Advanced Engineering Informatics 73 (2026) 104537

Contents lists available at ScienceDirect

Advanced Engineering Informatics

journal homepage: www.elsevier.com/locate/aei

Full length article

HyPCV-Former: Hyperbolic spatio-temporal transformer for 3D point cloud video anomaly detection

Jiaping Cao [a,b], Kangkang Zhou [c], Juan Du [a,d,*]

[a] The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, 511453, China
[b] National University of Defense Technology, Changsha, 410073, China
[c] University of Chinese Academy of Sciences, Beijing, 101408, China
[d] The Hong Kong University of Science and Technology, Hong Kong Special Administrative Region of China

ARTICLE INFO

Keywords:
3D point cloud
Video anomaly detection
Hyperbolic space
Transformer

ABSTRACT

Video anomaly detection is a fundamental task in video surveillance, with broad applications in public safety and intelligent monitoring systems. Although previous methods leverage Euclidean representations in RGB or depth domains, such embeddings are inherently limited in capturing hierarchical event structures and spatio-temporal continuity. To address these limitations, we propose HyPCV-Former, a novel hyperbolic spatio-temporal transformer for anomaly detection in 3D point cloud videos. Our approach first extracts per-frame spatial features from point cloud sequences via point cloud extractor and then embeds them into Lorentzian hyperbolic space, which better captures the latent hierarchical structure of events. To model temporal dynamics, we introduce a hyperbolic multi-head self-attention (HMHA) mechanism that leverages Lorentzian inner products and curvature-aware softmax to learn temporal dependencies under non-Euclidean geometry. All feature transformations and anomaly scoring are all directly operated within full Lorentzian space. Extensive experiments demonstrate that HyPCV-Former achieves state-of-the-art performance across multiple anomaly categories, with a 7% improvement on the TIMo dataset and a 5.6% gain on the DAD dataset compared to benchmarks.

1. Introduction

Video anomaly detection (VAD), often referred to as video violence detection, is a fundamental task in video surveillance that aims to identify abnormal events deviating from expected patterns [1]. Typical scenario-level anomalies in video surveillance include street fights, supermarket robberies, and unattended luggage on trains [2,3]. Additionally, crowds gathering or people running collectively in the same direction may indicate potential anomalies relevant to emergency management [4]. To address this, current methods often rely on RGB images [5,6], which provide rich semantic information. These methods extract per-frame features in Euclidean space and predict whether each frame corresponds to an anomalous event.

Unlike anomalies typically found in images, violent or anomalous events in videos are challenging to detect from single frames, as these frames often lack obvious geometric defects or color irregularities. Therefore, video anomaly detection requires analyzing a sequence of frames that collectively illustrate hierarchical structures, comprising frames before, during, and after an anomaly, as depicted in Fig. 1(a). However, anomaly detection based on RGB images is often sensitive to lighting conditions and lacks aggregated defect features. To leverage accurate 3D spatial information instead of conventional 2D data, some approaches utilize human pose estimation [7,8] or range images [9,10], as illustrated in Fig. 1(b) and (c). Nevertheless, pose estimation methods typically reconstruct 3D coordinates indirectly from 2D images, and range images used in anomaly detection tasks are computationally intensive and unsuitable for analyzing complex spatial structures [11]. In contrast, 3D point clouds are inherently unstructured and unordered, consisting of discrete points distributed across object surfaces.

3D point cloud videos, also referred to as 3D point cloud streams, consist of sequential frames of 3D point clouds [12]. To our knowledge, anomaly detection within 3D point cloud videos has not yet been extensively explored. Notably, some existing studies [13] directly utilize 3D point clouds to analyze human dynamics and detect anomalies, yet they learn point representations solely in Euclidean space, thereby overlooking the exponential growth of distances between normal and anomalous frames. Hyperbolic space is particularly effective for representing hierarchical or tree structures due to its exponential relationship between node quantity and tree depth, contrasting with the

* Corresponding author.
E-mail address: juandu@ust.hk (J. Du).

https://doi.org/10.1016/j.aei.2026.104537
Received 1 December 2025; Received in revised form 29 January 2026; Accepted 27 February 2026
Available online 5 March 2026

