ROAD: UNIFIED REFERRING OPERATION ANOMALY DETECTION VIA CROSS-SCENE DUAL MEMORY

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing operation video anomaly detectors are bound to specific scenes and require retraining when the viewpoint, objects, or procedure changes. To tackle this limitation, we formulate the task of referring operation anomaly detection, where a detector scores a test video against a reference video of the valid operation, adapting to new scenarios solely by swapping the reference video. We release a dataset of a large number of continuous one-take recordings from four scenarios, with annotated reference–test pairs and frame-level labels covering four process anomalies: wrong order, wrong placement, omitted step, and wrong component. We further propose a cross-scene dual-memory framework that separates target-specific progression in a reference-anchored temporal memory from cross-scene normality in a learnable prototype memory, fused by gated residual reconstruction. It reaches 91.63% mean AUROC across the three leave-one-scenario-out settings and 88.80% on a real-world packaging scenario, exceeding the previous strongest baselines by over 19.73 points. Code will be public after acceptance.

1 INTRODUCTION

Operation video anomaly detection takes a test video as input and assigns an anomaly score to each test frame. This task is important for automated manufacturing, assembly monitoring, and visual inspection, where anomalous operations are rare and costly to annotate. Most existing methods are developed for a specific scene and must be retrained when the viewpoint, manipulated objects, or operation procedure changes. As illustrated in Figure 1(a), a model trained for one scene may work well in that scene but fail when directly applied to another scene.

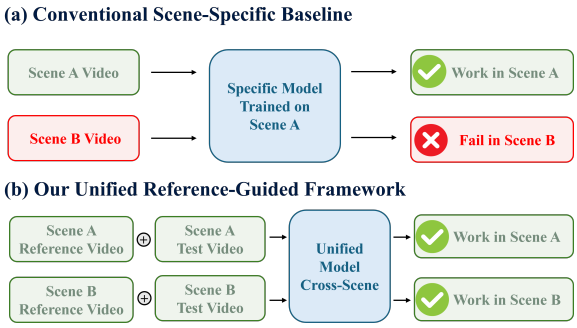

Figure 1: Conventional scene-specific adaptation versus our reference-guided formulation. (a) Conventional models require retraining when transferred to a new scene. (b) Our model keeps its weights fixed and adapts by replacing only the reference video.

A primary reason for this limitation is that existing operation video anomaly detection datasets generally cover only a small number of fixed scenarios and primarily adopt within-scenario training and test splits. Consequently, they mainly evaluate detection in familiar scenarios and cannot adequately measure generalization across different viewpoints, layouts, and manipulated objects. Moreover, their annotated anomalies are often identifiable from local appearance, object placement, or short-term motion, with less emphasis on dependencies across multiple operation steps. They therefore provide limited evaluation of process anomalies whose identification depends on the reference procedure, such as incorrect ordering and omitted steps. Overall, limited scenario diversity and weak coverage of multi-step temporal dependencies prevent existing datasets from fully evaluating anomaly detection in practical operation scenarios.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Beyond data scarcity, existing methods for operation anomaly detection typically extract spatiotemporal features from video clips, retrieve the memory item most relevant to the current clip from a single memory bank based on feature similarity, and use the retrieved representation to reconstruct the corresponding target frame (Gong et al., 2019; Park et al., 2020; Wang et al., 2023; Liu et al., 2024). Anomalies are then detected based on the discrepancy between the input frame and its reconstruction. By storing normal patterns observed in the training scenarios, the memory bank provides a stable basis for within-scenario anomaly detection. However, since the memory bank is constructed only from the source scenarios, its stored representations are closely coupled with source-specific appearance, layout, and operation patterns and may therefore fail to provide an appropriate normality basis for an unseen target scenario.

To address these limitations, we first formulate a new task of referring operation anomaly detection and construct a multi-scenario industrial operation-video dataset for its evaluation. Given a reference video demonstrating a valid operation and a test video recorded in the same target scenario, the task assigns an anomaly score to each test frame according to whether the observed operation is consistent with the reference operation. As shown in Figure 1(b), the same detector can be applied to different target scenarios without retraining; deployment in a new scenario requires only replacing the reference video. The dataset contains four operation scenarios and organizes the recorded operations into annotated reference–test pairs with frame-level labels. Each video continuously records a complete multi-step operation in a single take, and the dataset includes four anomaly patterns: wrong order, wrong placement, omitted step, and wrong component. The dataset therefore enables controlled evaluation of both cross-scenario generalization and temporal dependencies among multiple operation steps.

For this task, we further develop a dual-memory reconstruction pipeline that separates target-specific information from common patterns shared across scenarios. A non-parametric target memory is constructed from the reference video to preserve target-specific appearance, layout, and operation progression. In parallel, a learnable residual prototype memory is trained across scenes to represent shared patterns and compensate for the limited variation contained in a single reference video. Given a test clip, the model retrieves representations from both memories and fuses them through a residual reconstruction process. This design enables a unified detector to adapt its reconstruction to a new target scenario by replacing only the reference video.

Our contributions are summarized as follows:

- We formulate a new task of referring operation anomaly detection, which scores a test video against a reference operation, and construct a multi-scenario operation-video dataset with annotated reference–test pairs, frame-level labels, and four process anomaly patterns
- We further design a unified detector whose frozen weights adapt to a new scenario solely by replacing the reference video, using a reference-anchored temporal memory and a shared residual prototype memory to reconstruct reference-consistent frames.
- Extensive experiments show our method attains 91.63% mean AUROC across the three leave-one-scenario-out settings and 88.80% on a real-world packaging scenario, demonstrating strong generalization without retraining.

2 Dataset Construction

2.1 Operation Scenarios

The proposed dataset contains four operation scenarios: three controlled tabletop scenarios—block assembly (A), object placement (B), and screw tightening (C)—and one real industrial product-packaging scenario involving an animal-grooming device and its accessories (D). These scenarios differ in their manipulated objects, workspace layouts, operation procedures, and action primitives, while all videos continuously record complete operations from a fixed viewpoint. Figure 2 summarizes the dataset construction pipeline, including scenario design, one-take video acquisition, reference-test pairing and anomaly design, and frame-level labeling.

Specifically, each scenario records a complete multi-step operation from a fixed viewpoint. **(A) Block assembly:** the operator places seven colored blocks at designated positions; executions vary in placement order yet may reach identical final arrangements, testing order-sensitive detection. **(B)**

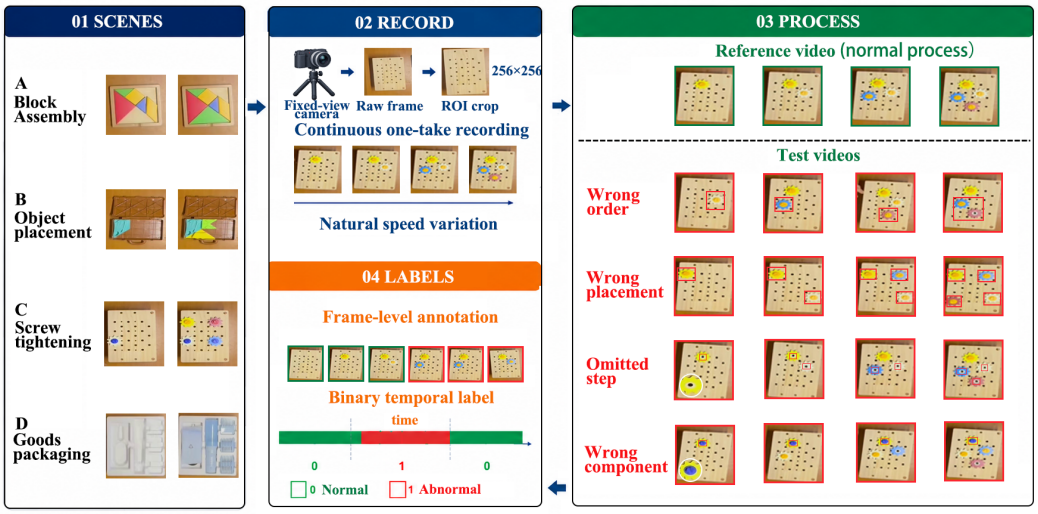

Figure 2: Overview of dataset construction. Videos are recorded as complete operations, then organized into reference–test pairs with frame-level labels. Four anomaly patterns are shown: wrong order, wrong placement, omitted step, and wrong component.

