Supplementary Material: Spatio-Temporal Learnable Proposals for End-to-End Video Object Detection

Khurram Azeem Hashmi[1,2]
khurram_azeem.hashmi@dfki.de
Didier Stricker[1,2]
didier.stricker@dfki.de
Muhammamd Zeshan Afzal[1,2]
muhammamd_zeshan.afzal@dfki.de

[1] German Research Centre for Artificial Intelligence (DFKI)
Kaiserslautern, Germany
[2] Technical University of Kaiserslautern
Germany

Overview. The supplementary material is organized as follows: Section 1 revisits the Sparse R-CNN [15], which is used as a still image detector in our method; Section 2 reports additional details of our experimental setup to reproduce the results; Section 3 presents additional yet important ablation studies omitted in the main paper due to space constraints. Section 4 provides a qualitative analysis of our method; Section 5 analyses the failure cases of our SparseVOD.

1 Revisiting Sparse R-CNN

Sparse R-CNN [15] has emerged as a strong baseline for object detection in still images by replacing dense predictions from Region Proposal Network (RPN) with a small set of candidate regions. It adopts an iterative architecture based on a Dynamic head to predict and enhance the predictions progressively. Each iterative stage takes multi-scale feature maps from the FPN-based ResNet backbone [7, 10], proposal boxes, and their corresponding proposal features. Alternative to predictions from RPN, proposal boxes are a small fixed set of learnable candidate regions $(N_p \times 4)$, highlighting the possible locations of objects in the image. Proposal features are high dimensional latent vectors $(N_p \times C)$, representing rich instance attributes of each proposal box like object pose and shape. The RoIAlign [8] operation is performed on each proposal box to extract RoI features. Then, each RoI feature and the corresponding proposal feature are fed to the proposed dynamic instance interactive head to learn optimal object features for classification and regression. Each stage returns predicted boxes, corresponding categories, and object features of the boxes. The predicted boxes and object features from one stage are enhanced input proposal boxes and proposal features for the next stage, respectively. After each stage, set prediction loss [1] is computed on the fixed number of predictions to achieve the best bipartite matching among prediction and ground truth objects. Due to the purely sparse design and object proposal learning capabilities, we employ Sparse R-CNN as our still image detector baseline.

2 Experimental Setup

2.1 Dataset and Evaluation Metrics.

We conduct experiments on the ImageNet VID dataset [14], which comprises 3862 training videos and 555 validation videos. Following prior works [3, 17, 19], we train our model on a combination of ImageNet VID and DET datasets and evaluate the results on the validation set. Besides evaluating the performance on the mean average precision (mAP) @IoU=0.5 as in [19, 20], we compute mAPs @IoU=0.75 and @IOU=0.5:95 as in [10] to compare the robustness of our SparseVOD with previous state-of-the-art methods.

2.2 Implementation Details.

Our implementation is based on MMTracking [2] and PyTorch. Analogous to [15], we use AdamW [12] optimizer with a weight decay of 10^{-4}. We train our network for 12 epochs with a batch size of 8 on 8 GPUs. Initially, the learning rate is set to 2.5×10^{-5} and divided by 10 at the 8-th and 11-th epochs. Following [1, 15, 21], we set $\lambda_{cls} = 2, \lambda_{L1} = 5, and \lambda_{giou} = 2$. We follow the basic settings of [15] and set the number of iterative stages, proposal boxes, and the corresponding proposal features to 6, 100, and 100, respectively. We adopt ImageNet pre-trained [4] ResNet-50 [7], ResNet-101, and ResNeXt-101 [18] to compare performance with SOTA methods. We follow identical frame sampling settings for the target and support frames as employed in [16, 17] during the inference for direct comparison. Furthermore, we do not require any complex post-processing methods such as NMS, which simplifies the overall pipeline of SparseVOD.

3 More Ablation Studies

Similar to [9], all experiments are conducted on ImageNet VID dataset with ResNet-50 as the backbone network. The run time (FPS) is tested on a single DGX A100 GPU.