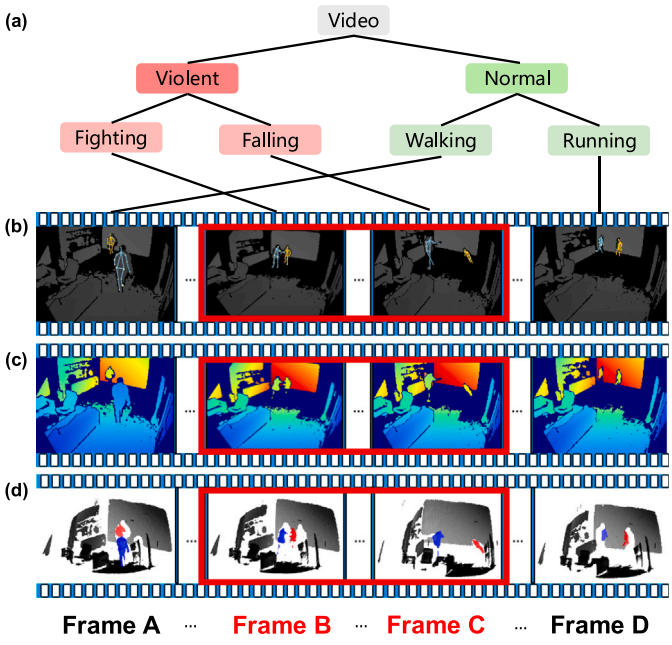


Fig. 1. Illustration of VAD and 3D data acquisition. (a) Hierarchical diagram in VAD. (b–d) Visualization of video frames using (b) human pose estimation, (c) depth images, and (d) 3D point clouds. Frames A and D are normal; Frames B and C are anomalous.

polynomial relationship characteristic of Euclidean space. Existing studies have applied hyperbolic representation learning to video anomaly detection tasks [1,14]. For example, HyperVD [14] employs Hyperbolic Graph Convolutional Networks (HGCN) to capture feature similarities and temporal relationships in hyperbolic space, while DSRL [1] integrates Euclidean and hyperbolic representations to enhance anomaly detection performance. However, both HyperVD [14] and DSRL [1] perform key operations such as linear transformations, neighbor aggregation, dropout, and nonlinear activations in the tangent space rather than directly in hyperbolic space. As discussed in prior work [15] , such tangent-space-based designs often require frequent logarithmic and exponential mappings, which may increase computational overhead and lead to accumulated numerical errors. This observation motivates recent efforts toward fully hyperbolic operations.

To address aforementioned drawbacks and challenges, we propose a novel 3D point cloud video anomaly detection recipe based on hyperbolic spatio-temporal transformer, abbreviating to HyPCV-Former. We use 3D point clouds modality to effectively capture geometric features [13] and introduce hyperbolic space to differentiate between normal and anomalous events, particularly ambiguous violence in surveillance videos [1,14]. In order to better learn the temporal dependency within frame sequences, we design the hyperbolic multi-head self-attention (HMHA) mechanism. Additionally, to efficiently manage irregular and unordered foreground points, we first apply background removal utilizing a motion detection approach specifically designed for range images [16]. Lorentzian intrinsic distance is used as an anomaly score, ensuring that all computations are performed entirely within hyperbolic space. Our key contributions can be summarized as follows:

- We propose a hyperbolic spatio-temporal transformer for 3D point cloud videos, which amplifies the separation between normal and abnormal instances to enhance anomaly discrimination.
- We introduce an HMHA mechanism operating entirely in Lorentzian space to model frame-level dynamics and enhance spatio-temporal representations for anomaly prediction.
- To the best of our knowledge, HyPCV-Former is the first to leverage hyperbolic geometry for anomaly detection in 3D point cloud videos, achieving state-of-the-art performance on violent event detection.

The remainder of this paper is organized as follows: Section 2 reviews the related literature. Section 3 describes fundamental information of hyperbolic geometry. Section 4 gives details of our proposed method. Section 5 presents the experimental evaluation, and Section 6 concludes the paper.