Object placement: a similar pick-and-place procedure, but with geometrically distinct objects on a compartmentalized board, differing from A in object geometry, appearance, and layout. **(C) Screw tightening:** at each of six positions, the operator inserts a screw and fastens a nut, introducing rotational tightening and coupled sub-steps. **(D) Goods packaging:** a real industrial packaging line for an animal-grooming device, involving real products, materials, and workspace. These scenarios enable controlled transfer evaluation: AB→C tests generalization to unseen action primitives, AC→B and BC→A test transfer across object families and layouts, and ABC→D tests transfer from designed tasks to a real process. In all protocols, parameters remain frozen and only the target reference video is replaced.

2.2 Data Collection and Label Construction

Video Record and Pairing. Every video captures one independent, complete operation in a single take from a fixed viewpoint and remains unlabeled at acquisition. Annotation then selects a valid execution as the reference video and pairs it with an independently recorded test video; frames are labeled by comparing semantic operation progression, ignoring differences in execution speed, as shown in the left and middle part of Figure 2.

Anomaly pattern design. Right part of Figure 2 illustrates four anomaly patterns, each defined relative to the state progression demonstrated by the reference video. In the illustrated reference operation, the yellow, off-white, blue, and red components are installed in this order. **(1) Wrong order** changes this temporal sequence; for example, the test operation installs the off-white component before the yellow component, although the yellow component should be installed first. **(2) Wrong placement** uses the correct component but installs it at a position different from the one specified by the reference operation; in the example, the yellow screw is inserted into an incorrect position. **(3) Omitted step** skips a required action; in the example, the yellow screw has been placed, but its corresponding nut is not installed. **(4) Wrong component** preserves the expected position but uses a component with an incorrect identity: a blue nut is installed on the yellow screw instead of the required yellow nut. The outer red borders denote frames labeled as inconsistent, while the smaller red boxes highlight the evidence of each inconsistency rather than pixel-level anomaly annotations.

Frame-level state labels. As shown in the labelling panel of Figure 2, each frame of a test video receives a binary label according to whether its observable operation state is consistent with the reference operation. A frame is assigned label 0 when the current action and object arrangement agree with the reference, including normal pauses and execution-speed differences. The label be-

Table 1: Dataset statistics and characteristic comparison with IPAD (Liu et al., 2024).

(a) Scenario-level totals.

Scenario	Videos	Pairs	Frames
A: Block assembly	148	100	73,496
B: Object placement	159	100	70,434
C: Screw tightening	152	100	128,005
D: Goods packaging	110	100	63,915
Total	**569**	**400**	**335,850**

(b) Dataset characteristics.

Metric	IPAD	Ours
Real-frame ratio (%)	25.0	**100.0**
Average frames per video	559.9	**590.25**
Interval visual difference	0.062	**0.139**
Start-to-end visual difference	0.065	**0.202**
Operation patterns	1.08	**15**

comes 1 at the first frame where an inconsistent action or state is observable and remains 1 while the resulting operation state differs from the reference, even after the action causing the deviation has ended. Once a later action brings the observable state back into agreement with the reference, subsequent frames are assigned label 0. For an omitted step, labelling begins at the first frame where the required state is absent and continues while this state inconsistency remains observable.

2.3 Dataset statistics Analysis

The scenario-level statistics of the proposed dataset are shown in Table 1a. The proposed dataset contains 569 videos and 335,850 frames across four operation scenarios. Each scenario provides ten reference–query pairs, resulting in 400 pairs overall. The number of videos and frames varies across scenarios because each operation contains different recording durations and temporal patterns.

IPAD (Liu et al., 2024) is an industrial process video anomaly detection benchmark comprising real and synthetic recordings, where anomalies are detected from the test video alone without a paired reference video; in contrast, our dataset provides reference–query video pairs and is designed to evaluate reference-conditioned cross-scene anomaly detection.The operation characteristics of the proposed dataset are shown in Table 1b: **(1) Real frame ratio** measures the proportion of frames from physically recorded videos relative to the total number of frames from real and synthetic videos. Our dataset consists entirely of physically recorded videos, whereas IPAD contains both real and synthetic recordings. **(2) Frame number per video** is the average number of frames in a video. The larger value of our dataset means that each recording preserves more frame-level observations of a complete operation. **(3) Interval visual difference** measures the average visual-feature difference among frames sampled at uniform temporal intervals throughout an operation. Specifically, we calculate the cosine distance between every pair of sampled-frame features and average these distances. The larger value of our dataset indicates that its sampled intermediate states are less visually redundant. **(4) Start-to-end visual difference** compares the average visual features of the first and last 10% of the sampled frames. Its larger value for our dataset indicates a greater overall configuration change from the initial to the completed state. **(5) Operation pattern number** counts the distinct valid operation patterns represented in each scenario, and the reported value is averaged across the four scenarios. Our dataset contains multiple valid operation patterns, whereas IPAD is represented by a single repeated process configuration. This explicit difference in operation patterns provides a direct basis for evaluating whether a detector can determine the consistency of a test operation with a specified reference operation. Overall, the comparison shows that our dataset contains longer continuous recordings, richer interval-sampled state changes, larger start-to-end configuration changes, and more diverse valid operation patterns.

3 Methodology

The proposed referring operation anomaly detection dataset is divided into a training set and a test set. The training set contains a large number of independent operation videos collected under different operating scenarios, and these videos are used individually for model training. The test set consists of reference–query video pairs, where the two videos in each pair belong to the same operating scenario, while the operations performed in these two videos may be the same or different.

As illustrated in Figure 3, our model is trained on the training set with a reconstruction-based objective to learn the spatiotemporal encoder E, reconstruction decoder D, and our proposed Cross-Scene

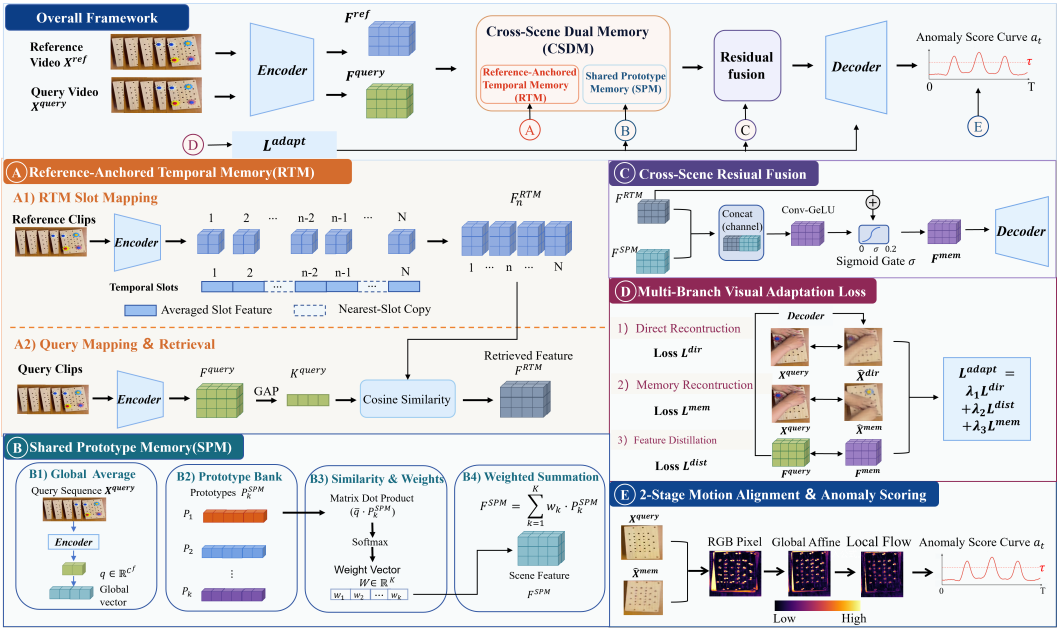

Figure 3: Overview of the proposed framework. The Reference-Anchored Temporal Memory (RTM) and Shared Prototype Memory (SPM) are formally defined in Section 3.2. Reconstruction pre-training first learns the spatiotemporal reconstruction backbone. Cross-Scene Dual Memory Reconstruction then uses a reference video to construct the episode-specific memory and combines it with source-shared prototypes. Finally, the reconstructed query frame is aligned with the corresponding query frame and converted into a frame-level anomaly score.

