Hierarchical Semantic Contrast for Scene-aware Video Anomaly Detection
(Supplementary Materials)

Shengyang Sun  Xiaojin Gong*
College of Information Science & Electronic Engineering, Zhejiang University, Hangzhou, Zhejiang, China
{sunshy, gongxj}@zju.edu.cn

In this supplementary material, we provide more details about motion augmentation and the scene-dependent mixture datasets created upon ShanghaiTech. Besides, more implementation details on video parsing and hierarchical semantic contrast (HSC) are provided. More visualization results including frame-level anomaly scores and feature distribution of the ShanghaiTech test set are also illustrated. Finally, we also include the running time of our proposed method for the sake of reference.

1. Details on Motion Augmentation

1.1. Rotation-driven Skeleton Augmentation

We use HRNet [6] pre-trained on the MS-COCO keypoint detection dataset [4] to extract skeleton features. Fig. 1 illustrates the skeleton keypoints defined in MS-COCO. The parent and descendant nodes of each keypoint are listed in Table 1. In our motion augmentation, spatial transformation is conducted by randomly choosing a keypoint to rotate. The chosen keypoint is rotated around its parent node by a rotation angle $\alpha$ that is randomly sampled within a pre-defined range. The reasonable rotation range of each keypoint is different due to the human anatomical structure. Therefore, we manually define the range of each keypoint and list all in Table 1.

1.2. Selection of Normal/Abnormal Samples

The samples generated in motion augmentation are not guaranteed to be normal, therefore we apply our trained model to discriminate normal and abnormal samples as introduced in Section 3.5. (Training and Test). More specifically, we use the reconstruction error of the motion stream defined in Eq.(11) for discrimination. That is, given an augmented sample $s$ and its encoded motion feature $f_{s}^{mot}$, the reconstruction error is

$$S_{mot}(f_{s}^{mot}) = \|f_{s}^{mot} - \Theta_{mot}(\sum_{i=1}^{N} w_{i}M_{mot}(i))\|_{2}^{2}. \quad (1)$$

Figure 1. 2D human skeleton keypoints defined in MS-COCO.
Table 1. The pre-defined rotation range of each keypoint in our spatial transformation for motion augmentation.

<table>
<thead>
<tr>
<th>Keypoint</th>
<th>Parent</th>
<th>Descendant</th>
<th>Rotation Range (degree)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0, 1, 2, 3, 4</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>7, 9</td>
<td>[-10, 10]</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>8, 10</td>
<td>[-10, 10]</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
<td>9</td>
<td>[-90, 90]</td>
</tr>
<tr>
<td>8</td>
<td>6</td>
<td>10</td>
<td>[-90, 90]</td>
</tr>
<tr>
<td>9</td>
<td>7</td>
<td>-</td>
<td>[0, 90]</td>
</tr>
<tr>
<td>10</td>
<td>8</td>
<td>-</td>
<td>[0, 90]</td>
</tr>
<tr>
<td>11</td>
<td>5</td>
<td>13, 15</td>
<td>[-10, 10]</td>
</tr>
<tr>
<td>12</td>
<td>6</td>
<td>14, 16</td>
<td>[-10, 10]</td>
</tr>
<tr>
<td>13</td>
<td>11</td>
<td>15</td>
<td>[-90, 90]</td>
</tr>
<tr>
<td>14</td>
<td>12</td>
<td>16</td>
<td>[-90, 90]</td>
</tr>
<tr>
<td>15</td>
<td>13</td>
<td>-</td>
<td>[-90, 0]</td>
</tr>
<tr>
<td>16</td>
<td>14</td>
<td>-</td>
<td>[-90, 0]</td>
</tr>
</tbody>
</table>

The notations are defined the same as in Eq.(11) in the paper.

