Supplementary for Anomaly Detection in Video via Self-Supervised and Multi-Task Learning

Mariana-Iuliana Georgescu[1,3], Antonio Bărbălău[1], Radu Tudor Ionescu[1,3], Fahad Shahbaz Khan[2], Marius Popescu[1,3], Mubarak Shah[4]
[1]University of Bucharest, Romania, [2]MBZ University of Artificial Intelligence, Abu Dhabi
[3]SecurifAI, Romania, [4]University of Central Florida, Orlando, FL

Abstract

In the supplementary, we include additional examples of frame-level scores predicted by our object-centric framework. Along with the frame-level scores, we also show anomaly localization examples in specific frames. Besides showing correct detections, we also include a set of false positive and false negative examples. Moreover, the supplementary provides details about the running time and a discussion about the reliance on object detectors and the chosen proxy tasks. Along with the figures included in this document, we attach test videos with annotations made by our object-centric anomaly detection system.

1. Additional Results

1.1. Qualitative Results

The supplementary results are structured as follows. Figure 1 illustrates a set of true positive, false positive and false negative examples extracted from our runs on the benchmark data sets. Figures 2 and 3 showcase the overlap between our frame-level anomaly predictions and the ground-truth labels for two videos from Avenue. Similarly, Figures 4 and 5 illustrate the overlap between our frame-level anomaly predictions and the ground-truth labels for two ShanghaiTech videos. Finally, Figures 6, 7 and 8 showcase our frame-level performance for three UCSD Ped2 videos.

Avenue. Our framework reaches a state-of-the-art frame-level AUC performance of 92.8% on the Avenue data set, being able to detect anomalies such as: (i) the two, mostly overlapped, individuals dressed in white preforming a dance on one side of the scene, (ii) the child dressed in red that was moving very close to the camera and (iii) the man running on the main alley, all shown in Figure 1 (top row). Aside from these true positive detections, we present a false positive example of two people that act strangely. In this specific instance, the security agent that took a stance in front of the main alley was wrongly labeled as anomalous, probably because this behavior is not observed during training. Finally, due to the detection failure of the object detector, our framework is not able to label the backpack thrown in the air as an anomaly, generating the false negative illustrated in Figure 1 (top row). This deficiency is compensated by recognizing that the gesture of throwing a backpack into the air performed by the human is indeed anomalous. Figure 2 illustrates how our framework is able to capture the gesture of throwing, labeling the individual as anomalous. Our framework reaches an almost perfect frame-level AUC performance of 99.88% on the fifth test video from the Avenue data set. Additionally, Figure 3 showcases how our framework is able to detect other object-related anomalies. In this instance, our anomaly score starts to increase as the bike appears in the scene. Our method reports it as a clear anomalous occurrence as it becomes fully visible and moves towards the camera.

ShanghaiTech. On ShanghaiTech our framework is able to correctly identify most vehicle-related anomalies. As show in Figure 1 (second row), objects such as cars and bicycles are regularly labeled as anomalies. However, in the specific scenario presented as false negative in Figure 1 (second row), a bicycle that was used by two individuals simultaneously managed to pass as a normal event. Aside from vehicles, our framework also labels strange (meaning not previously seen) objects as anomalies when encountered. Accordingly, in the false positive example, the umbrella was detected and labeled as anomalous. Figures 4 and 5 showcase our anomaly score predictions together with the frame-level ground-truth labels for test videos *06_0144* and *12_0149* from ShanghaiTech, respectively. In the first instance, our method correctly identifies the car as an anomaly, reaching a frame-level AUC of 98.97%, while in the second instance, our framework accurately identifies the individual running behind the group as abnormal, reaching a frame-level AUC of 98.51%.

Figure 1. True positive, false positive and false negative examples from Avenue (top row), ShanghaiTech (second row) and UCSD Ped2 (bottom row). Best viewed in color.

