 2
- [18] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014. 1
- [19] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. SSD: Single shot multibox detector. In *ECCV*, 2016. 2, 6
- [20] Yongyi Lu, Cewu Lu, and Chi-Keung Tang. Online video object detection using association LSTM. In *ICCV*, 2017. 3
- [21] Marina Meila and Jianbo Shi. A random walks view of spectral segmentation. In *AISTATS*, 2001. 4, 5
- [22] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, 2016. 2
- [23] Joseph Redmon and Ali Farhadi. YOLO9000: better, faster, stronger. *ICCV*, 2017. 2
- [24] Joseph Redmon and Ali Farhadi. YOLOv3: An incremental improvement. *arXiv:1804.02767*, 2018. 2
- [25] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 1, 2
- [26] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *IJCV*, 2015. 5
- [27] Subarna Tripathi, Zachary C Lipton, Serge Belongie, and Truong Nguyen. Context matters: Refining object detection in video with recurrent neural networks. *arXiv:1607.04648*, 2016. 3
- [28] Jasper RR Uijlings, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *International journal of computer vision*, 104(2):154–171, 2013. 2
- [29] Shiyao Wang, Yucong Zhou, Junjie Yan, and Zhidong Deng. Fully motion-aware network for video object detection. In *ECCV*, 2018. 2, 3, 6, 7
- [30] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2018. 2
- [31] Xiaolong Wang and Abhinav Gupta. Videos as space-time region graphs. *ECCV*, 2018. 5
- [32] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *ICIP*, 2017. 3
- [33] Fanyi Xiao and Yong Jae Lee. Video object detection with an aligned spatial-temporal memory. In *ECCV*, 2018. 3, 6, 7
- [34] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017. 1, 5
- [35] Xizhou Zhu, Jifeng Dai, Lu Yuan, and Yichen Wei. Towards high performance video object detection. In *CVPR*, 2018. 3
- [36] Xizhou Zhu, Yujie Wang, Jifeng Dai, Lu Yuan, and Wei Yichen. Flow-guided feature aggregation for video object detection. In *ICCV*, 2017. 1, 2, 3, 5, 6, 7
- [37] Xizhou Zhu, Yuwen Xiong, Jifeng Dai, Lu Yuan, and Yichen Wei. Deep feature flow for video recognition. In *CVPR*, 2017. 2, 3, 6