Then, we set a normality threshold $T_{norm}$ and an anomaly threshold $T_{abn}$ for the selection of normal and abnormal samples. When $S_{mot}(f_{s}^{mot}) < T_{norm}$, the sample $s$ is determined as normal. When $S_{mot}(f_{s}^{mot}) > T_{abn}$, the...
sample is viewed as abnormal. Any augmented sample with a reconstruction error between $T_{\text{norm}}$ and $T_{\text{abn}}$ is discarded. In our experiments, we set $T_{\text{norm}} = 0.2$ and $T_{\text{abn}} = 0.8$ for all datasets. For the sake of reference, Fig. 2 illustrates the histograms of reconstruction errors obtained on the original Avenue and ShanghaiTech training sets, which contain normal samples only. From this figure, we see that $T_{\text{norm}} = 0.2$ is a quite strict threshold for normality and $T_{\text{abn}} = 0.8$ is reasonable for anomalies.

1.3. Normal/Abnormal Binary Classifier

As mentioned in Section 3.5. (Training and Test), we leverage both normal and abnormal augmented samples to additionally train a binary classifier, which can boost the performance further. More specifically, a normal augmented sample is assigned with a pseudo label $y = 0$ and an abnormal sample is assigned with a pseudo label $y = 1$. Then, the binary classifier is trained using the cross-entropy loss

$$L_{\text{aug}}^{\text{mot}} = -y \log(\hat{y}) - (1 - y) \log(1 - \hat{y}), \quad (2)$$

where $\hat{y}$ is the predicted classification probability. The classifier is a three-layer MLP, where the number of hidden units is 512, 32, and 2 respectively. The ReLU function is used in the first layer and dropout with a probability of 0.6 is deployed between each layer. The Softmax function is utilized after the last layer.

2. Details on ShanghaiTech Mixture Datasets

To investigate the performance of our method on scene-dependent anomaly detection, we additionally create three mixture datasets based on ShanghaiTech. The Mixture [01, 02] set consists of videos selected from the scene ‘01’ and ‘02’. We partly take the test videos of ‘01’ containing the cyclist events into the training set and delete them from the test set. It implies that cyclist becomes normal in the scene ‘01’, but it is still abnormal in the scene ‘02’. The Mixture [04, 08] set includes videos of scene ‘04’ and ‘08’. We take some videos including cyclist events from the test set of scene ‘04’ into the training set, and take some videos including running and skater events from the test set of scene ‘08’ into the training set. Mixture [10, 12] consists of videos selected from scene ‘10’ and ‘12’. We take some videos including cyclist or tricyclist events from the test set of scene ‘12’ into the training set. More details of the mixture datasets are shown in Table 2.

3. More Implementation Details

3.1. Details on Video Parsing

we adopt the pre-trained YOLOv3 [5] and FairMOT [8] to detect and track objects. The threshold of detection is set to 0.8 for ShanghaiTech and Avenue, but set to 0.5 for UCSD Ped2 due to its low resolution. The threshold of tracking is set to 0.7, 0.5, and 0.3 for ShanghaiTech, Avenue, and UCSD Ped2, respectively. Besides, we only track people, bicycle, tricycle, cars and such vehicles. Meanwhile, to achieve meaningful motion features, we only feed an object tracklet containing more than 10 tracked frames into PoseConv3D [3]. For scene feature extraction, we deploy the max-pooling with a size of 4 after the classifier of DeepLabV3+ [1], which leads to the dimension of the scene feature $D_B$ to be 1590, 880, and 330 for ShanghaiTech, Avenue and UCSD Ped2, respectively.

3.2. Details on Hierarchical Semantic Contrast

In our hierarchical semantic contrast method, the scene-appearance/motion feature encoders are implemented by two-layer MLPs, where the number of hidden units is set to [2048, 1280] and [1792, 1280], respectively, for appearance and motion encoders. The appearance/motion feature decoders are also implemented by two-layer MLPs, where the number of hidden units is set to [1152, 1024] and [1024, 512] for appearance and motion decoders, respectively. The ReLU function is employed in the first layer and batch normalization is utilized between each layer. The linear classifiers consist of one linear layer, and the number of outputs is 23, 2, and 1 for ShanghaiTech, Avenue, and UCSD Ped2 respectively, which are the number of scene clusters obtained by DBSCAN. Besides, all models are trained for 200 epochs in our experiments.