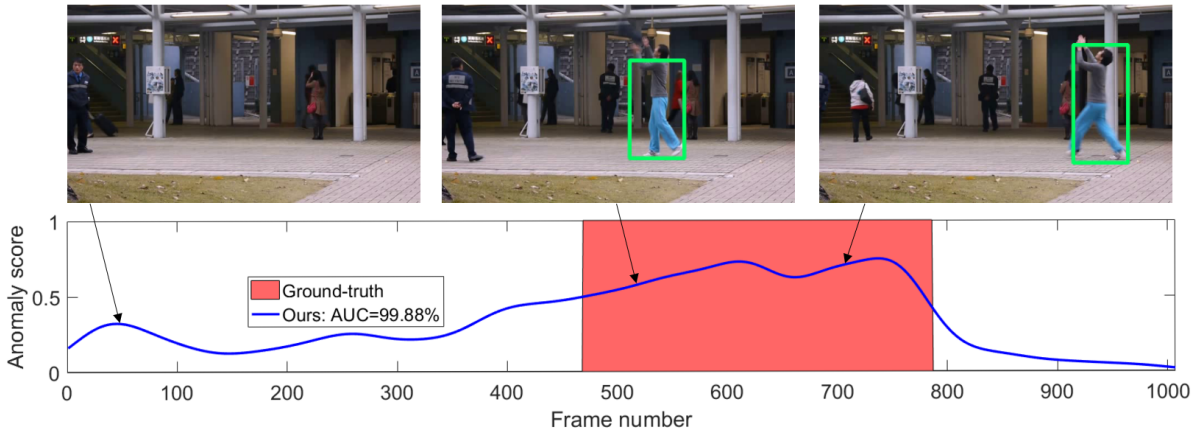

Figure 2. Frame-level scores and anomaly localization examples for test video 05 from Avenue. Best viewed in color.

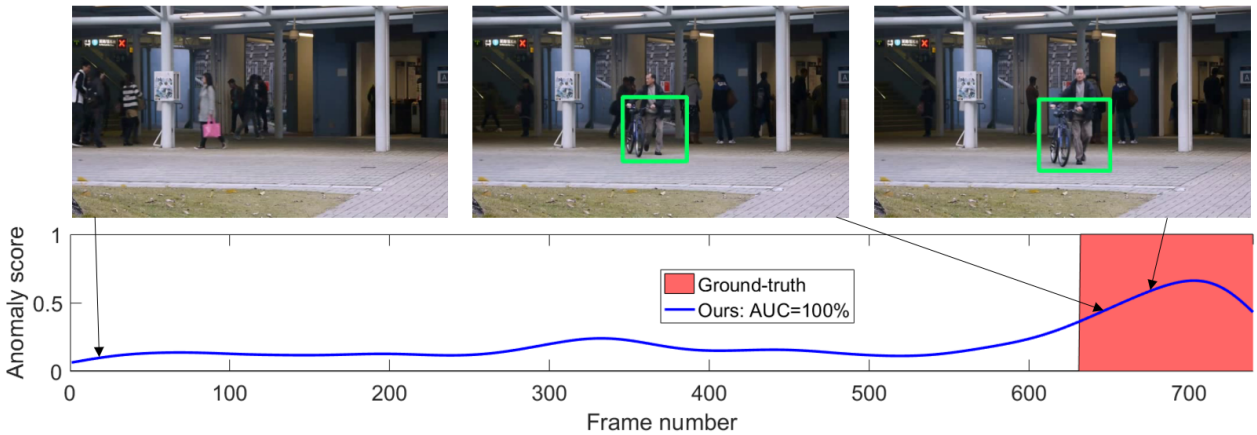

Figure 3. Frame-level scores and anomaly localization examples for test video 16 from Avenue. Best viewed in color.

UCSD Ped2. On UCSD Ped2, our method reaches a frame-level AUC of 99.8%, accurately and almost perfectly capturing all anomalous events such as people riding bicycles among the crowd or vehicles making an appearance in the pedestrian area. Objects are missed only in very few particular frames, such as when the bike did not completely