2. Related work

2.1. 3D point cloud analysis

3D Point cloud data has received a lot of attention for its superior accuracy and robustness in a variety of adverse situations [17]. Some previous works firstly transfer point clouds into octrees [18] or hashed voxel lists [19] to address the unstructured challenge of 3D point cloud. Others use point-based deep architecture to learn individual point representation through stacking several multilayer perceptron (MLP), such as PointNet [20] and PointNet++ [21]. In addition, there are some existing works that regard point clouds as graphs in Euclidean space to capture dependencies between adjacent points. For example, DGCNN [22] dynamically constructs a graph at each layer, where the edges are defined by the k-nearest neighbors in feature space. PointMLP [23] discards sophisticated local geometric extractors in favor of a lightweight geometric affine module and a deeper MLP design. This approach efficiently captures and fuses local geometry, achieving leading performance on multiple datasets. Thus, we introduce 3D point clouds to obtain geometric characteristics of each frame, which can be better projected into features containing spatial information related to anomalies and protecting personal information.

2.2. Video anomaly detection

Due to the rarity and unpredictability of anomalous events, compiling exhaustive labeled datasets encompassing all potential anomalies is generally infeasible [9]. Consequently, unsupervised methods are particularly advantageous, as they primarily seek to model normal spatio-temporal patterns without presupposing the nature of anomalies. These methods detect anomalies by identifying substantial deviations from the learned manifold of standard motion and appearance. Unsupervised approaches in this domain are primarily categorized into reconstruction-based and prediction-based methods [3]. However, these recipes are predominantly applied to RGB video, where the lack of precise coordinate information and potential disclosure of personal details remain notable concerns [13].

To address these issues, a range of 3D-based strategies for anomaly detection have been explored, encompassing human pose estimation, depth imaging, and point cloud video streams. Specifically, Zhang et al. [7,8] employ human pose estimation approaches to extract 3D human information from frame sequences. Besides, Schneider et al. [9] employ Time-of-Flight (ToF) depth images to capture 3D information for unsupervised video anomaly detection, highlighting the advantages of depth data in robust geometric representation and reduced privacy risk. He et al. [13] leverage point cloud video sequences to obtain more precise 3D spatial information for video anomaly detection. By converting each depth frame into a 3D point cloud representation and applying a specialized reconstruction-based autoencoder, their approach captures fine-grained geometric and motion details while maintaining privacy.

2.3. Hyperbolic learning

In representation learning, hyperbolic geometry has been extensively studied for its ability to model complex non-Euclidean data, demonstrating enhanced representational capacity and generalization when dealing with hierarchical structures [24]. Leveraging hyperbolic space, numerous neural network architectures have been devised to exploit the geometric advantages it offers [25–27]. Hyperbolic geometry has found wide-ranging applications in computer vision [28–30], recommendation [31,32], and graph learning [33–35]. Leng et al. [1] propose a hyperbolic neural network-based framework for video violence detection, leveraging hyperbolic space to capture hierarchical structures more effectively. However, they also highlight that relying too frequently on hyperbolic operations risks destabilizing the training process, underscoring the need for strategies that maintain stable representation learning.

Recently, there have also been efforts to adapt Transformers and other neural architectures to hyperbolic or mixed-curvature spaces. Gulcehre et al. [36] propose hyperbolic attention networks, which reinterpret the standard soft-attention mechanism in hyperbolic geometry to better encode hierarchical data and empirically verify the resulting improvements. Chen et al. [37] extend this direction by formulating a fully hyperbolic network in the Lorentz model, avoiding reliance on tangent-space transformations and thereby enhancing both representation ability and computational stability. Shimizu et al. [38] further refine hyperbolic network components under the HNN++ framework, introducing more parameter-efficient implementations of hyperbolic multinomial logistic regression and convolutional layers. Finally, Cho et al. [39] target mixed-curvature Transformers, demonstrating that by end-to-end learning of curvature parameters, the architecture can flexibly adapt to complex relational structures and improve performance on various graph-centered tasks.

