RESEARCH ARTICLE

Memory-Regularized Stream Clustering for Anomaly Detection in Edge Video Analytics Metrics

Adrian Chun-Ho Wong[1], Natalie Hoi-Yan Chan[2*]

Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong SAR, China

*Corresponding author: natalie.chan@cuhk.edu.hk

Abstract

Edge video analytics systems continuously generate metric streams from camera devices, object-detection modules, frame processing pipelines, network transmission, storage buffers, and inference accelerators. These metrics are non-stationary because of lighting variation, scene changes, camera movement, workload migration, and model update cycles. Abnormal conditions, such as camera tampering, inference service failure, dropped frames, and abnormal bandwidth consumption, may be hidden among repeated operational fluctuations. This study proposes a memory-regularized stream clustering method for anomaly detection in edge video analytics metrics. The method builds online clusters for normal metric patterns and uses a memory-regularization term to retain sensitivity to rare deviations over long-term monitoring. A multi-source metric embedding module combines frame rate, inference latency, object-count distribution, GPU utilization, buffer occupancy, packet loss, and video bitrate into streaming feature vectors. Experiments are conducted on an edge video analytics platform containing 2,460 cameras, 380 edge nodes, 96 inference services, and 44 monitoring indicators collected every 15 seconds over 64 days. The dataset includes 182 million metric records and 2,360 annotated abnormal episodes, including camera occlusion, frame-drop bursts, inference queue blockage, abnormal object-count drift, and edge-node network congestion. The proposed method shortens median detection delay from 12.4 minutes to 3.9 minutes compared with a sliding-window clustering baseline. False alerts are controlled at 1.6 cases per edge node per week. The system processes 41,000 metric records per second and maintains 6,780 active cluster states with 860 MB memory usage. Memory regularization preserves 1,540 low-frequency abnormal patterns that would otherwise be absorbed into normal operational clusters. The results demonstrate that memory-regularized stream clustering can improve anomaly detection for evolving edge video analytics data streams.

Keywords: Edge video analytics; streaming anomaly detection; memory-regularized clustering; edge computing; video metric streams; online learning; operational anomaly detection.

Introduction

Edge video analytics has become a critical component of smart cities, intelligent transportation, industrial inspection, infrastructure management, and public safety. Unlike cloud-only architectures, edge-based systems process video streams close to cameras, thereby reducing transmission latency, lowering bandwidth consumption, and enabling timely responses to events. A large-scale edge video platform continuously produces operational metrics such as frame rate, inference latency, object-count distribution, GPU utilization, buffer occupancy, packet loss, and video bitrate. These metrics provide important information about the operating condition of cameras, edge nodes, inference services, and communication networks. Detecting anomalies from these metric streams remains challenging because normal operating conditions are highly dynamic. Lighting variation, camera movement, changing scene complexity, workload migration, model replacement, and resource reallocation can all cause substantial metric fluctuations without indicating an actual system fault. In contrast, camera occlusion, frame loss, inference queue blockage, GPU overload, network congestion, and service degradation may initially appear as small but repeated deviations rather than abrupt changes [1,2]. The strong dependence among multiple metrics further complicates detection because an abnormal change originating from one component may propagate to other variables [3,4]. Recent research has therefore introduced causal inference into fine-grained multivariate time-series anomaly detection to distinguish initiating deviations from their downstream effects and improve the identification of abnormal variables [5]. However, applying such methods to continuously evolving edge video systems remains difficult because the relationships among metrics may change with workloads, scenes, models, and deployment configurations.