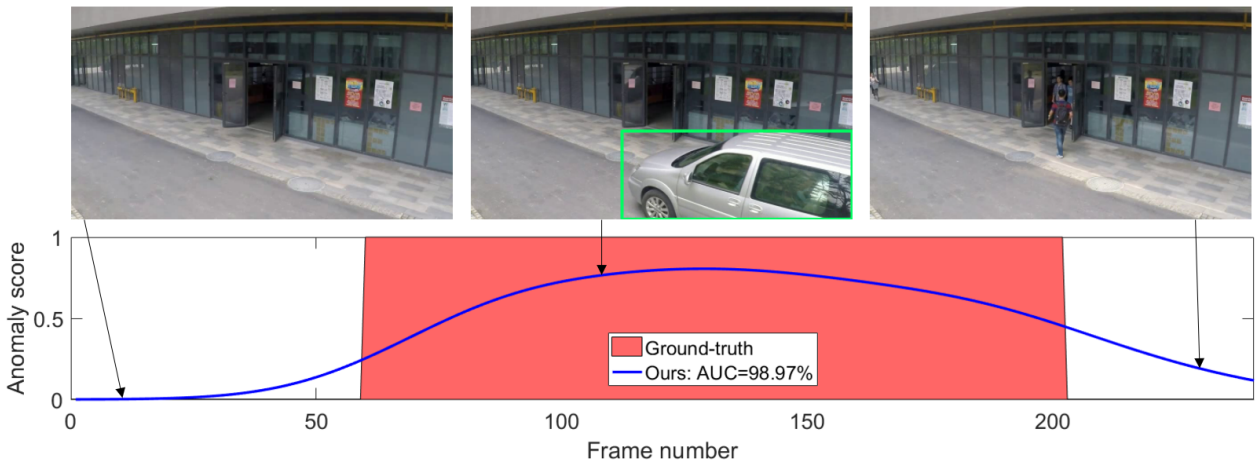

Figure 4. Frame-level scores and anomaly localization examples for test video 06_0144 from ShanghaiTech. Best viewed in color.

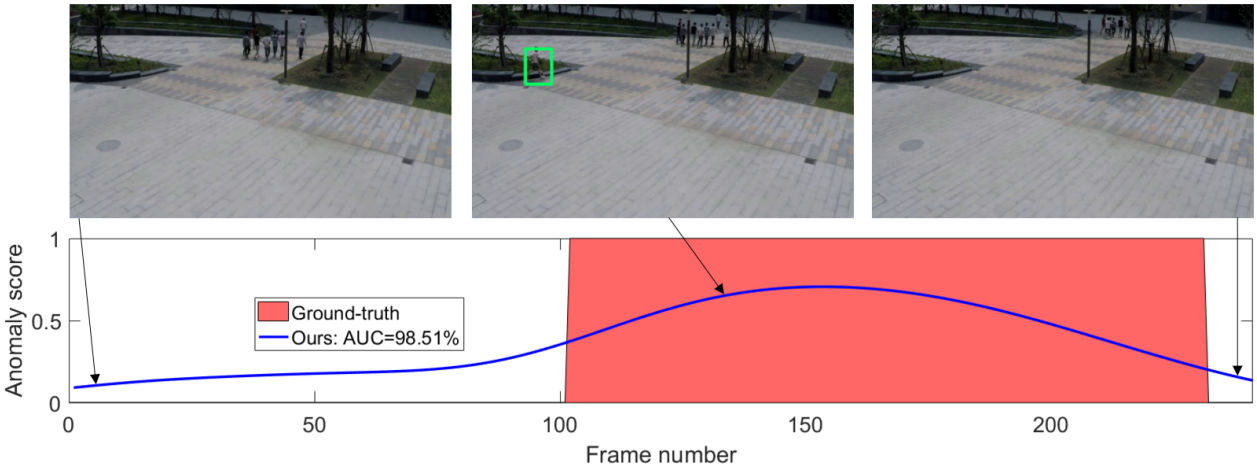

Figure 5. Frame-level scores and anomaly localization examples for test video 12_0149 from ShanghaiTech. Best viewed in color.