3.1 Proposal Initialization Scheme

Following [15], we conduct ablation studies to assess the impact of the proposal initialization scheme on the performance of our method in Table 1. We simply adopt four different initialization strategies. (1) Center: all object proposals are initialized at the centre. The height and width of the boxes are one-tenth of the frame size. (2) Image: the size of all object proposals is equal to the size of the frame. (3) Grid: initializing proposals as a grid by adopting the grid initialization strategy of [13]. (4) Random: adopting Gaussian distribution to randomly initialize centre, width, and height of region proposals. The results in Table 1 demonstrate that the detection performance of our method is relatively independent of the proposal initialization technique.

Initialization	$AP_{50}(\%)$	$AP_{75}(\%)$	$AP_{50:95}(\%)$
Center	80.1	59.9	53.9
Image	80.5	60.3	54.8
Grid	80.0	59.7	53.7
Random	80.3	60.1	54.7

Table 1: Ablation on proposal initialization method. Note that the proposal box initialization technique does not contribute to the detection performance.

3.2 Effect of Number of Proposals

We also investigate the effect of the number of proposals on our method in Table 2. Analogous to [15], we observe a direct relationship between the rise in proposals to the achieved performance. However, increasing proposals from 100 to 300 significantly increases the run time due to the involved spatio-temporal feature aggregation between video frames. Thus, we choose 100 proposals as the best tradeoff in the default settings.

Proposals	$AP_{50}(\%)$	$AP_{75}(\%)$	FPS
100	80.3	60.1	14.4
300	80.7	60.5	7.8
500	80.9	61.3	5.5

Table 2: Effect of number of proposals.

3.3 Impact of Number of Stages on Increasing IoU thresholds

Although we summarize the impact of the number of stages on the performance in the main paper (Section 4.4), here, we intend to investigate performance on increasing IoU thresholds ($0.5 \geq \text{IoU} \leq 0.95$). As shown in Figure 1, even with a number of stages set to 3, our SparseVOD already reaches comparable $AP_{50:95}$ of 52.7%. Note that as presented in the main paper (Table 1), the previous best competitor TROI [5] achieves $AP_{50:95}$ of 52.8% with a run time of 7.5 FPS. The detection performance of our SparseVOD keeps increasing with the rise in the number of stages and finally stabilizes with 6 iterative stages after accomplishing 57.7% $AP_{50:95}$. Thanks to the Spatio-temporal proposal learning, our SparseVOD attains comparable performance on increasing IoU thresholds with a sparse set of object proposals while yielding a run time of 23.7 FPS (52.7% $AP_{50:95}$, number of stages = 3).

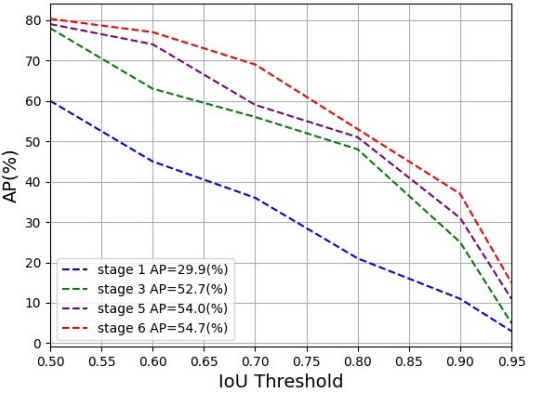

Figure 1: The detection performance of our method at different stages under varying IoU threshold.

4 Qualitative Comparison

We present a qualitative comparison between a single-frame baseline [15], the previous state-of-the-art proposal-based VOD approach [5], and our proposed SparseVOD in Figure 2. The single-frame detector misses objects (Watercraft) and misclassifies (Squirrel as Fox) in low-quality frames, as depicted in (a) and (b), respectively. Despite leveraging spatio-temporal feature aggregation, the prior proposal-based method TROI [5] (built upon SELSA [17]) overlooks detection (Watercraft) in Figure 2(a) and yields false positive (Squirrel as Fox) in Figure 2(b). We argue that since these methods generate unreliable object proposals on low-quality frames (3rd and 4th column in Figure 2(a), the spatio-temporal proposal feature aggregation produces sub-optimal proposal features for the target frame. Alternatively, our SparseVOD exploits the spatio-temporal feature aggregation to generate proposals. This ensures optimal proposal features even on low-quality frames, which not only resolves missed detection (Watercraft) in Figure 2(a) but also alleviates misclassification (Fox to Squirrel) in Figure 2(b).