4. More Results

4.1. Qualitative Analysis

ShanghaiTech. The proposed method is able to detect multiple abnormal events in one video, such as multiple bicycles in the test video 01_0134 shown in Fig. 3. Besides, our method is not sensitive to the change of object views, for instance, the varying view of bicycles in the test video 01_0134 and 06_0153. We also present a failed case as the test video 08_0159 shown in Fig. 3. Our method may fail
to detect the anomalies if objects are occluded and not detected, e.g., an occluded runner in this case. In addition, an abnormal object may be not detected if it moves so fast that meaningful skeleton keypoints cannot be extracted due to motion blur.

**Avenue.** The proposed method outperforms all previous SOTA methods on the Avenue dataset. As shown in Fig. 4, our method detects diverse anomalies in the test video 06, including a man moving towards cameras and a man throwing some clutter. Our method achieves a high frame-level AUC of 98.32% for test video 12, where *leaping* and *throwing* actions are correctly detected as anomalies. Some failed cases occurred due to occlusion, motion blur, or the distance far away from cameras, as shown in the test video 01 in Fig. 4.

**UCSD Ped2.** Despite only the scene stream of our model being utilized on the UCSD Ped2 dataset, it reaches a perfect frame-level AUC in the test video 02 and 07 as shown in Fig. 5. But it fails to detect a couple of anomalies, e.g., an occluded bicycle in the test video 09, partially due to no consideration of motion information.

**ShanghaiTech Mixture.** Fig. 6 shows the performance of the proposed method on normal videos on the created ShanghaiTech Mixture datasets. In these scene-dependent sets, cyclists in the test videos 01_0129 and 12_0174, and a man running in the test video 08_0078 become normal. From Fig. 6 we observe that our HSC model without motion augmentation (MA) can successfully recognize most of these events as normality with a low anomaly score in the test videos 01_0129 and 08_0078. In the test video 12_0174, the HSC model without MA detects the cyclist as an anomaly because the training set contains few cyclist samples. With motion augmentation, our model generates lower anomaly scores for these normal events, validating the effectiveness of our augmentation strategy.

**The distribution of encoded latent features on ShanghaiTech test set.** Fig. 7 visualizes the distribution of scene-dependent features from the ShanghaiTech test set, obtained

---

**Table 2. Details of ShanghaiTech Mixture datasets.**

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Training Videos</th>
<th>Test Videos</th>
<th>Scene</th>
<th>Normality</th>
<th>Anomaly</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mixture [01, 02]</td>
<td>01_0016, 01_0029, 01_0051, 01_0052, 01_0054, 01_0063, 01_0073, 01_0076, 01_0132, 01_0133, 01_0138, 01_0139, 01_0140, 01_0162, 01_0163, 01_0177, 01_020, 01_030, 02_001, 02_002, 02_003, 02_004, 02_005, 02_006, 02_007, 02_008, 02_009, 02_010, 02_011, 02_012, 02_013, 02_014, 02_01</td>
<td>01_0014, 01_0015, 01_0030, 01_0053, 01_0064, 01_0129, 01_0130, 01_0131, 01_0134, 01_0135, 01_0136, 01_0141, 02_0128, 02_0161, 02_0164</td>
<td>01</td>
<td>pedestrian, cyclist</td>
<td>skater, tricyclist, running, vehicle</td>
</tr>
<tr>
<td>Mixture [04, 08]</td>
<td>04_001, 04_002, 04_003, 04_004, 04_005, 04_006, 04_007, 04_009, 04_010, 04_011, 04_012, 04_013, 04_014, 04_015, 04_016, 04_017, 04_018, 04_019, 04_020, 04_0003, 08_001, 08_005, 08_006, 08_007, 08_008, 08_009, 08_010, 08_011, 08_012, 08_013, 08_014, 08_015, 08_016, 08_017, 08_018, 08_019, 08_022, 08_029, 08_030, 08_031, 08_038, 08_041, 08_042, 08_047, 08_0044, 08_0079, 08_0080, 08_0156, 08_0157, 08_0158, 08_0159, 08_01</td>
<td>04_0001, 04_0004, 04_0101, 04_0111, 04_0102, 04_013, 04_046, 04_050, 08_0058, 08_0077, 08_0078, 08_0179</td>
<td>04</td>
<td>pedestrian, cyclist</td>
<td>running, chasing, jumping, tumbling, walking circularly, throwing, skater, stroller</td>
</tr>
<tr>
<td>Mixture [10, 12]</td>
<td>10_001, 10_002, 10_003, 10_004, 10_005, 10_006, 10_007, 10_008, 10_009, 10_010, 10_011, 12_001, 12_002, 12_003, 12_004, 12_005, 12_006, 12_007, 12_008, 12_009, 12_010, 12_011, 12_012, 12_013, 12_014, 12_015, 12_012, 12_013, 12_014, 12_015, 12_016, 12_0143, 12_0148, 12_0151, 12_015</td>
<td>10_0037, 10_0038, 10_0042, 10_0074, 10_0075, 12_0149, 12_0152, 12_0173, 12_0174, 12_017</td>
<td>10</td>
<td>pedestrian</td>
<td>cyclist, motorcyclist, running</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>12</td>
<td>pedestrian, cyclist</td>
<td>running, vehicle</td>
</tr>
</tbody>
</table>
Figure 3. Frame-level anomaly scores of test videos on ShanghaiTech. The first two rows present successful cases, and the last row presents a failed case. The green (red) bounding boxes represent the detected (undetected) objects, and the red regions indicate ground truth.