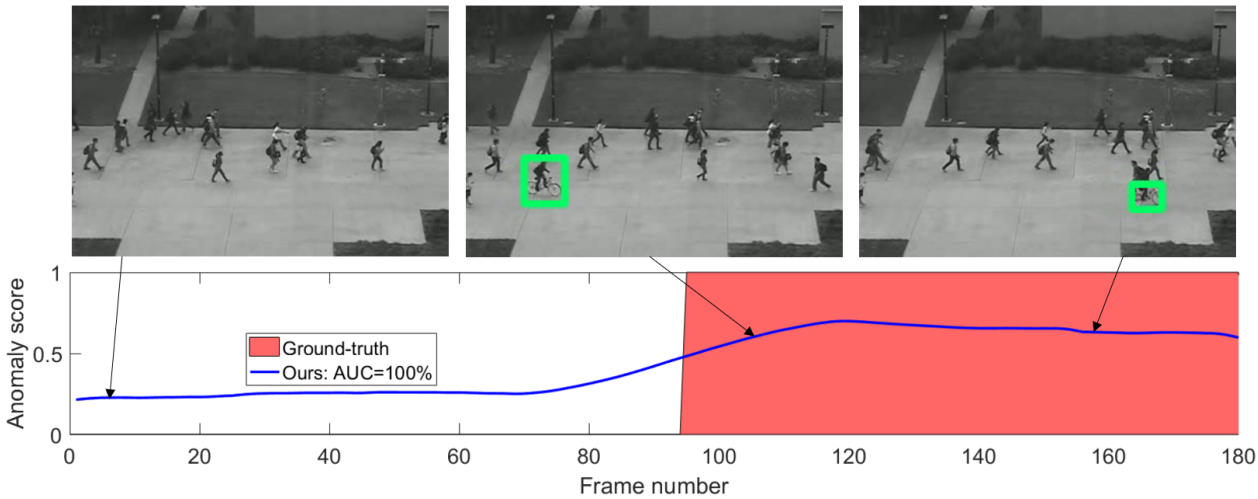

Figure 6. Frame-level scores and anomaly localization examples for test video 02 from UCSD Ped2. Best viewed in color.

entered the scene (being truncated), shown as the false negative example from UCSD Ped2 in Figure 1 (bottom row). In addition, the individual featured as the false positive leaving the alley through the camera-facing exit is also wrongly labeled as an anomaly. Figures 6 and 7 showcase the general performance of our method on the UCSD Ped2 data set,

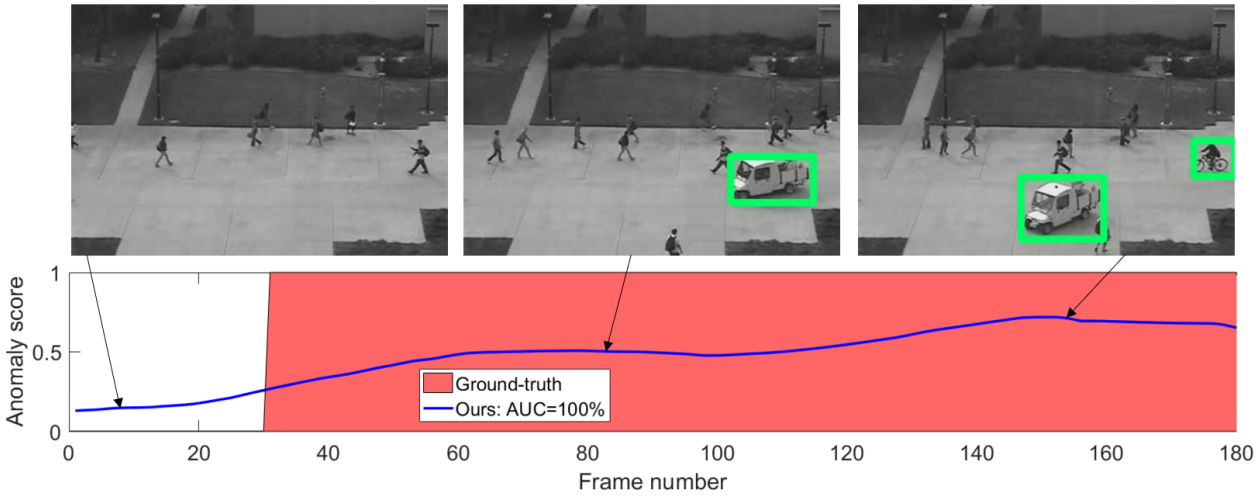

Figure 7. Frame-level scores and anomaly localization examples for test video 04 from UCSD Ped2. Best viewed in color.