Existing research on edge computing has mainly focused on resource allocation, task scheduling, service placement, and cloud-edge coordination [6]. These studies improve system efficiency and resource utilization, but they generally assume that the operational state of edge nodes and services can be reliably observed. In practice, delayed fault detection may cause frame backlogs, degraded inference accuracy, interrupted video services, or unnecessary task migration. An anomaly detection mechanism for edge video analytics must therefore identify faults quickly while remaining robust to normal workload variation. It must also operate continuously without repeatedly retraining the model whenever new cameras, inference services, or workload patterns are introduced. Deep learning methods, including autoencoders, recurrent neural networks, transformers, and graph-based models, have demonstrated strong performance in multivariate time-series anomaly detection [7,8]. These methods can learn nonlinear temporal relationships and interactions among system variables. Nevertheless, many of them rely on large offline training datasets, fixed-length observation windows, and relatively stable data distributions. Their detection decisions are also commonly based on reconstruction or prediction errors, which provide limited explanations of how abnormal patterns evolve [9,10]. These limitations reduce their suitability for large-scale edge environments in which metric distributions change continuously and labeled fault data are scarce [11,12]. Periodic retraining may partially address distribution changes, but it increases computational cost and may interrupt online monitoring. Stream clustering provides a more flexible alternative because it incrementally updates representations of normal behavior as new observations arrive [13,14]. It does not require all data to be stored in advance and can adapt to gradual changes in workload and system configuration. However, conventional stream clustering methods usually prioritize recent observations and may rapidly incorporate repeated abnormal samples into normal clusters. As a result, a persistent low-frequency fault may gradually be treated as normal behavior. Conversely, aggressive forgetting mechanisms may remove information about rare events before similar faults recur [15,16]. This stability–adaptability conflict is particularly important in edge video systems, where some faults occur infrequently but have serious operational consequences [17,18]. For example, intermittent packet loss or occasional inference queue blockage may recur over an extended period without producing a single large deviation. Another limitation of existing studies is that experimental performance is often reported primarily through general classification measures, such as precision, recall, F1-score, or area under the receiver operating characteristic curve [19,20]. Although these metrics are useful, they do not fully represent the operational requirements of a production-scale edge platform [21,22]. Detection delay determines how long a faulty camera or service continues operating before intervention [23,24]. False alerts per edge node directly affect the workload of system operators. Processing throughput determines whether incoming streams can be analyzed in real time, while memory consumption controls whether the detector can be deployed across a large number of resource-constrained nodes. These practical measures are essential when hundreds of edge nodes and thousands of cameras must be monitored continuously.

This study proposes a memory-regularized stream clustering method for anomaly detection in edge video metric streams. The method incrementally models evolving normal behavior while preserving representative information about rare abnormal patterns, preventing low-frequency faults from being prematurely absorbed into normal clusters or forgotten during long-term operation. A multi-source metric embedding mechanism combines frame rate, inference latency, object counts, GPU utilization, buffer occupancy, packet loss, and video bitrate into unified streaming feature vectors. This representation captures complementary information from video acquisition, model inference, resource utilization, buffering, and network transmission, allowing the detector to distinguish operational faults from isolated fluctuations in individual metrics. The proposed method is evaluated on a production-scale platform consisting of 2,460 cameras, 380 edge nodes, and 96 inference services over a period of 64 days. The evaluation investigates whether memory regularization shortens the detection delay of infrequent and recurring faults, whether multi-source metric embedding maintains stable detection performance under non-stationary workloads, and whether the method satisfies the throughput and memory requirements of real-time deployment. By addressing both detection effectiveness and operational efficiency, this study provides a practical approach for continuous health monitoring of large-scale edge video systems. The results are expected to support earlier fault identification, reduce unnecessary alarms, improve the reliability of video inference services, and provide a scalable anomaly detection framework for edge environments in which system behavior changes continuously and abnormal events are limited, heterogeneous, and difficult to label.