5 Failure Case Analysis

Although the proposed SparseVOD simplifies the overall VOD pipeline and provides high-quality detections, it fails in some cases. A couple of such scenarios are illustrated in Figure 3. We observe that our method either overlooks or misclassifies objects suffering from rare poses and occlusions in the entire video. In the second and third frames of the first row, the detector misses a car (highlighted in pink) because of the rare pose challenge where only a fragment of an object is visible. The second row depicts missed (in pink) and false detections (in blue). This is due to the large occlusion of zebras in the entire video. Despite exploiting temporal information, the model yields uncertain predictions, leading to missed detections or false classification. We believe that more optimized temporal modelling, such as incorporating inter-video context [6] in our SparseVOD to generate object proposals, is required to further this research.

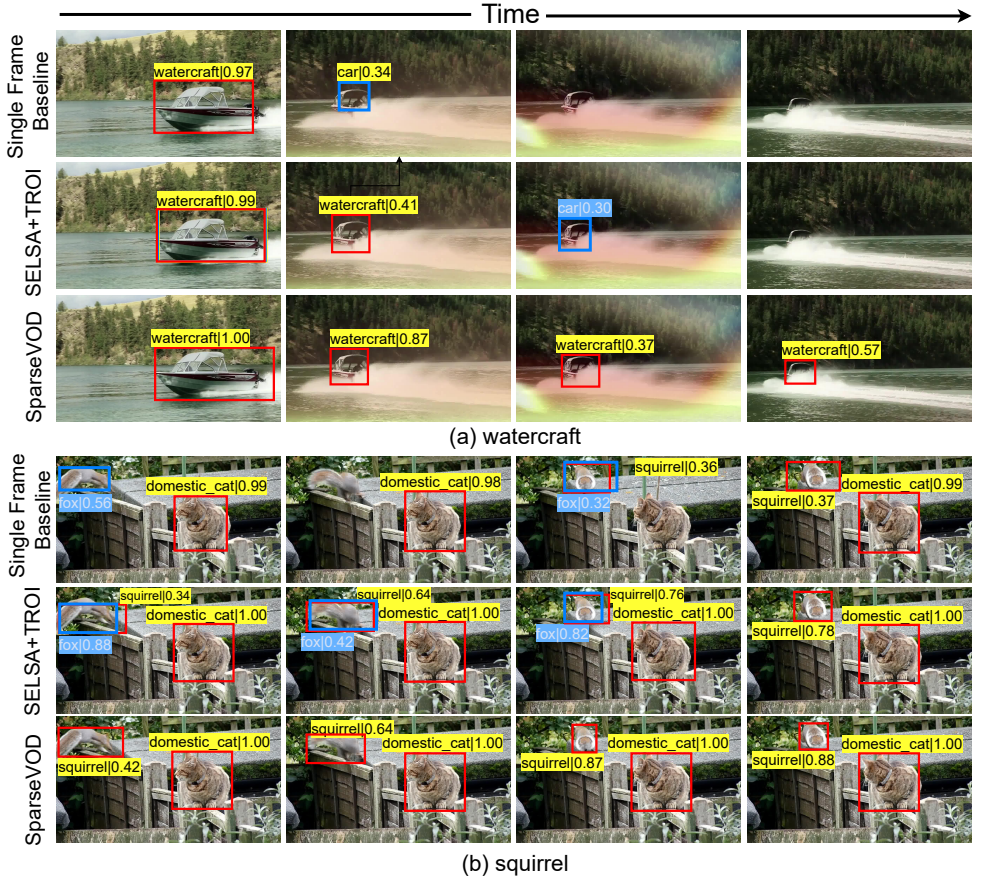

Figure 2: Qualitative Comparison. Correct and missed predictions are highlighted in red and blue, respectively. For each video, the first row represents detections from a single-frame baseline [ ]. The second row depicts results from a recent proposal-based method [ ], whereas the third row depicts results from our proposed SparseVOD. Thanks to spatio-temporal learnable proposals, our SparseVOD effectively recognizes missed detections and resolves misclassifications in low-quality frames, as illustrated in (a). Similarly, in (b), the iterative proposal learning not only corrects misclassification from proposal-based methods but also enhances localization, leading to more accurate predictions.