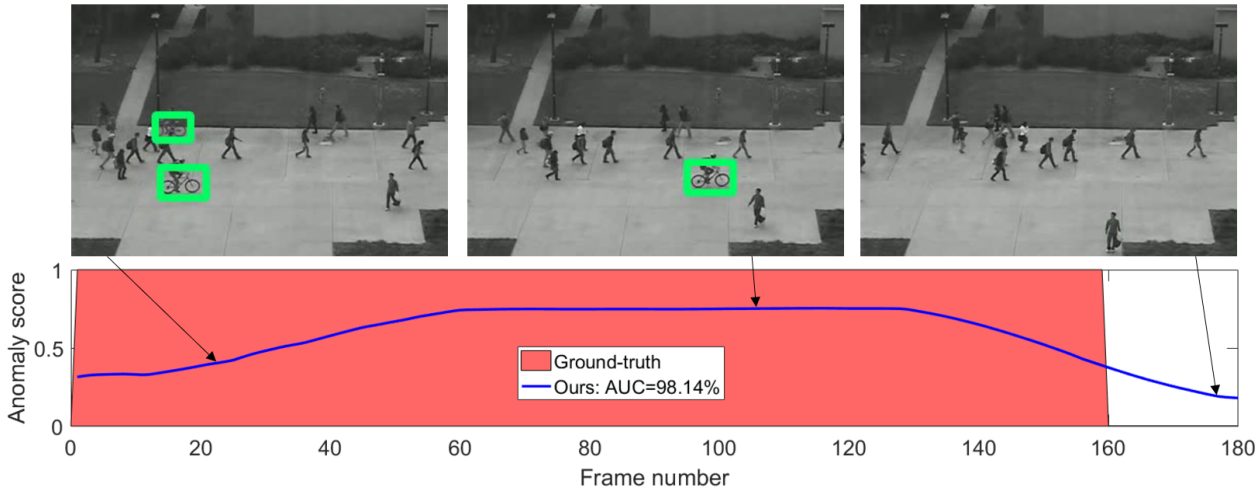

Figure 8. Frame-level scores and anomaly localization examples for test video 06 from UCSD Ped2. Best viewed in color.

reaching perfect frame-level AUC scores.

1.2. Running Time

Our lightweight model infers the anomaly score of a single object in 6 milliseconds (ms). The YOLOv3 model takes 26 ms per frame to detect the objects. Reassembling the anomaly map from the object-level anomaly scores takes less than 1 ms. With all components in place, our framework runs at 23 FPS with an average of 5 objects per frame. The reported time includes only the object-level inference, which is the most heavy part (due to the object detector). When we add the frame-level inference, the speed decreases by a small margin, from 23 FPS to 21 FPS. The FPS rates are measured on a single GeForce GTX 1080Ti GPU with 11GB of VRAM.

2. Discussion

Dependence on object detector. We note that object-centric methods are influenced by the quality of object detectors. For example, on Avenue, we observed that our object-centric method does not detect papers (*paper* is not in the COCO set of classes) or backpacks thrown in the air (*backpack* is in the COCO set of classes, but the detector fails due to motion blur). Despite not explicitly detecting papers or backpacks, the detector detects the person throwing these objects and our framework labels the respective person as abnormal. The same can happen in the case of fire or explosion, if there is a person nearby that runs away from the fire or that is thrown on the ground by the blast. A pure object-centric framework is expected to increase the number of false negatives due to detection failures, but, in the same time, it significantly reduces the number of false positives (as the framework is focused on objects). Our results show that the object-centric pipeline attains signif-

icantly better results compared to its frame-level counterpart. Thus, the benefits of the object detector outweigh its limitations. Moreover, our final framework combines both object-centric and frame-level streams, alleviating the limitations of a pure object-centric method and improving the overall performance. Indeed, the frame-level pipeline can detect all anomaly types. The frame-level framework can localize anomalies by considering the magnitude of reconstruction errors in the output of the middle frame prediction head, just as other reconstruction-based approaches.

Generating object-centric temporal sequences. We take the bounding box of an object x in frame i and apply the same bounding box in preceding or subsequent frames to form an object-centric temporal sequence. If the object x is detected in another frame, say $i+1$, we will use the respective bounding box to generate another object-centric temporal sequence. Although we may end up with multiple slightly different sequences for the same object, this is better than applying an object tracker (which increases time and introduces errors).

Notes on the chosen proxy tasks. We underline that anomalies can be caused by both abnormal motion and abnormal appearance. Our multi-task framework can detect both anomaly types, since the first two proxy tasks (arrow of time, motion irregularity) focus on motion anomalies, while the last two tasks (middle box prediction, knowledge distillation) focus on appearance anomalies. Although our framework is simple, it is based on careful design thinking and significant effort in formulating the proxy tasks, in a single architecture, to be beneficial for anomaly detection. We believe that its simplicity coupled with its effectiveness in anomaly detection is interesting and compelling. Nevertheless, in future work, additional or alternative proxy tasks can be considered while seeking to further improve the results.