Dual Memory module. During inference, the paired reference video and query video are encoded by E to obtain their spatiotemporal features. The Reference-anchored Temporal Memory (RTM) is then constructed by arranging the reference clip features into chronologically ordered slots, and each query clip retrieves its best-matched reference feature via cosine-similarity search within a local temporal window. In parallel, the Shared Prototype Memory (SPM), a bank of learnable prototypes shared across scenarios, is queried once for each query video: the global feature of the fixed prefix of query clips retrieves a weighted combination of prototypes. The retrieved RTM and SPM features are fused by a gated residual adapter and decoded by D to reconstruct the query clip. Finally, each reconstructed query frame is first aligned with its corresponding query frame through image registration. The resulting image-space residual is then used to obtain the frame-level anomaly score.

3.1 Reconstruction Pre-training

The first step of the propose method is to train a spatiotemporal reconstruction backbone on the training set. Let $\mathbf{V}^{\text{train}}$ denote a training video. A clip $\mathbf{X}$ is sampled from this video, with the size of $\mathbb{R}^{3\times T^{\text{clip}}\times H\times W}$, where 3 denotes the RGB channels, T^{clip} denotes the number of frames in the clip, and H and W denote the frame height and width, respectively. The encoder E maps the clip to a spatiotemporal feature $\mathbf{F} = E(\mathbf{X})$. The feature is subsequently passed to the reconstruction decoder D, which consists of transposed-convolution blocks and restores its temporal and spatial dimensions. The resulting reconstruction is defined as $\widehat{\mathbf{X}} = D(\mathbf{F})$, with the same size as the input clip $\mathbf{X}$. The encoder E and decoder D are pre-trained through $\mathcal{L}^{\text{pre}} = \text{MSE}\left(\mathbf{X}, \widehat{\mathbf{X}}\right)$.

3.2 Cross-Scene Dual Memory

Training Sequences Construction. To train the proposed cross-scene dual memory module, two video sequences are first sampled from each video in the training set for each training episode: a reference sequence and a query sequence. During training, the reference sequence mimics the test-

time reference video, providing the reference operation information needed for the retrieval process. Meanwhile, the query sequence mimics the test-time query video and is used to conduct retrieval using the proposed cross-scene dual memory module. In other words, the reference and query sequences are sampled from the same training video.

Specifically, several clips are first sampled from the training video $\mathbf{V}^{\text{train}}$ with a fixed interval, each containing T^{clip} consecutive frames, to form the reference sequence. We then sample query clips of the same length from the remaining portions of $\mathbf{V}^{\text{train}}$ to form the query sequence. Arranging the two groups in temporal order forms the reference sequence and the query sequence, respectively, as illustrated in Figure 4. The resulting sequences are then used for the training of train the proposed cross-scene dual memory module.

Reference-anchored Temporal Memory. The reference sequence information is first converted into the proposed Reference-anchored Temporal Memory (RTM). The frozen encoder E extracts the spatiotemporal feature from each reference clip within the reference sequence, and each clip feature is assigned to one of N ordered slots according to its temporal position in the reference sequence. Let $\mathbf{F}_n^{\text{RTM}}$ denote the feature stored in slot n. If multiple reference clips are assigned to the same slot, their features are averaged. If a slot is empty, it copies the feature from the nearest preceding populated slot. Note that, RTM is rebuilt for each reference sequence and contains no learnable parameters. Once RTM has been constructed, the frozen encoder E encodes each query clip $\mathbf{X}_i^{query}$ in the query sequence to generate its feature $\mathbf{F}_i^{\text{query}}$. The query key and the retrieval key of RTM slot n are obtained as $\mathbf{k}_i^{\text{query}} = \text{GAP}(\mathbf{F}_i^{\text{query}})$ and $\mathbf{k}_n^{\text{RTM}} = \text{GAP}(\mathbf{F}_n^{\text{RTM}})$, respectively. Here, $\text{GAP}(\cdot)$ averages a spatiotemporal feature over its temporal, height, and width dimensions while retaining the feature-channel dimension. The query key is compared with RTM retrieval key and selects the slot with the highest cosine similarity:

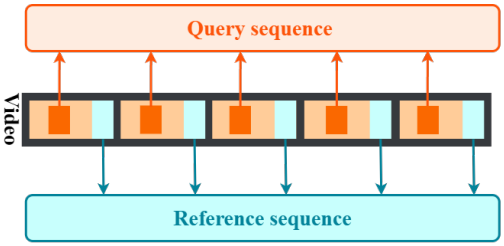

Figure 4: Training-time construction of a reference–query sequence pair from one training video. Reference clips are sampled first, and query clips of the same length are sampled from the remaining portions of the same video; each group is then arranged in temporal order.

$$n^* = \underset{|n-U(i)|\le Y}{\arg\max} \frac{(\mathbf{k}_i^{\text{query}})^\top \mathbf{k}_n^{\text{RTM}}}{\|\mathbf{k}_i^{\text{query}}\|_2 \, \|\mathbf{k}_n^{\text{RTM}}\|_2}. \tag{1}$$

Here, $U(i)$ indicates the slot position in RTM corresponding to $\mathbf{X}_i^{query}$ and Y denotes the retrieval radius measured in slots, and n^* denotes the selected slot index. The selected feature $\tilde{\mathbf{F}}^{\text{RTM}} = \mathbf{F}_{n^*}^{\text{RTM}}$ is passed to the subsequent residual fusion module. Note that, the same retrieval procedure is applied independently to each query clip in the query sequence.

Shared Prototype Memory. SPM is a bank of K^{SPM} learnable prototypes shared across different scenarios, denoted as $\mathbf{F}^{SPM}$, where the common scene information of different scenario is stored. Unlike RTM, which stores temporally ordered clip-level features from the current reference sequence, SPM stores source-shared prototypes that represent stable scene-level normality across different scenarios. The averaged feature of the query sequence $\bar{\mathbf{X}}^{query}$ is used to calculate the similarity with all prototypes in SPM to generate the result feature $\tilde{\mathbf{F}}^{\text{SPM}}$, as follows:

$$\tilde{\mathbf{F}}^{\mathbf{SPM}} = \text{Softmax}\left(\frac{\text{Norm}_2(\bar{\mathbf{X}}^{query})\left(\text{Norm}_2(\mathbf{F}^{SPM})\right)^\top}{\sigma}\right) \cdot \mathbf{F}^{SPM}. \tag{2}$$

Here, the operator $\text{Norm}_2(\cdot)$ performs ℓ_2 normalization along the feature dimension. Note that, the same $\tilde{\mathbf{F}}^{\text{SPM}}$ is reused for all query clips from the same query sequence, whereas RTM independently retrieves $\tilde{\mathbf{F}}^{\text{RTM}}$ for each current query clip.