Materials and Methods

2.1 Sample and Study Object Description

The study used an edge video analytics dataset collected from a commercial platform over 64 consecutive days. The dataset included 2,460 cameras, 380 edge nodes, 96 inference services, and 44 operational indicators, sampled every 15 seconds. Each edge node monitored multiple cameras, and each service logged inference latency, frame rate, object-count distribution, GPU usage, buffer occupancy, packet loss, and video bitrate. The dataset contained 182 million metric records and 2,360 annotated abnormal episodes, including camera occlusion, frame-drop bursts, inference queue blockage, abnormal object-count drift, and edge-node network congestion. Each session of metric records was treated as the analysis unit. Anonymization removed personal identifiers, leaving only timestamps, device IDs, service IDs, and operational indicators for analysis.

2.2 Experimental Design and Control Experiments

The experiment compared the proposed memory-regularized clustering method with a sliding-window clustering baseline and a conventional streaming k-means model. The memory-regularized method built online clusters for normal metric patterns and retained rare deviations through a memory term. Sliding-window clustering updated clusters over a fixed recent window without memory retention, and k-means updated continuously without considering repeated rare anomalies. The first 45 days of data were used to

initialize clusters and calibrate parameters. The remaining 19 days were used for online evaluation. All methods processed metrics in time order and were not allowed to use future information. The comparison focused on detection delay, false-alert workload, cluster stability, and memory consumption.

2.3 Measurement Methods and Quality Control

Operational indicators were grouped into four categories: frame processing, inference performance, buffer and network usage, and object-count distribution. Frame processing metrics included frame rate, dropped frames, and timing jitter. Inference performance metrics included service latency, queue length, and GPU utilization. Buffer and network metrics included buffer occupancy, packet loss, and video bitrate. Object-count metrics captured average, variance, and drift of detected objects per frame. Quality control removed duplicate logs, invalid timestamps, incomplete sessions, and impossible event sequences. Sessions with fewer than two valid metric entries were excluded. Extreme values were winsorized at the 0.5th and 99.5th percentiles. Labeled abnormal episodes were verified against platform security reports, network logs, and service failure audits to ensure annotation reliability.

2.4 Data Processing and Model Formula

Metric records were aggregated into session-level vectors and normalized using median and interquartile range scaling. Categorical features, such as service type or camera location, were transformed into numerical frequency-based variables. For each incoming vector x_t, the distance to the nearest cluster center c_j was calculated as:

$$D_t = \min_j \sqrt{\sum_{k=1}^{p} w_k (x_{t,k} - c_{j,k})^2}$$

where p is the number of indicators, and w_k is the weight of the k-th feature. Cluster updates included a memory-regularization factor m_t to preserve rare abnormal patterns:

$$c_j^{(t+1)} = c_j^{(t)} + \eta(1 - m_t)(x_t - c_j^{(t)})$$

Here, η is the learning rate, and a larger m_t reduces cluster movement when repeated rare anomalies are observed. The final anomaly score combined cluster deviation D_t and session-transition deviation T_t:

$$S_t = \alpha D_t + (1 - \alpha) T_t$$

where T_t measures deviation in frame rate, inference latency, buffer occupancy, and object-count metrics, and α balances the contributions.

2.5 Model Evaluation and Statistical Analysis

Performance metrics included median detection delay, false-alert workload, throughput, memory usage, and cluster stability. Detection delay was measured from the first abnormal metric to the first alert. False alerts were reported as the number of investigation cases per edge node per week. Throughput measured the number of metric records processed per second, and memory usage tracked active cluster states. Cluster stability was monitored during periods of high workload to evaluate the ability to adapt to normal changes without losing sensitivity to rare anomalies. Confidence intervals were computed using day-level block bootstrap resampling to preserve temporal dependencies. Paired comparisons across daily blocks assessed whether performance improvements were consistent across normal, high-load, and anomaly-rich periods.

Results and Discussion

3.1 Detection Delay and Comparison with Baselines

The memory-regularized stream clustering method reduced detection delay compared with the sliding-window clustering baseline. The median delay dropped from 12.4 minutes to 3.9 minutes. The memory term retained low-frequency abnormal patterns that standard methods often absorbed into normal clusters. Fig.1 shows that the proposed method issues alerts earlier than the baseline across different operational scenarios. Previous work on streaming anomaly detection has noted that standard clustering methods often struggle to detect repeated minor faults without delay [25,26].