Overall, existing research reveals several fundamental limitations across the related domains. First, most 3D point cloud methods are tailored for static 3D point cloud sets and thus fail to generalize to 3D point cloud videos, which require explicit modeling of temporal evolution. Second, mainstream video anomaly detection approaches remain predominantly RGB-based, raising concerns regarding privacy exposure and limiting their applicability in privacy-sensitive scenarios. Third, prior 3D-based anomaly detection strategies rely almost exclusively on Euclidean representations, which are inherently constrained in capturing the hierarchical structures commonly exhibited in violent videos. Finally, although recent advances in hyperbolic representation learning demonstrate strong potential for modeling hierarchical data, existing hyperbolic frameworks typically depend on frequent mappings between hyperbolic space and tangent space, lacking fully hyperbolic operations that preserve geometric consistency throughout the entire pipeline. These gaps collectively underscore the need for a unified, privacy-preserving, and geometrically expressive framework for anomaly detection in 3D point cloud videos.

3. Preliminaries

3.1. Problem definition

Point cloud video anomaly detection aims to identify anomalous events in spatial and temporal dimensions within a sequence of 3D point cloud frames. Formally, a point cloud video is represented as a sequence $\mathcal{P} = \{\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_T\}$, where tth frame $\mathbf{P}_t \in \mathbb{R}^{N\times 3}$ consists of N points, with each point described by its 3D coordinates (x, y, z). Each frame $\mathbf{P}_t$ has its label L_t that can supervise the training of anomaly score. In our study, we regard VAD as a prediction or a classification task according to different benchmarks.

At testing time, the model evaluates a point cloud sequence $\mathcal{P}$ and assigns an anomaly score s_{T+1} to the target frame $\mathbf{P}_{T+1}$ based on its deviation from learned spatio-temporal patterns in the prediction setting:

$$s_{T+1} = \mathcal{D}\left(\mathbf{P}^{\mathbb{L}}_{T+1}, f_\theta(\mathcal{P})\right), \tag{1}$$

where $\mathcal{D}(\cdot,\cdot)$ denotes a prediction-based distance, and $\mathbf{P}^{\mathbb{L}}_{T+1} \in \mathbb{L}^{N\times 3}_{\kappa}$ represents the projection of $\mathbf{P}_{T+1}$ into hyperbolic space. Frames with high anomaly scores are flagged as abnormal, indicating significant deviations from expected behavior. In classification setting, the objective is to learn a model f_θ that assigns an anomaly score to the target frame, reflecting the likelihood of abnormal behavior.

3.2. Lorentz model for hyperbolic geometry

Hyperbolic geometry, a non-Euclidean geometry with constant negative curvature, has proven particularly powerful for modeling hierarchical and tree-like structures often encountered in complex datasets. Among various hyperbolic geometric models, including the Poincaré ball [25] and Lorentz models [40], the Lorentz model stands out due to its computational efficiency, numerical stability, and simplicity in defining geometric operations [14,37].

3.2.1. Lorentz model

Formally, the Lorentz model, also known as the hyperboloid model, is represented as an n-dimensional Riemannian manifold $\mathbb{L}^n_\kappa = (\mathcal{L}^n, g^\kappa_{\mathbf{x}})$, where $\kappa < 0$ indicates a negative constant curvature. The set $\mathcal{L}^n$ describes an upper hyperboloid sheet in an $(n+1)$-dimensional Minkowski space, given by:

$$\mathcal{L}^n = \{\mathbf{x} \in \mathbb{R}^{n+1} : \langle \mathbf{x}, \mathbf{x}\rangle_{\mathcal{L}} = 1/\kappa, \mathbf{x}_0 > 0\}, \tag{2}$$

where $\langle\cdot,\cdot\rangle_{\mathcal{L}}$ is the Lorentzian scalar product defined as:

$$\langle \mathbf{x}, \mathbf{y}\rangle_{\mathcal{L}} = -x_0 y_0 + \sum_{i=1}^{n} x_i y_i. \tag{3}$$

The Lorentzian scalar product distinguishes the temporal axis (x_0) from the spatial axes $(x_i, i \geq 1)$, borrowing terminology from special relativity.

