Supplementary Material

A. Experimental details

We summarize the OOD detection evaluation task in Table 6. The OOD test dataset is selected from MS-COCO and nuImages dataset, which contains disjoint labels from the respective ID dataset. For the Youtube-VIS dataset, we use the dataset released in year 2021. Since there are no ground truth labels available for the validation images, we select the last 597 videos in the training set as the in-distribution evaluation dataset. The remaining 2,388 videos are used for training. The BDD100K and Youtube-VIS model are both trained for a total of 52,500 iterations. See detailed ablations on the hyperparameters in Section 4.3 of the main paper.

<table>
<thead>
<tr>
<th>Task 1</th>
<th>Task 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID train dataset</td>
<td>BDD100K train</td>
</tr>
<tr>
<td>ID val dataset</td>
<td>BDD100K val</td>
</tr>
<tr>
<td>OOD dataset</td>
<td>COCO / nuImages</td>
</tr>
<tr>
<td>#ID train images</td>
<td>273,406</td>
</tr>
<tr>
<td>#ID val images</td>
<td>39,973</td>
</tr>
<tr>
<td>#OOD images from COCO</td>
<td>1,914</td>
</tr>
<tr>
<td>#OOD images from nuImages</td>
<td>2,100</td>
</tr>
<tr>
<td>#OOD images from Youtube-VIS train</td>
<td>67,861</td>
</tr>
<tr>
<td>#OOD images from Youtube-VIS val</td>
<td>21,889</td>
</tr>
<tr>
<td>#OOD images from Youtube-VIS val</td>
<td>28,922</td>
</tr>
<tr>
<td>#OOD images from Youtube-VIS val</td>
<td>2,100</td>
</tr>
</tbody>
</table>

Table 6. OOD detection evaluation tasks.

B. In-distribution classes

We provide a detailed description of the in-distribution classes for the two video datasets as follows.

BDD100K dataset contains 8 classes, which are pedestrian, rider, car, truck, bus, train, motorcycle, bicycle.

The Youtube-VIS dataset contains 40 classes, which are airplane, bear, bird, boat, car, cat, cow, deer, dog, duck, earless seal, elephant, fish, flying disc, fox, frog, giant panda, giraffe, horse, leopard, lizard, monkey, motorbike, mouse, parrot, person, rabbit, shark, skateboard, snake, snowboard, squirrel, surfboard, tennis racket, tiger, train, truck, turtle, whale, zebra.

C. Software and hardware

We run all experiments with Python 3.8.5 and PyTorch 1.7.0, using NVIDIA GeForce RTX 2080Ti GPUs.

D. Baselines

To evaluate the baselines, we follow the original methods in MSP [17], ODIN [33], Generalized ODIN [20], Mahalanobis distance [31], CSI [59], energy score [36] and gram matrices [54] and apply them accordingly on the classification branch of the object detectors. For ODIN [33], the temperature is set to be $T = 1000$ following the original work. For both ODIN and Mahalanobis distance [31], the noise magnitude is set to 0 because the region-based object detector is not end-to-end differentiable given the existence of region cropping and ROIAlign. For GAN [30], we follow the original paper and use a GAN to generate OOD images. The prediction of the OOD images/objects is regularized to be close to a uniform distribution, through a KL divergence loss with a weight of 0.05. We set the shape of the generated images to be 100×100 and resize them to have the same shape as the real images. We optimize the generator and discriminator using the Adam optimizer [26], with a learning rate of 0.001. For CSI [59], we use the rotations (0°, 90°, 180°, 270°) as the self-supervision task. We set the temperature in the contrastive loss to 0.5. We use the features right before the classification branch (with the dimension to be 1024) to perform contrastive learning. The weights of the losses that are used for classifying shifted instances and instance discrimination are both set to 0.1 to prevent training collapse. For Generalized ODIN [20], we replace and train the classification head of the object detector by the most effective Deconf-C head shown in the original paper.