Residual fusion and reconstruction. For the current query clip, the feature selected from RTM and SPM are combined using a residual adapter. The selected RTM feature $\tilde{\mathbf{F}}^{\text{RTM}}$ serves as the base

representation while the SPM feature $\tilde{\mathbf{F}}^{\mathrm{SPM}}$ contributes a residual information. RTM is used as the base because it provides temporally localized information matched to the current query clip, whereas SPM is introduced as a residual because it provides complementary scene-level information. The residual adapter, denoted by Φ, consists of two convolutional mappings with a GELU nonlinearity. A bounded learnable gate controls the magnitude of the residual, calculated as follows:

$$\mathbf{F}^{\mathrm{mem}} = \tilde{\mathbf{F}}^{\mathrm{RTM}} + \beta_{\max} \cdot b_\beta \cdot \Phi\left(\mathrm{Concat}\left(\tilde{\mathbf{F}}^{\mathrm{RTM}}, \tilde{\mathbf{F}}^{\mathrm{SPM}}\right)\right). \tag{3}$$

Here, $\mathrm{Concat}(\cdot,\cdot)$ concatenates the two features along the feature-channel dimension. The constant $\beta_{\max}$ specifies the upper bound of the residual scale. The resulting $\mathbf{F}^{\mathrm{mem}}$ preserves the selected RTM feature as its base feature while incorporating a controlled update conditioned on SPM. The decoder D maps $\mathbf{F}^{\mathrm{mem}}$ to the memory-conditioned reconstruction $\widehat{\mathbf{X}}^{\mathrm{mem}}$ of the current query clip.

Dual Memory Optimization Objective. The Cross-Scene Dual Memory module is optimized using the training episodes described above. Let $\mathbf{X}_i^{\mathrm{query}}$ denote the current query clip. The decoder D reconstructs it from the fused memory feature $\mathbf{F}^{\mathrm{mem}}$. The optimization objective for the current query clip is calculated as follows:

$$\begin{aligned}\mathcal{L}^{\mathrm{adapt}} = {} & \lambda^{\mathrm{dir}}\,\mathrm{MSE}\left(D\left(\mathbf{F}_i^{\mathrm{query}}\right), \mathbf{X}_i^{\mathrm{query}}\right) \\ & + \lambda^{\mathrm{mem}}\,\mathrm{MSE}\left(\widehat{\mathbf{X}}^{\mathrm{mem}}, \mathbf{X}_i^{\mathrm{query}}\right) \\ & + \lambda^{\mathrm{dist}}\,\mathrm{MSE}\left(\mathbf{F}^{\mathrm{mem}}, \mathrm{stopgrad}\left(\mathbf{F}_i^{\mathrm{query}}\right)\right).\end{aligned} \tag{4}$$

The first term preserves the direct reconstruction capability learned during pre-training. The second term optimizes the RTM–SPM fusion branch by comparing the memory-conditioned reconstruction with the same query clip. The third term constrains the fused feature to remain close to the encoded query feature, while $\mathrm{stopgrad}(\cdot)$ treats the latter as a fixed target in this term. The coefficients λ^{dir}, λ^{mem}, and λ^{dist} control the contributions of the three terms. Such optimization objectives are averaged over the query clips in each training iteration.

3.3 Aligned Anomaly Scoring

We use the final frame of each reconstructed clip to calculate the anomaly score. The reconstructed frame is aligned with the query frame through image registration. This alignment reduces residual responses caused by small camera motion, object displacement, or imperfect spatial reconstruction. Specifically, for the t-th scored query frame, let $\mathbf{x}_t$ denotes the original frame and let $\widehat{\mathbf{x}}_t$ denote its reconstruction. Registration is performed in two stages. First, a constrained global affine transformation is selected to align the overall spatial structure of the reconstruction with the query frame. A bounded local displacement is applied to compensate for the remaining small spatial misalignment:

$$\begin{aligned}\mathbf{G}_t^* &= \arg\max_{\mathbf{G}\in\mathcal{G}} \mathrm{ECC}\left(\mathbf{x}_t, \mathcal{W}_{\mathrm{aff}}\left(\widehat{\mathbf{x}}_t; \mathbf{G}\right)\right), \\ \widetilde{\mathbf{x}}_t &= \mathcal{W}_{\mathrm{local}}\left(\mathcal{W}_{\mathrm{aff}}\left(\widehat{\mathbf{x}}_t; \mathbf{G}_t^*\right); \mathbf{u}_t\right).\end{aligned} \tag{5}$$

Here, $\mathcal{G}$ denotes the allowed range of global affine transformations, and $\mathbf{G}_t^*$ denotes the transformation selected for frame t. The Enhanced Correlation Coefficient (ECC) measures the alignment between the query frame and the reconstruction Evangelidis & Psarakis (2008). The operator $\mathcal{W}_{\mathrm{aff}}(\cdot; \mathbf{G}_t^*)$ applies the selected affine transformation to the entire reconstructed frame. The local displacement field $\mathbf{u}_t$ is subsequently estimated between the query frame and reconstruction using dense Farneback optical flow Farneb"ack (2003). Its displacement magnitude is bounded so that $\mathcal{W}_{\mathrm{local}}(\cdot; \mathbf{u}_t)$ corrects only small residual misalignment. The resulting $\widetilde{\mathbf{x}}_t$ is the aligned reconstruction. The frame-level anomaly score a_t is obtained by averaging the pixel-wise residual:

$$a_t = \frac{1}{|\Omega_t|} \sum_{\boldsymbol{\xi}\in\Omega_t} \left\|\mathbf{x}_t(\boldsymbol{\xi}) - \widetilde{\mathbf{x}}_t(\boldsymbol{\xi})\right\|_1. \tag{6}$$

Here, Ω_t denotes the set of pixel locations that remain valid after global and local warping, and $|\Omega_t|$ denotes the number of these locations. Invalid boundary pixels are excluded from the average.

Table 2: Main comparison result under the reference-query cross-scene protocol. Performance is evaluated by frame-level AUROC (%). The method with the highest performance is marked in **bold**.

Method	A+B→C	A+C→B	B+C→A	Mean (LOSO)	A+B+C→D
FSAD Lu et al. (2020)	32.28	70.51	63.45	55.41	40.46
AD-FSAVAD Zahid et al. (2025)	38.09	41.25	62.77	47.37	36.57
TEAM-RefNN Lee et al. (2025)	70.20	71.64	62.36	68.07	68.00
OTAM-Ref Cao et al. (2020)	69.97	77.48	68.25	71.90	64.59
Ours	**88.98**	**93.34**	**92.59**	**91.63**	**88.80**

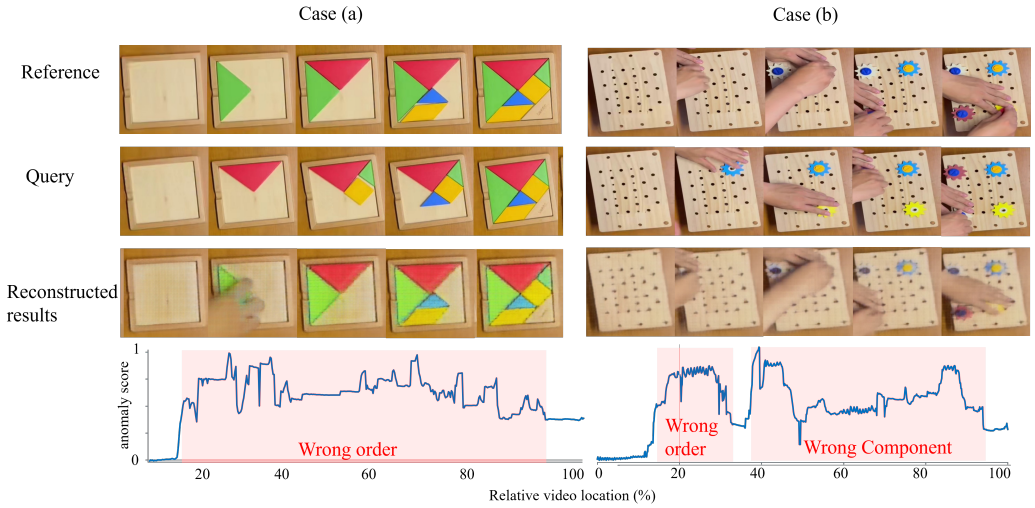

Figure 5: Qualitative results on case (a) and case (b). For each example, the rows show reference frames, query frames, and memory-conditioned reconstructions; the bottom plots show frame-level anomaly scores. Shaded regions indicate ground-truth anomaly intervals.

4 Experiments

4.1 Experimental Protocol

We evaluate three LOSO transfers (A+B→C, A+C→B, and B+C→A) and an external transfer to D after training on A+B+C. Each target provides ten predefined reference–query groups, and all methods use the same groups and frame-level AUROC evaluation.