3.2.2. Tangent space

At any point $\mathbf{x} \in \mathbb{L}^n_\kappa$, the tangent space $\mathcal{T}_{\mathbf{x}}\mathbb{L}^n_\kappa$ represents a local linear approximation to the hyperboloid and is defined by:

$$\mathcal{T}_{\mathbf{x}}\mathbb{L}^n_\kappa = \{\mathbf{y} \in \mathbb{R}^{n+1} : \langle \mathbf{y}, \mathbf{x}\rangle_{\mathcal{L}} = 0\}. \tag{4}$$

This tangent space is essentially a Euclidean subspace embedded in $\mathbb{R}^{n+1}$ and is crucial for performing optimization and mapping operations in hyperbolic neural networks [15].

3.2.3. Exponential and logarithmic maps

Operations such as neural network training in hyperbolic spaces require seamless transitions between hyperbolic manifolds and their tangent spaces. This transition is achieved via exponential and logarithmic maps, essential mathematical tools defined in hyperbolic geometry [1]. The exponential map $\exp^\kappa_{\mathbf{x}}$ transfers a vector from the tangent space at point x to the hyperbolic manifold, mathematically defined as:

$$\exp^\kappa_{\mathbf{x}}(\mathbf{v}) = \cosh(\alpha)\mathbf{x} + \sinh(\alpha)\frac{\mathbf{v}}{\alpha}, \quad \alpha = \sqrt{-\kappa\langle \mathbf{v}, \mathbf{v}\rangle_{\mathcal{L}}}. \tag{5}$$

Conversely, the logarithmic map $\log^\kappa_{\mathbf{x}}$ projects points from the manifold back onto the tangent space:

$$\log^\kappa_{\mathbf{x}}(\mathbf{y}) = \frac{\operatorname{arcosh}(\beta)}{\sqrt{\beta^2 - 1}}(\mathbf{y} - \beta\mathbf{x}), \quad \beta = \kappa\langle \mathbf{x}, \mathbf{y}\rangle_{\mathcal{L}}. \tag{6}$$

These mappings enable efficient geometric manipulations necessary for representation learning and model optimization within hyperbolic spaces, particularly in tasks involving hierarchical data structures such as anomaly detection in 3D point clouds [41].

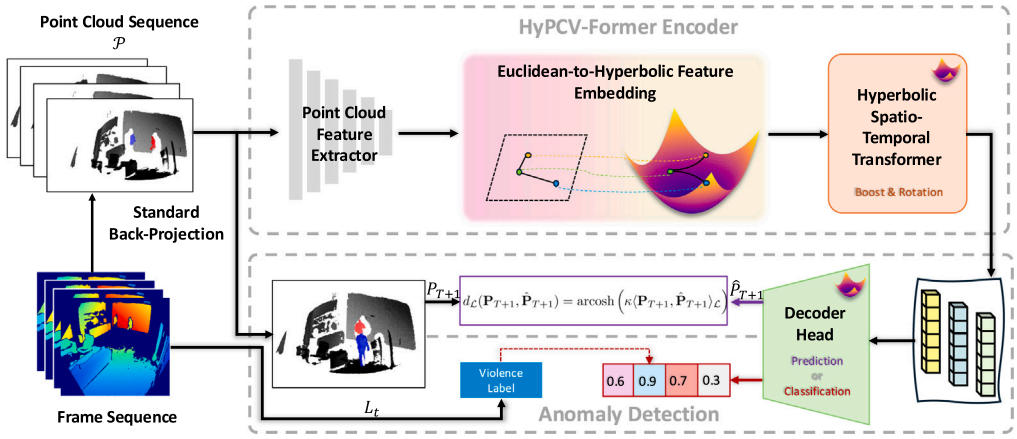


Fig. 2. The pipeline of HyPCV-Former.

4. Method

In this section, we present the overall framework of HyPCV-Former for anomaly detection in 3D point cloud videos, as depicted in Fig. 2. Our approach consists of two stages: (1) HyPCV-Former encoder, (2) anomaly detection. There are two operations in the encoder, including hyperbolic representation learning and hyperbolic spatio-temporal transformation. Before the overall encoder module, we first project range images into 3D point clouds and then remove the background points since anomalies appear in the foreground part. Based on the learned representations, an anomaly score is computed for the target frame using a task-specific criterion defined in hyperbolic space.