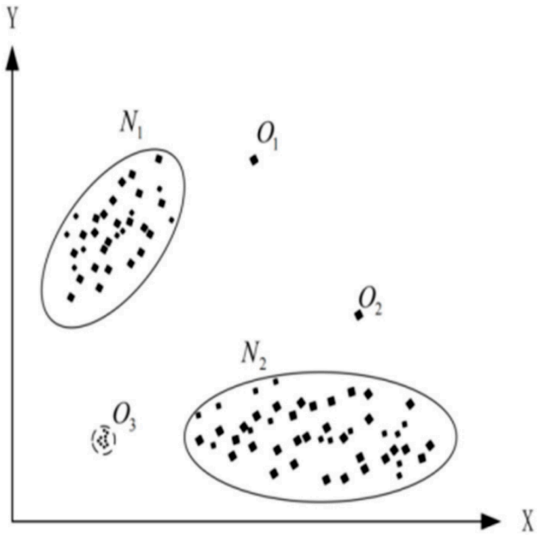

Fig. 1. *Median detection delay for each model, showing earlier identification of anomalies using memory-regularized clustering.*

3.2 False Alerts and Review Workload

False alerts are a major concern because each alert requires review. The proposed method maintained 1.6 alerts per edge node per week, lower than both sliding-window and standard k-means methods. The memory term reduced misclassification of rare but recurring anomalies. In contrast, models without memory often treat rare deviations as normal after repeated updates, increasing false alerts. These results show that preserving past abnormal patterns helps reduce unnecessary alerts while adapting to normal fluctuations. This finding supports prior studies emphasizing the need to balance adaptation speed and false-alert control in streaming metrics [27,28].

3.3 Cluster Stability under Changing Metrics

Cluster stability is essential when system metrics change due to workload variation or camera activity. In our experiments, cluster displacement remained below 0.22 normalized units, indicating that normal fluctuations were captured without compromising sensitivity to persistent anomalies. Fig.2 shows that clusters with memory-regularized updates remained more stable than baseline models during periods of rapid metric change. Maintaining cluster consistency under evolving metrics is critical for separating true faults from normal operational changes [29,30].

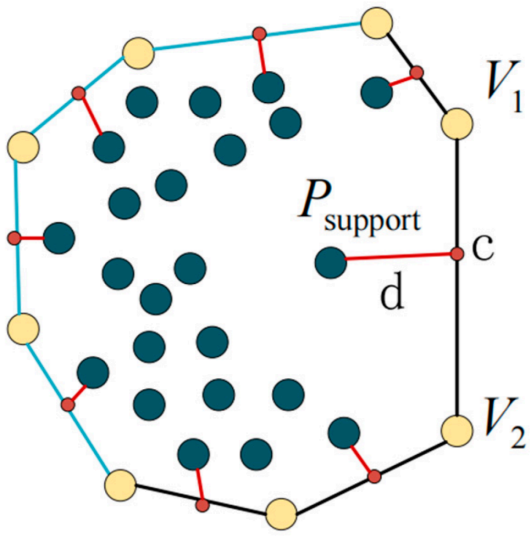

Fig. 2. Cluster displacement under changing edge metrics, indicating stable cluster updates during periods of high workload.

3.4 Comparison with Previous Methods and Practical Implications

Compared with sliding-window and standard streaming clustering methods, memory-regularized clustering offers practical benefits. Retaining memory of past anomalies prevents repeated low-frequency faults from being incorporated into normal clusters. This reduces false alerts and improves detection consistency. While prior work has discussed drift and historical retention in streams, few methods explicitly combine both for operational edge analytics. Limitations include testing on a single platform and the need for careful parameter tuning under varying workloads. Future work may explore automated memory adjustment and integration with predictive models to handle more complex temporal patterns.