4.2 Comparison with Reference-Conditioned Methods

We compare few-shot anomaly detectors, FSAD Lu et al. (2020) and AD-FSAVAD Zahid et al. (2025), with reference-conditioned matching methods, TEAM-RefNN Lee et al. (2025) and OTAM-Ref Cao et al. (2020). Table 2 shows that few-shot detectors vary substantially across targets, whereas reference-conditioned methods transfer more robustly; OTAM-Ref is the strongest LOSO baseline and TEAM-RefNN transfers best to D. CSDM consistently outperforms all baselines, supporting the combination of RTM and SPM.

Representative episodes from cases (a) and (b) illustrate the temporal behavior of CSDM in Figure 5. In case (a), the query departs from the reference operation order after an initially normal segment; the score correspondingly rises from a low baseline and remains elevated through much of the annotated interval. In case (b), anomalous components appear at distinct stages, producing peaks within some annotated intervals but weaker responses in others. The reconstruction rows provide a visual counterpart to these curves, while the varying score amplitudes indicate that response strength depends on the anomaly and its temporal extent.

Table 3: Ablation results in frame-level AUROC (%) across three LOSO settings: (a) memory components; (b) anomaly-score computation; (c) RTM retrieval radius; and (d) SPM prefix length.

(a) Memory components

Variant	A+B→C	A+C→B	B+C→A
Direct reconstruction	78.10	70.87	75.54
RTM-only	**89.03**	92.36	**92.76**
SPM-only	15.79	61.40	28.55
Full model	88.98	**93.34**	92.59

(b) Image-space scoring

Variant	A+B→C	A+C→B	B+C→A
RGB pixel	88.74	92.10	92.84
Global affine only	88.76	92.29	**92.92**
Local flow only	85.05	92.11	92.01
Global affine + local flow	**88.98**	**93.34**	92.59

(c) RTM retrieval range

RTM radius (slots)	A+B→C	A+C→B	B+C→A
40	86.17	92.34	92.02
80 (default)	**88.98**	**93.34**	**92.59**
160	88.60	90.80	91.55
All slots	87.79	85.70	86.92

(d) SPM prefix length

SPM prefix clips	A+B→C	A+C→B	B+C→A
1	87.69	92.12	92.35
5	87.68	92.14	92.35
10 (default)	**88.98**	**93.34**	**92.59**
16	87.69	92.14	92.35

4.3 Ablation Study

All ablations use the same source–target partitions, reference–query pairs, and evaluation protocol. Table 3 separates the contributions of the memory components, image-space scoring, RTM retrieval range, and SPM prefix length.

Memory components. Panel (a) shows that SPM-only performs poorly because its global scene descriptor captures appearance and layout but not the temporal state of the operation. RTM-only performs better by retrieving temporally local reference states that preserve the target-specific operation progression. It slightly exceeds the full model on targets C and A, where this reference evidence is already sufficient and SPM fusion may introduce minor reconstruction interference, while SPM improves the full model on target B by providing complementary global context.

Image-space scoring. Panel (b) compares anomaly scores computed from the same reconstructions. RGB residuals may incorrectly respond to spatial displacement, while global affine alignment removes the dominant global shift and local flow further handles residual nonrigid motion. Their combination improves detection on targets C and B. On target A, global affine-only performs slightly better because the displacement is mainly global and limited; additional local flow may introduce small warping errors or absorb subtle anomalous motion, weakening the residual contrast.

RTM retrieval range. Panel (c) shows that retrieval benefits from a moderate temporal neighborhood. A narrow range may miss the corresponding reference state under operation-speed variation, whereas an overly broad or unrestricted range admits temporally mismatched states. We therefore use $Y = 80$ as the default, based on the expected trade-off between retrieval flexibility and temporal locality; the results support this choice on the LOSO settings.

SPM prefix length. Panel (d) shows that the SPM descriptor is largely stable once several ordered query clips are available. A single or very long prefix provides no consistent benefit: the former may contain insufficient scene evidence, while the latter may include additional operation-related variation. The value of 10 was fixed as the default before the reported ablation comparison, providing a compact but sufficiently informative description of the target scene.

5 Conclusion

We presented CSDM for reference-conditioned cross-scene video anomaly detection and evaluated it under LOSO and external transfer settings. Future work will extend the evaluation to more diverse operation scenarios and recording conditions, including variations in objects, viewpoints, illumination, and execution patterns. We will also study online adaptation as operations evolve, with attention to maintaining reliable anomaly scores and computational efficiency. Finally, we plan to refine temporal annotations and systematically analyze failure cases to identify challenging anomaly types and improve the method's reliability in practical settings.

AI Disclosure

In this work,Generative AI tools were used only to assist with language polishing and proofreading of this manuscript. All scientific content, technical decisions, experimental results, and final revisions were reviewed and verified by the authors.

References

Andra Acsintoae, Andrei Florescu, Mariana-Iuliana Georgescu, Tudor Mare, Paul Sumedrea, Radu Tudor Ionescu, Fahad Shahbaz Khan, and Mubarak Shah. UBnormal: New benchmark for supervised open-set video anomaly detection. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 20143–20153, 2022.

Sunghyun Ahn, Youngwan Jo, Kijung Lee, and Sanghyun Park. VideoPatchCore: An effective method to memorize normality for video anomaly detection. In *Proc. Asian Conf. Comput. Vis. (ACCV)*, pp. 2179–2195, 2024.

Antonio Barbalau, Radu Tudor Ionescu, Mariana-Iuliana Georgescu, Jacob Dueholm, Bharathkumar Ramachandra, Kamal Nasrollahi, Fahad Shahbaz Khan, Thomas B. Moeslund, and Mubarak Shah. SSMTL++: Revisiting self-supervised multi-task learning for video anomaly detection. *Computer Vision and Image Understanding*, 234:103656, 2023.

Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. MVTec AD: A comprehensive real-world dataset for unsupervised anomaly detection. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 9592–9600, 2019.

Paul Bergmann, Kilian Batzner, Michael Fauser, David Sattlegger, and Carsten Steger. The uninformed student: Learning deep representation for anomaly detection. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 4183–4192, 2020.

Kaidi Cao, Jingwei Ji, Zhangjie Cao, Chien-Yi Chang, and Juan Carlos Niebles. Few-shot video classification via temporal alignment. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 10618–10627, 2020.

Yunpeng Chang, Zhigang Tu, Wei Xie, and Junsong Yuan. Clustering driven deep autoencoder for video anomaly detection. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*, pp. 329–345, 2020.

Georgios D. Evangelidis and Emmanouil Z. Psarakis. Parametric image alignment using enhanced correlation coefficient maximization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(10):1858–1865, 2008. doi: 10.1109/TPAMI.2008.113.

Gunnar Farneb"ack. Two-frame motion estimation based on polynomial expansion. In *Scand. Conf. Image Anal. (SCIA)*, pp. 363–370. Springer, 2003.

Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pp. 1705–1714, 2019.

Denis Gudovskiy, Shun Ishizaka, and Kazuki Kozuka. CFlow-AD: Real-time unsupervised anomaly detection with localization via conditional normalizing flows. In *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, pp. 98–107, 2022.

Chongye Guo, Hongbo Wang, Yingjie Xia, and Guorui Feng. Learning appearance-motion synergy via memory-guided event prediction for video anomaly detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(3):1519–1531, 2024. doi: 10.1109/TCSVT.2023.3297114.

Mahmudul Hasan, Jonghyun Choi, Jan Neumann, Amit K. Roy-Chowdhury, and Larry S. Davis. Learning temporal regularity in video sequences. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 733–742, 2016.

Chaoqin Huang, Haoyan Guan, Aofan Jiang, Ya Zhang, Michael Spratling, and Yan-Feng Wang. Registration-based few-shot anomaly detection. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*, pp. 303–319. Springer, 2022.