4.1. Hyperbolic representation

4.1.1. Per-frame point cloud feature extraction

To extract frame-wise features from each point cloud $\mathbf{P}_t \in \mathbb{R}^{N\times 3}$, we employ PointNet [20] for its efficiency and permutation invariance. Other backbones such as PointMLP [23] and DGCNN [22] are also compatible with our framework. Each frame is encoded into a D-dimensional Euclidean feature vector:

$$\mathbf{x}_t = \text{PointNet}(\mathbf{P}_t) \in \mathbb{R}^D. \tag{7}$$

Thus, the video sequence is transformed into a Euclidean tensor $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_T] \in \mathbb{R}^{B\times T\times D}$, where B is the batch size, T is the number of frames, and D is the dimensionality of features per-frame.

4.1.2. Euclidean-to-hyperbolic feature embedding

The extracted Euclidean features inherently fail to encode hierarchical or complex structured information. To map Euclidean features onto hyperbolic manifold $\mathbb{L}_\kappa^D$, we first associate each Euclidean feature vector $\mathbf{x}_t \in \mathbb{R}^D$ with a vector on the tangent space $\mathcal{T}_{\mathbf{o}}\mathbb{L}_\kappa^D$ at the manifold origin $\mathbf{o} = (\sqrt{-1/\kappa}, 0, \dots, 0)$:

$$\mathbf{x}_t^{(0)} = [0, \mathbf{x}_t] \in \mathcal{T}_{\mathbf{o}}\mathbb{L}_\kappa^D. \tag{8}$$

Then, we use the exponential map $\mathbf{x}_t^{\mathbb{L}} = \exp_{\mathbf{o}}^{\kappa}\left(\mathbf{x}_t^{(0)}\right)$ at the origin $\mathbf{o}$ to project $\mathbf{x}_t^{(0)}$ onto the hyperbolic manifold $\mathbb{L}_\kappa^D$. Consequently, the Euclidean features from the entire frame sequence are transformed into a hyperbolic representation $\mathbf{X}^{\mathbb{L}} = [\mathbf{x}_1^{\mathbb{L}}, \mathbf{x}_2^{\mathbb{L}}, \dots, \mathbf{x}_T^{\mathbb{L}}] \in \mathbb{L}_\kappa^{B\times T\times(D+1)}$.

4.2. Hyperbolic spatio-temporal transformation

We propose a Hyperbolic Spatio-Temporal Transformer to effectively capture complex spatio-temporal dependencies in hyperbolic embeddings, as shown in Fig. 3. Specifically, given a Lorentzian representation $\mathbf{X}^{\mathbb{L}} \in \mathbb{L}_\kappa^{B\times T\times(D+1)}$, our model builds on two core modules, inspired by Yang et al. [15], including hyperbolic transformation with curvatures (HTC) and hyperbolic readjustment and refinement with curvatures (HRC). The hyperbolic transformation with curvatures (HTC) module for hyperbolic embeddings is defined as follows [15]:

$$\text{HTC}(\mathbf{x}_t^{\mathbb{L}}; \mathbf{W}_h, \kappa_1, \kappa_2) = \left(\sqrt{\frac{\kappa_1}{\kappa_2}\|\mathbf{x}_t^{\mathbb{L}}\mathbf{W}_h\|_2^2 - \frac{1}{\kappa_2}}, \sqrt{\frac{\kappa_1}{\kappa_2}}\mathbf{x}_t^{\mathbb{L}}\mathbf{W}_h\right), \tag{9}$$

where $\mathbf{W}_h$ denotes the learnable parameters for each attention head, and κ_1, κ_2 represent the curvatures before and after the transformation.