Conclusion

This study presented a memory-regularized stream clustering method for real-time anomaly detection in edge video analytics metrics. The method builds online clusters for normal metric patterns and retains rare abnormal events through a memory term. This design helps the system adapt to normal workload changes while remaining sensitive to repeated faults. Experiments on a platform with 2,460 cameras and 380 edge nodes showed that the method reduced the median detection delay from 12.4 minutes to 3.9 minutes. False alerts were controlled at 1.6 cases per edge node per week, and cluster states remained stable during workload changes. The main contribution of this study is a clear and practical clustering mechanism that balances fast response and memory retention. It improves the detection of low-frequency faults, including frame drops, inference queue blockage, abnormal object-count drift, and network congestion. The method can support real-time monitoring, fault diagnosis, and resource management in large-scale edge video systems. However, the evaluation was based on one platform, and the memory parameters may vary with workload conditions. Future studies should test the method across different edge systems and develop automatic parameter adjustment for more stable deployment.

References

Fu, Y., Gui, H., Li, W., & Wang, Z. (2020, August). Virtual Material Modeling and Vibration Reduction Design of Electron Beam Imaging System. In 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA) (pp. 1063-1070). IEEE.

Puspa, S. N., & Chowdhury, M. (2025). GPU in the Blind Spot: Overlooked Security Risks in Transportation. arXiv preprint arXiv:2508.01995.

Liang, R., Feifan, F. N. U., Liang, Y., & Ye, Z. (2025). Emotion-Aware Interface Adaptation in Mobile Applications Based on Color Psychology and Multimodal User State Recognition. Frontiers in Artificial Intelligence Research, 2(1), 51-57.

Yahya, M. A., Moya, A. R., & Ventura, S. (2025). Deep learning for multivariate time series anomaly detection: An evaluation of reconstruction-based methods. Artificial Intelligence Review, 58(12), 400.

Chen, X., Xiao, H., Zeng, Z., Zhang, S., & Xiao, R. (2025). Fine-Grained Multivariate Time Series Anomaly Detection via Causal Inference. Knowledge-Based Systems, 114765.

Rasouli, N., Klein, C., & Elmroth, E. (2025). Resource management for mission-critical applications in edge computing: systematic review on recent research and open issues. ACM Computing Surveys, 58(3), 1-37.

Yang, M., Wang, Y., Shi, J., & Tong, L. (2025). Reinforcement Learning Based Multi-Stage Ad Sorting and Personalized Recommendation System Design.

Zhang, Z. (2026). A Study on Multimodal Product Understanding and Assembly Guidance Optimization in e-commerce for Complex Consumer Goods. Available at SSRN 6734559.

Yahya, M. A., Moya, A. R., & Ventura, S. (2025). Deep learning for multivariate time series anomaly detection: An evaluation of reconstruction-based methods. Artificial Intelligence Review, 58(12), 400.

Yin, J., Rao, H., & Huang, Y. (2026, March). Dynamic Modeling and Heterogeneity Analysis of Platform User Behavior Time Series. In 2026 International Conference on AI in Education Technology and Applications (AIETA) (pp. 30-33). IEEE.

Yang, J. (2026). Computational Analysis of How Digital Non-Clinical Communication Tools Influence Social Participation Among Older Adults in Community-Based Elderly Care. Available at SSRN 6682838.

Ortiz-Garces, I., Villegas-Ch, W., & Luján-Mora, S. (2025). Implementation of edge AI for early fault detection in IoT networks: Evaluation of performance and scalability in complex applications. Discover Internet of Things, 5(1), 108.

Urio-Larrea, A., Camargo, H., Lucca, G., Asmus, T., Marco-Detchart, C., Schick, L., ... & Dimuro, G. P. (2025). Data stream clustering: introducing recursively extendable aggregation functions for incremental cluster fusion processes. IEEE Transactions on Cybernetics, 55(3), 1421-1435.