Chaoqin Huang, Haoyan Guan, Aofan Jiang, Ya Zhang, Michael Spratling, Xinchao Wang, and Yanfeng Wang. Few-shot anomaly detection via category-agnostic registration learning. *IEEE Transactions on Neural Networks and Learning Systems*, 36(7):13527–13541, 2025. doi: 10.1109/TNNLS.2024.3465446.

Jongheon Jeong, Yang Zou, Taewan Kim, Dongqing Zhang, Avinash Ravichandran, and Onkar Dabeer. WinCLIP: Zero-/few-shot anomaly classification and segmentation. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 19606–19616, 2023.

SuBeen Lee, WonJun Moon, Hyun Seok Seong, and Jae-Pil Heo. TEAM: Temporal alignment-free video matching for few-shot action recognition. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2025.

Jinfan Liu, Yichao Yan, Junjie Li, Weiming Zhao, Pengzhi Chu, Xingdong Sheng, Yunhui Liu, and Xiaokang Yang. IPAD: Industrial process anomaly detection dataset. In *Proc. ACM Int. Conf. Multimedia (ACM MM)*, 2024. 10 pages.

Wen Liu, Weixin Luo, Dongze Lian, and Shenghua Gao. Future frame prediction for anomaly detection: A new baseline. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 6536–6545, 2018.

Yang Liu, Zhaoyang Xia, Mengyang Zhao, Donglai Wei, Yuzheng Wang, Siao Liu, Bobo Ju, Gaoyun Fang, Jing Liu, and Liang Song. Learning causality-inspired representation consistency for video anomaly detection. In *Proc. ACM Int. Conf. Multimedia (ACM MM)*, pp. 203–212, 2023.

Zhian Liu, Yongwei Nie, Chengjiang Long, Qing Zhang, and Guiqing Li. A hybrid video anomaly detection framework via memory-augmented flow reconstruction and flow-guided frame prediction. In *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pp. 13588–13597, 2021.

Yiwei Lu, Frank Yu, Mahesh Kumar Krishna Reddy, and Yang Wang. Few-shot scene-adaptive anomaly detection. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*, pp. 125–141. Springer, 2020.

Weixin Luo, Wen Liu, and Shenghua Gao. A revisit of sparse coding based anomaly detection in stacked RNN framework. In *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, pp. 341–349, 2017.

Hyunjong Park, Jongyoun Noh, and Bumsub Ham. Learning memory-guided normality for anomaly detection. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 14372–14381, 2020.

Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Sch"olkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 14318–14328, 2022.

Mohammadreza Salehi, Niousha Sadjadi, Soroosh Baselizadeh, Mohammad H. Rohban, and Hamid R. Rabiee. Multiresolution knowledge distillation for anomaly detection. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 14902–14912, 2021.

Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 6479–6488, 2018.

Shengyang Sun and Xiaojin Gong. Hierarchical semantic contrast for scene-aware video anomaly detection. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 22846–22856, 2023.

Guodong Wang, Yunhong Wang, Jie Qin, Dongming Zhang, Xiuguo Bao, and Di Huang. Video anomaly detection by solving decoupled spatio-temporal jigsaw puzzles. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*, pp. 494–511, 2022.

Le Wang, Junwen Tian, Sanping Zhou, Haoyue Shi, and Gang Hua. Memory-augmented appearance-motion network for video anomaly detection. *Pattern Recognition*, 138:109335, 2023. doi: 10.1016/j.patcog.2023.109335.

Muhammad Zaigham Zaheer, Arif Mahmood, Marcella Astrid, and Seung-Ik Lee. CLAWS: Clustering assisted weakly supervised learning with normalcy suppression for anomalous event detection. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*, pp. 358–376, 2020.

Yumna Zahid, Christine Zarges, Bernie Tiddeman, and Jungong Han. Adversarial diffusion for few-shot scene-adaptive video anomaly detection. *Neurocomputing*, 614:128796, 2025. doi: 10.1016/j.neucom.2024.128796.

Ji Zhang, Xiao Wu, Zhi-Qi Cheng, Qi He, and Wei Li. Improving anomaly segmentation with multi-granularity cross-domain alignment. In *Proc. ACM Int. Conf. Multimedia (ACM MM)*, pp. 8515–8524, 2023.

Jia-Xing Zhong, Nannan Li, Weijie Kong, Shan Liu, Thomas H. Li, and Ge Li. Graph convolutional label noise cleaner: Train a plug-and-play action classifier for anomaly detection. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 1237–1246, 2019.

Hang Zhou, Jiale Cai, Yuteng Ye, Yonghui Feng, Chenxing Gao, Junqing Yu, Zikai Song, and Wei Yang. Video anomaly detection with motion and appearance guided patch diffusion model. In *Proc. AAAI Conf. Artif. Intell. (AAAI)*, pp. 10761–10769, 2025.

Qihang Zhou, Guansong Pang, Yu Tian, Shibo He, and Jiming Chen. AnomalyCLIP: Object-agnostic prompt learning for zero-shot anomaly detection. In *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2024.

Dongliang Zhu, Ruimin Hu, Shengli Song, Xiang Guo, Xixi Li, and Zheng Wang. Cross-illumination video anomaly detection benchmark. In *Proc. ACM Int. Conf. Multimedia (ACM MM)*, pp. 2516–2525, 2023.

Appendix

A Related Work

A.1 Video Anomaly Detection

Video anomaly detection learns regular visual dynamics from training videos and identifies frames or temporal segments that deviate from them. Unsupervised methods commonly reconstruct observed frames or predict future frames, using reconstruction or prediction errors as anomaly signals (Hasan et al., 2016; Liu et al., 2018). Other approaches learn normality through sparse coding, clustering, or motion-aware representations, whereas supervised and weakly supervised methods exploit anomaly annotations to learn discriminative decision functions (Luo et al., 2017; Chang et al., 2020; Sultani et al., 2018; Zhong et al., 2019; Zaheer et al., 2020; Acsintoae et al., 2022). Self-supervised methods replace anomaly labels with pretext or consistency objectives to learn spatiotemporal representations from predominantly normal videos (Wang et al., 2022; Barbalau et al., 2023; Liu et al., 2023).

Memory-augmented methods represent normality using a memory bank of normal feature prototypes and retrieve entries relevant to each input feature (Gong et al., 2019; Park et al., 2020). Subsequent methods jointly model appearance and motion, use event prediction to organize memory representations, or retain pretrained normal patch features for nearest-neighbor matching (Liu et al., 2021; Wang et al., 2023; Guo et al., 2024; Ahn et al., 2024). Beyond memory-based reconstruction, hierarchical contrastive learning models scene-dependent semantics, while diffusion-based approaches estimate normal appearance and motion through conditional generation (Sun & Gong, 2023; Zhou et al., 2025). For industrial process videos, IPAD introduces a process-oriented benchmark and uses periodic memory within a reconstruction framework (Liu et al., 2024).

A.2 Cross-Scene Generalization and Adaptation

Scene-adaptive video anomaly detection studies how a detector trained on source scenarios can be applied to a previously unseen target scenario. Existing approaches use a small set of normal target-scene frames to adapt a pretrained detector, learn scene-aware normality representations, or align source and target features under appearance changes (Lu et al., 2020; Sun & Gong, 2023; Zhu et al., 2023; Zhang et al., 2023). For industrial process videos, IPAD transfers representations between synthetic and real process settings through parameter-efficient adaptation (Liu et al., 2024). In these settings, target information is typically introduced through model adaptation or collections of normal target frames, rather than through a reference video that specifies the complete temporal progression of a multi-step operation.

Industrial image anomaly detection provides a complementary line of work on modeling normality from limited target-category data. MVTec AD established a widely used benchmark for normal-only industrial inspection (Bergmann et al., 2019). Representative methods model normality through student–teacher feature discrepancies, multi-resolution knowledge distillation, conditional normalizing flows, or coreset-based nearest-neighbor matching of pretrained patch features (Bergmann et al., 2020; Salehi et al., 2021; Gudovskiy et al., 2022; Roth et al., 2022). These methods compare a test image with normal representations learned from a target category and primarily characterize appearance and structural defects, while the target evidence is provided as a collection of independent images rather than as a temporally ordered operation.