E. Ablation study on a different backbone architecture

In this section, we evaluate the proposed STUD using a different backbone architecture of the Faster-RCNN, which is RegNetX-4.0GF [49]. Similarly, we compare with the same set of OOD detection baselines as stated in the main paper. The
<table>
<thead>
<tr>
<th>In-distribution $\mathcal{D}$</th>
<th>Method</th>
<th>FPR95 ↓</th>
<th>AUROC ↑</th>
<th>mAP (ID) ↑</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>OOD: MS-COCO / nullages</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BDD100K</td>
<td>MSP [17]</td>
<td>80.09 / 93.05</td>
<td>74.19 / 63.14</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td>ODIN [33]</td>
<td>64.74 / 82.08</td>
<td>77.65 / 67.09</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td>Mahalanobis [31]</td>
<td>54.02 / 79.85</td>
<td>82.38 / 75.48</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td>Gram matrices [54]</td>
<td>63.96 / 63.61</td>
<td>67.56 / 67.47</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td>Energy score [36]</td>
<td>64.79 / 81.62</td>
<td>78.78 / 69.43</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td>Generalized ODIN [20]</td>
<td>60.76 / 82.00</td>
<td>80.14 / 70.74</td>
<td>32.5</td>
</tr>
<tr>
<td></td>
<td>CSI [59]</td>
<td>52.98 / 80.00</td>
<td>83.57 / 74.91</td>
<td>31.8</td>
</tr>
<tr>
<td></td>
<td>GAN-synthesis [30]</td>
<td>58.35 / 83.65</td>
<td>81.43 / 70.39</td>
<td>31.5</td>
</tr>
<tr>
<td></td>
<td><strong>STUD</strong> (ours)</td>
<td><strong>52.51 / 79.75</strong></td>
<td><strong>84.03 / 76.55</strong></td>
<td><strong>32.3</strong></td>
</tr>
<tr>
<td>Youtube-VIS</td>
<td>MSP [17]</td>
<td>89.86 / 97.42</td>
<td>67.04 / 54.02</td>
<td>26.7</td>
</tr>
<tr>
<td></td>
<td>ODIN [33]</td>
<td>89.28 / 96.30</td>
<td>67.54 / 60.82</td>
<td>26.7</td>
</tr>
<tr>
<td></td>
<td>Mahalanobis [31]</td>
<td>90.00 / 94.44</td>
<td>70.47 / 54.83</td>
<td>26.7</td>
</tr>
<tr>
<td></td>
<td>Gram matrices [54]</td>
<td>87.64 / 91.25</td>
<td>69.76 / 61.43</td>
<td>26.7</td>
</tr>
<tr>
<td></td>
<td>Energy score [36]</td>
<td>88.54 / 90.21</td>
<td>67.83 / 58.02</td>
<td>26.7</td>
</tr>
<tr>
<td></td>
<td>Generalized ODIN [20]</td>
<td>85.15 / 98.00</td>
<td>71.57 / 64.23</td>
<td>27.3</td>
</tr>
<tr>
<td></td>
<td>CSI [59]</td>
<td>82.43 / 88.61</td>
<td>71.81 / 54.00</td>
<td>24.2</td>
</tr>
<tr>
<td></td>
<td>GAN-synthesis [30]</td>
<td>85.75 / 93.75</td>
<td>72.95 / 56.94</td>
<td>25.5</td>
</tr>
<tr>
<td></td>
<td><strong>STUD</strong> (ours)</td>
<td><strong>81.14 / 80.77</strong></td>
<td><strong>74.82 / 69.52</strong></td>
<td><strong>27.2</strong></td>
</tr>
</tbody>
</table>

Table 7. Comparison with competitive out-of-distribution detection methods. All baseline methods are based on a model trained on ID data only using RegNetX-4.0GF as the backbone. ↑ indicates larger values are better, and ↓ indicates smaller values are better. All values are percentages. **Bold** numbers are superior results.

results are shown in Table 7.

From Table 7, we demonstrate that STUD is effective on alternative neural network architectures. In particular, using RegNet [49] as backbone yields better OOD detection performance compared with the baselines. Moreover, we show that STUD achieves stronger OOD detection performance while preserving or even slightly increasing the object detection accuracy on ID data (measured by mAP). This is in contrast with CSI, which displays significant degradation, with mAP decreasing by 3% on Youtube-VIS.

F. Additional related work

Video anomaly detection (VAD) aims to identify anomalous events on both the object level [7, 22, 68] and frame level [35, 39, 51] by techniques such as skeleton trajectory modeling [43], weakly supervised learning [69], attention [47], temporal pose graph [38], self-supervised learning [10] and autoencoders [3]. Compared with STUD, the anomalies in VAD do not necessarily have different semantics from the ID training data. Moreover, none of the approaches considered synthesizing unknowns with the help of videos or energy-based model regularization.

G. Additional visualization examples

We provide additional visualization of the detected objects on different OOD datasets with models trained on different in-distribution datasets. The results are shown in Figures 7-10.
Figure 7. Additional visualization of detected objects on the OOD images (from MS-COCO) by a vanilla Faster-RCNN (top) and STUD (bottom). The in-distribution is BDD100K dataset. **Blue:** Objects detected and classified as one of the ID classes. **Green:** OOD objects detected by STUD, which reduce false positives among detected objects.
Figure 8. Additional visualization of detected objects on the OOD images (from nuImages) by a vanilla Faster-RCNN (top) and STUD (bottom). The in-distribution is BDD100K dataset. **Blue**: Objects detected and classified as one of the ID classes. **Green**: OOD objects detected by STUD, which reduce false positives among detected objects.
Figure 9. Additional visualization of detected objects on the OOD images (from MS-COCO) by a vanilla Faster-RCNN (top) and STUD (bottom). The in-distribution is Youtube-VIS dataset. **Blue**: Objects detected and classified as one of the ID classes. **Green**: OOD objects detected by STUD, which reduce false positives among detected objects.
Figure 10. Additional visualization of detected objects on the OOD images (from nuImages) by a vanilla Faster-RCNN (top) and STUD (bottom). The in-distribution is Youtube-VIS dataset. **Blue:** Objects detected and classified as one of the ID classes. **Green:** OOD objects detected by STUD, which reduce false positives among detected objects.
References