Figure 4. Frame-level anomaly scores of test videos on Avenue. The first two rows present successful cases, and the last row presents a failed case. The green (red) bounding boxes represent the detected (undetected) objects, and the red regions indicate ground truth.
Figure 5. Frame-level anomaly scores of test videos on UCSD Ped2. The first two rows present successful cases, and the last row presents a failed case. The green (red) bounding boxes represent the detected (undetected) objects, and the red regions indicate ground truth.

Figure 6. Frame-level anomaly scores of normal test videos on ShanghaiTech Mixture. The blue lines are anomaly scores predicted by HSC w/o MA, and the orange lines represent the scores predicted by HSC w/ MA ($\text{MA}^{-\text{+}}$ denotes generating normal and abnormal samples).
by the models without or with HSC. Like those features on
the training set as shown in Fig. 4 in the paper, we ob-
serve that the features on the test set also distribute more
compactly within identical classes and more separately be-
tween different classes, demonstrating the effectiveness of
our proposed hierarchical semantic contrast.

4.2. Running Time

All our experiments are run on one NVIDIA A6000
FPS with an average of 8 objects/frame. For one clip from
ShanghaiTech (16 frames and more than 100 objects), the
scene feature extractor takes 235.9 milliseconds (ms), the
ViT [2] takes 460.3 ms, the HRNet [6] and PoseConv3D [3]
totally take 537.5 ms, and the scene-appearance and scene-
motion autoencoders totally take 3.4 ms. The entire frame-
work runs about 21 FPS.

References

[1] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian
Schroff, and Hartwig Adam. Encoder-decoder with atrous
separable convolution for semantic image segmentation. In
ECCV, pages 801–818, 2018. 2
[2] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov,
Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner,
Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain
Gelly, et al. An image is worth 16x16 words: Transformers for
2020. 6
[3] Haodong Duan, Yue Zhao, Kai Chen, Dahua Lin, and Bo Dai.
Revisiting skeleton-based action recognition. In CVPR, pages
2969–2978, 2022. 2, 6
Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence
Zitnick. Microsoft coco: Common objects in context. In
resolution representation learning for human pose estimation.
In CVPR, pages 5693–5703, 2019. 1, 6
data using t-sne. Journal of machine learning research, 9(11),
2008. 6
[8] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng,
and Wenyu Liu. Fairmot: On the fairness of detection and re-
identification in multiple object tracking. International Jour-