Few- and zero-shot industrial anomaly detection studies how normal evidence from an unseen target category can support inspection with limited target-specific data. Registration-based methods align a small number of normal target images to establish transferable correspondences, while category-agnostic registration extends this idea to previously unseen categories (Huang et al., 2022; 2025). Vision–language methods instead use pretrained semantic knowledge for zero- or few-shot anomaly classification and segmentation (Jeong et al., 2023; Zhou et al., 2024). These methods reduce the need for category-specific training, but their target evidence is provided as static images or isolated frames. In contrast, our reference video specifies the temporal progression of a complete multi-step operation and is used to judge the consistency of a test video.

Table 4: Scenario-wise video partition. "Pairs" denotes the ten predefined reference–query evaluation groups in each scenario and does not imply one-to-one pairing between individual videos.

Scenario	Normal	Train	Reference	Query	Pairs	Total frames
A: Block assembly	48	38	10	100	100	73,496
B: Object placement	59	49	10	100	100	70,434
C: Screw tightening	52	42	10	100	100	128,005
D: Goods packaging	10	0	10	100	100	63,915
Total	**169**	**129**	**40**	**400**	**400**	**335,850**

Table 5: Source–target transfer partitions used for evaluation. Each target contains ten reference videos, 100 query videos, and ten predefined reference–query evaluation groups.

Transfer	Source training videos	Source frames	Target	Reference / query videos	Pairs
A+B→C	87	45,565	C	10 / 100	100
A+C→B	80	64,602	B	10 / 100	100
B+C→A	91	67,323	A	10 / 100	100
A+B+C→D	129	88,745	D	10 / 100	100

B Dataset Protocol and Splits

We evaluate three leave-one-scenario-out (LOSO) transfers and one external transfer. The scenario-wise video partition is summarized in Table 4. Source training uses only normal videos from the listed source scenarios. For each target scenario, ten normal videos are reserved as reference videos, while 100 annotated videos are used as query videos. The reference videos are included in the normal-video count and are not additional recordings. Query annotations are used only for evaluation. Scenario D is excluded from training and serves as the external target.

The resulting source-training sets contain 87 videos and 45,565 frames for A+B, 80 videos and 64,602 frames for A+C, 91 videos and 67,323 frames for B+C, and 129 videos and 88,745 frames for A+B+C. The exact source–target transfer partitions are reported in Table 5.

For each evaluation pair, the reference video is used to construct the Reference Temporal Memory (RTM). The first ten ordered query clips are used to construct the Shared Prototype Memory (SPM) scene descriptor, which is then shared by the query clips in that pair. No target-specific gradient updates are applied during testing.

Dataset Examples. Table 1a summarizes the dataset scale, while Figure 6 illustrates the visual structure of the four scenarios. The rows show complete multi-step operations recorded continuously in a single take, including block assembly, object placement, screw tightening, and goods packaging. The changing object configurations across each row demonstrate why temporal progression and reference-conditioned comparison are important for detecting process anomalies.

C Implementation Details

Each input clip contains 16 consecutive frames with a temporal stride of one and is resized to 448×448. Both training stages use Adam for 100 epochs with a batch size of one. During reconstruction pre-training, the learning rates are 1×10^{-5} for the encoder and temporal-position head and 5×10^{-5} for the decoder; the temporal-position loss uses 200 bins and a weight of 0.02. During dual-memory adaptation, the encoder is frozen and the learning rate is 5×10^{-5}. Each episode contains 16 ordered query clips and at most 120 reference clips sampled with an interval of four. We set $(\lambda^{\mathrm{dir}}, \lambda^{\mathrm{mem}}, \lambda^{\mathrm{dist}}) = (0.3, 1.0, 0.3)$. RTM uses $N = 2{,}000$ slots and a retrieval radius of $Y = 80$. SPM uses $K^{\mathrm{SPM}} = 2{,}048$ prototypes, $C^{\mathrm{f}} = 768$ feature channels, a matching temperature of $\sigma = 0.07$, and $N^{\mathrm{ctx}} = 10$ query clips with a stability temperature of 1.0. The residual gate is initialized to 0.05 and bounded by $\beta_{\max} = 0.2$. For image registration, the matching resolution is 224×224, the maximum global and local displacements are both five pixels, and frame scores are

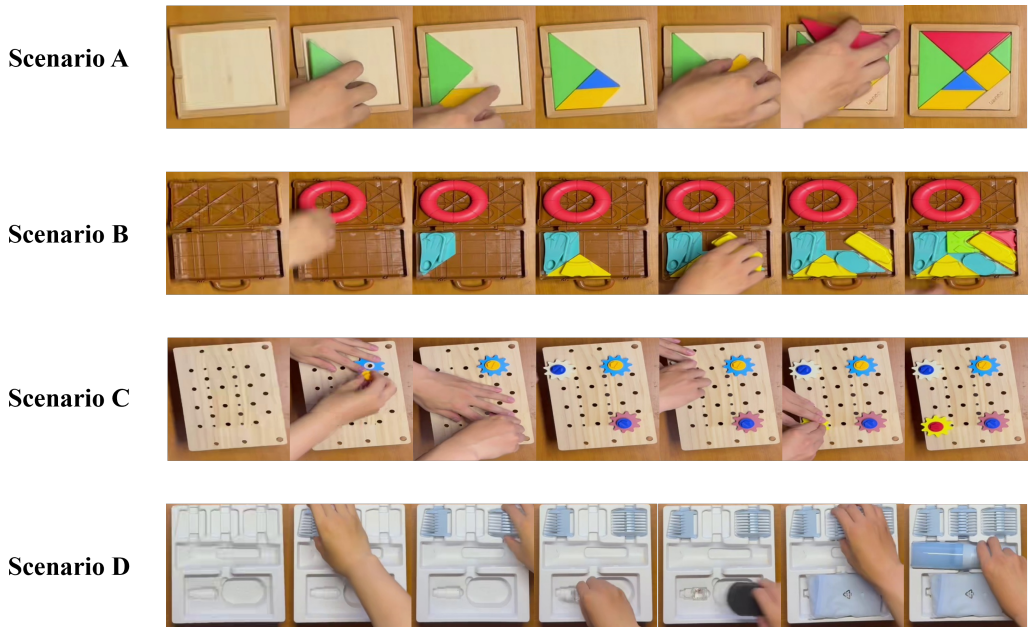

Figure 6: Representative frames from complete operation recordings in the four scenarios. From top to bottom: block assembly, object placement, screw tightening, and goods packaging. Each row shows the temporal progression of one continuous recording.

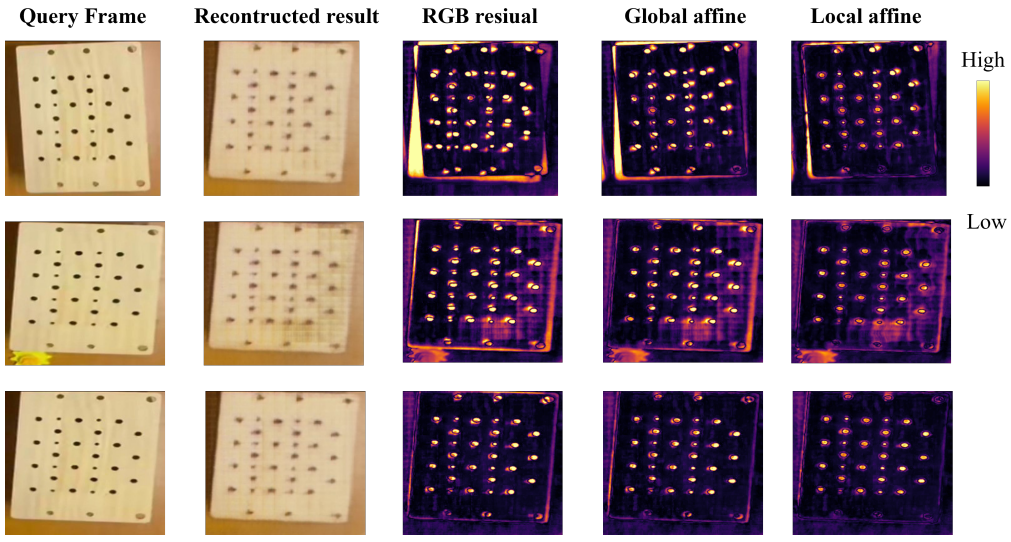

Figure 7: Overview of the image-space anomaly-score computation. The query and reconstructed frames are aligned before their residual is converted into a frame-level anomaly score.

temporally smoothed using a Gaussian kernel with a standard deviation of 10. All variants use the same settings unless otherwise specified.