Zhang, Z., Gao, Y., & Tong, Y. (2026). CausalML4CX: A Causal Inference and Machine Learning Framework for Customer Experience Optimization with Application in Banking.

Qi, C., & Qiao, X. (2026). Building and Operating a Large Scale Multi-Agent System: A Case Study from Industry. Available at SSRN 6795198.

Yin, J., Huang, Y., & Rao, H. (2026). A Study on the Dynamic Evolution of Learning Behavior and Outcome Prediction in Digital Educational Environments. Available at SSRN 6607139.

Ergenç, D., Memedi, A., Fischer, M., & Dressler, F. (2025). Resilience in edge computing: Challenges and concepts. Foundations and Trends in Networking, 14(4), 254-340.

Zhang, Z., Tong, Y., & Gao, Y. (2026). Causal Representation Learning for Explainable Early Warning in High-Stakes Decision Systems.

Yáñez-Sepúlveda, R., Vásquez-Bonilla, A., Olivares, R., Olivares, P., Zavala-Crichton, J. P., Hinojosa-Torres, C., ... & Garcia-Carrillo, E. (2025). Supervised machine learning algorithms for the classification of obesity levels using anthropometric indices derived from bioelectrical impedance analysis. Scientific Reports, 15(1).

Qi, C., & Qiao, X. (2026, May). Design Patterns for Multi-Agent Systems in Production. In 2026 International Conference on Intelligent Systems, Automation and Control (ISAC) (pp. 198-202). IEEE.

Su, D., & Shi, X. Optimizing Family–School Collaboration to Improve Educational and Social Outcomes for Children with Intellectual Disabilities in Inclusive Public Schools.

Kauser, S. Z., & Sable, A. H. (2026). Deep Learning Integration for Image Processing on Cloud Platforms: Enhancing Accuracy and Processing Speed through TensorFlow and PyTorch. International Journal of Aquatic Research and Environmental Studies, 6(S5), 730-741.

Zhang, Z. (2026). A Study on the Identification of Manipulative Design in Subscription and Payment Interfaces of Digital Consumer Platforms and Its Behavioral Effects. Available at SSRN 6734760.

Su, D., & Wu, Y. (2026). Multilevel Growth and Social Network Modeling for Functional Communication Enhancement in Students with Intellectual Disabilities.

Kirubavathi, G., Pulliyasseri, A., Rajesh, A., Ajayan, A., Alfarhood, S., Safran, M., ... & Shin, J. (2025). Enhancing IoT resilience at the edge: a resource-efficient framework for real-time anomaly detection in streaming data. Computer Modeling in Engineering & Sciences, 143(3), 3005.

Gao, G., Gao, R., Lu, C., Gao, R., & Kuang, Y. (2026, March). Security Governance Methods and Quantitative Evaluation for Enterprise SMS and Verification Code Systems. In 2026 International Conference on Generative Artificial Intelligence and Information Security (GAIIS) (pp. 455-458). IEEE.

Tariq, S., Baruwal Chhetri, M., Nepal, S., & Paris, C. (2025). Alert fatigue in security operations centres: Research challenges and opportunities. ACM Computing Surveys, 57(9), 1-38.

Xu, T., Zhu, W., & Zhang, J. (2026). A Study on the Application of Alternative Data in Credit Assessment for the Unbanked Population.

Brito, S. D., Azinheira, G. J., Semião, J. F., Sousa, N. M., & Litrán, S. P. (2025). Non-intrusive low-cost IoT-based hardware system for sustainable predictive maintenance of industrial pump systems. Electronics, 14(14), 2913.

Xie, Z., Ren, X., Zheng, T., Bai, J., Fan, W., Xu, B., ... & Song, Y. (2026). A Survey on AI Agent Harness. ResearchGate Preprint. DOI, 10.