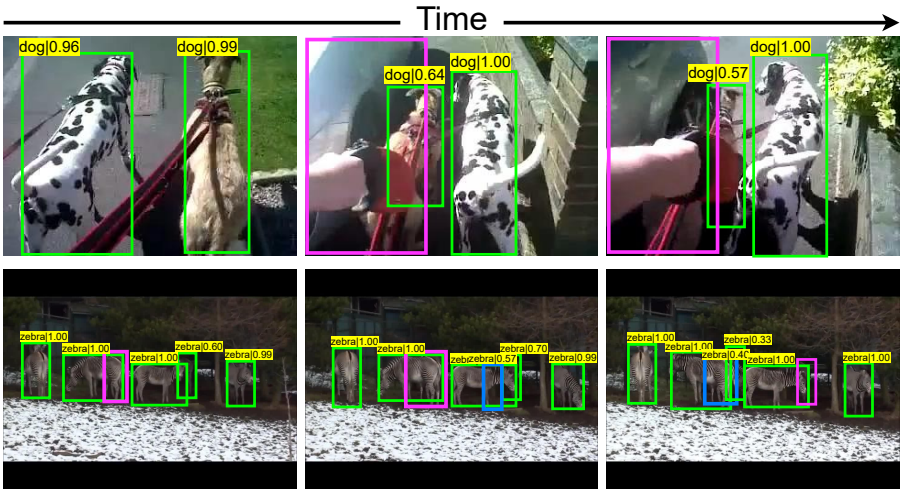

Figure 3: Failure case analysis. The green highlights correct detections, whereas pink and blue depict missed and false detections. The results are achieved on our SparseVOD with ResNeXt-101 as the backbone network.

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[2] MMTracking Contributors. MMTracking: OpenMMLab video perception toolbox and benchmark. https://github.com/open-mmlab/mmtracking, 2020.

[3] Yiming Cui, Liqi Yan, Zhiwen Cao, and Dongfang Liu. Tf-blender: Temporal feature blender for video object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8138–8147, 2021.

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[5] Tao Gong, Kai Chen, Xinjiang Wang, Qi Chu, Feng Zhu, Dahua Lin, Nenghai Yu, and Huamin Feng. Temporal roi align for video object recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1442–1450, 2021.

[6] Mingfei Han, Yali Wang, Xiaojun Chang, and Yu Qiao. Mining inter-video proposal relations for video object detection. In *European conference on computer vision*, pages 431–446. Springer, 2020.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[8] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[9] Lu He, Qianyu Zhou, Xiangtai Li, Li Niu, Guangliang Cheng, Xiao Li, Wenxuan Liu, Yunhai Tong, Lizhuang Ma, and Liqing Zhang. End-to-end video object detection with spatial-temporal transformers. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 1507–1516, 2021.

[10] 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 *European conference on computer vision*, pages 740–755. Springer, 2014.

[11] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[12] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[13] Mahyar Najibi, Mohammad Rastegari, and Larry S Davis. G-cnn: an iterative grid based object detector. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2369–2377, 2016.

[14] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115 (3):211–252, 2015.

[15] Peize Sun, Rufeng Zhang, Yi Jiang, Tao Kong, Chenfeng Xu, Wei Zhan, Masayoshi Tomizuka, Lei Li, Zehuan Yuan, Changhu Wang, et al. Sparse r-cnn: End-to-end object detection with learnable proposals. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14454–14463, 2021.

[16] Chi Wang, Yang Hua, Zheng Lu, Jian Gao, and Neil Robertson. Temporal meta-adaptor for video object detection. In *British Machine Vision Conference 2021*, 2021.

[17] Haiping Wu, Yuntao Chen, Naiyan Wang, and Zhaoxiang Zhang. Sequence level semantics aggregation for video object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9217–9225, 2019.

[18] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017.

[19] Xizhou Zhu, Yujie Wang, Jifeng Dai, Lu Yuan, and Yichen Wei. Flow-guided feature aggregation for video object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 408–417, 2017.

[20] Xizhou Zhu, Yuwen Xiong, Jifeng Dai, Lu Yuan, and Yichen Wei. Deep feature flow for video recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2349–2358, 2017.

[21] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.