Anomaly Score Computation. The final anomaly score is computed in image space from the query frame and its memory-conditioned reconstruction. Figure 7 summarizes this procedure: global affine alignment first compensates for the dominant camera or object displacement, bounded local flow then refines the remaining spatial mismatch, and the resulting residual is aggregated into a frame-level anomaly score. This design reduces responses caused by normal spatial motion while preserving discrepancies associated with abnormal operation states.

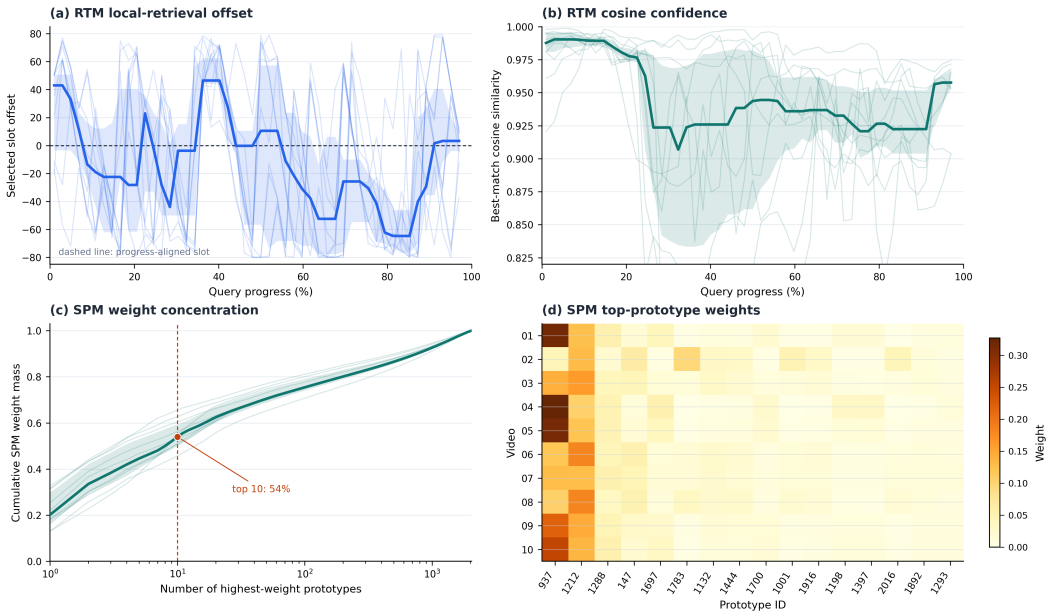

Figure 8: Dual-memory retrieval diagnostics on ten query videos. (a) RTM selected-slot offset from the progress-aligned slot over query progress; the dashed line marks zero offset. (b) Best-match cosine similarity for RTM retrieval. In (a,b), thin curves show individual videos, the thick curve is a five-bin rolling median, and shading denotes the interquartile range across videos. (c) Cumulative SPM weight mass captured by the highest-weight prototypes; the marker shows the median mass captured by the top ten. (d) SPM weights for the 16 prototypes with the highest mean weight across videos.

D Additional Experiments

D.1 Dual-Memory Retrieval Diagnostics

Figure 8 examines whether the two memories exhibit the intended division of labor. Across 5,531 query frames from ten videos, only 18.5% of RTM matches lie within ± 20 slots of the progress-aligned position, while 42.3% differ by at least 60 slots and 12.6% hit the ± 80-slot boundary. This indicates that RTM adjusts the phase estimate, while the boundary hits expose cases constrained by the fixed search window. The median match cosine similarity is 0.959.

For SPM, using a descriptor formed from the first ten ordered query clips, the weights over 2,048 prototypes are concentrated but distributed: the top prototype receives 22.2% on average, the ten highest-weight prototypes capture 55.1% (54.0% median; 46–66% across videos), and the top 50 capture 71.5%. The heatmap shows that this shared bank is reweighted differently across query videos, rather than using a fixed prototype mixture. These patterns are consistent with RTM providing target-specific local progress corrections and SPM providing a compact, query-dependent scene prior. They are descriptive evidence of retrieval behavior, not causal proof of an AUROC gain; the contribution to detection performance is evaluated separately by the controlled ablations.

D.2 Reference-Group Robustness

To examine the sensitivity of CSDM to the selected reference–query group, we report the frame-level AUROC for all ten predefined groups in each target scenario. Figure 9 shows both the per-group performance and the corresponding mean, standard deviation, and full range. The results confirm that performance varies across reference groups, while the averaged results remain consistently high across the LOSO and external transfer settings.

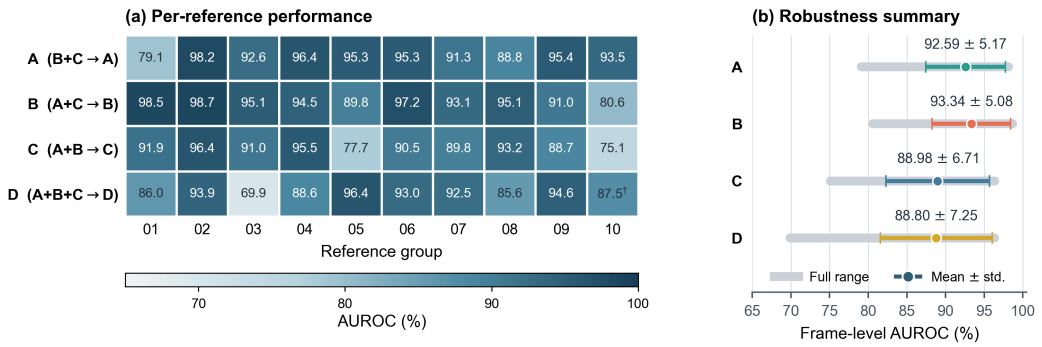

Figure 9: Reference-group robustness of the full CSDM model. The heatmap reports frame-level AUROC for the ten predefined reference–query groups in each scenario. The right panel summarizes the mean, standard deviation, and full range across groups.

Table 6: Ablation results on external target D, reported as episode-macro frame-level AUROC (%). Gray rows mark the final settings used by CSDM.

(a) Memory components

Setting	A+B+C→D
Direct reconstruction	63.24
RTM-only	**88.13**
SPM-only	28.28
Full CSDM	**88.80**

(b) Image-space scoring

Setting	A+B+C→D
RGB pixel residual	88.03
Global affine only	**88.86**
Local flow only	88.02
Global + bounded flow	88.80

(c) RTM retrieval radius

Setting	A+B+C→D
$Y = 40$	**90.12**
$Y = 80$ (default)	88.80
$Y = 160$	85.85
All slots	81.23

(d) SPM prefix length

Setting	A+B+C→D
1 clip	88.80
5 clips	88.80
10 clips (default)	88.80
16 clips	88.77

D.3 External-Target Ablations

Table 6 reports the D-specific results for all four ablation factors. Direct reconstruction uses the Stage-I model without memory construction. RTM-only is left unresolved because a matched D result was not present in the verified archive; it should be filled only after the corresponding evaluation is confirmed.

The D results expose two useful behaviors. First, the full dual-memory model substantially improves over direct reconstruction and SPM-only reconstruction, showing that target-specific temporal retrieval is important under the larger cross-scene shift. Second, the best retrieval radius on D is narrower than the default, suggesting that locally relevant temporal states can be more useful than unrestricted retrieval. The small differences among SPM prefix lengths indicate that the descriptor is stable once a short ordered prefix is available.