4.2.1. Hyperbolic positional encoding

To explicitly incorporate sequential information, we introduce learnable hyperbolic positional embeddings $\mathbf{p}_t := \text{HTC}(\mathbf{x}_t^{\mathbb{L}}) \in \mathbb{L}_\kappa^{D+1}$. Given the original hyperbolic embeddings $\mathbf{x}_t^{\mathbb{L}}$, the positional encoding operation is defined fully within hyperbolic space as [15,42,43]:

$$\mathbf{x}_t' = \frac{\mathbf{x}_t^{\mathbb{L}} + \epsilon\cdot\mathbf{p}_t}{\sqrt{|\kappa|}\|\mathbf{x}_t^{\mathbb{L}} + \epsilon\cdot\mathbf{p}_t\|_{\mathcal{L}}}, \tag{10}$$

where ϵ is a scaling hyperparameter which is set to 1 in the experiment, and the Lorentzian norm $\|\cdot\|_{\mathcal{L}}$ ensures that the resulting embedding remains valid on the hyperbolic manifold. Such positional encoding explicitly embeds temporal information into the representations $\mathbf{X}' = [\mathbf{x}_1', \mathbf{x}_2', \dots, \mathbf{x}_T'] \in \mathbb{L}_\kappa^{B\times T\times(D+1)}$.

4.2.2. Hyperbolic multi-head self-attention

The architecture of the HMHA is shown in Fig. 3. For each attention head h, we first compute query $\mathbf{Q}_h$, key $\mathbf{K}_h$, and value $\mathbf{V}_h$ through hyperbolic linear transformations defined as follows:

$$\mathbf{Q}_h = \text{HTC}(\mathbf{X}'; \mathbf{W}^{\mathbf{Q}}\mathbf{Q}_h) \tag{11}$$

$$\mathbf{K}_h = \text{HTC}(\mathbf{X}'; \mathbf{W}^{\mathbf{K}}\mathbf{K}_h) \tag{12}$$

$$\mathbf{V}_h = \text{HTC}(\mathbf{X}'; \mathbf{W}^{\mathbf{V}}\mathbf{V}_h), \tag{13}$$

where $\mathbf{Q}_h, \mathbf{K}_h, \mathbf{V}_h \in \mathbb{L}_K^{B\times T\times D_h}$, $D_h = (D+1)/H$, and H represents the number of heads.

Attention is computed using negative Lorentzian distances, where the full attention matrix is constructed via Lorentzian inner products followed by softmax normalization, as defined in Eq. (14). This enables effective modeling of complex spatio-temporal dependencies in sequential data.

$$A_h = \text{softmax}\left(\frac{2 + 2\langle \mathbf{Q}_h, \mathbf{K}_h\rangle_{\mathcal{L}}}{\sqrt{D_h}} + b_h\right), \tag{14}$$

where A_h is the attention weight for head h, and b_h is a learnable scalar bias.

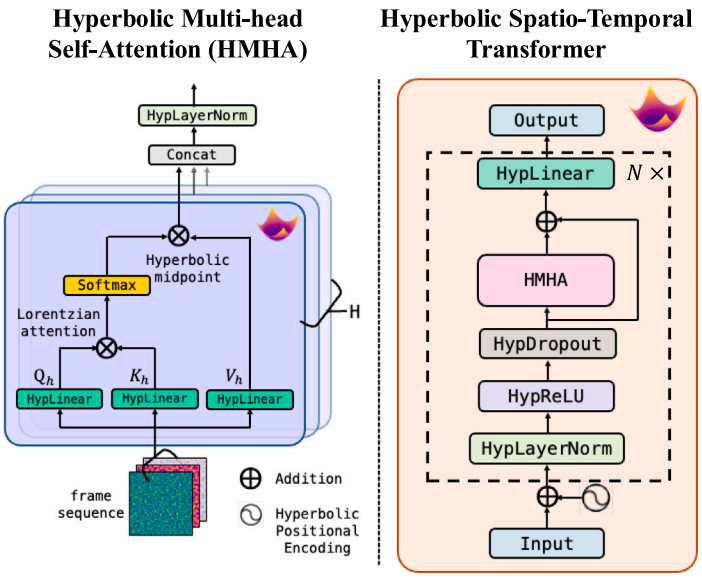


Fig. 3. The details of Hyperbolic Multi-head Self-Attention and Hyperbolic Spatio-Temporal Transformer.

We aggregate these attention-weighted values using the Lorentzian midpoint [42]:

$$A_h \odot^{\kappa} \mathbf{V}_h := \frac{A_h \mathbf{V}_h}{\sqrt{|\kappa|}\left\|A_h \mathbf{V}_h\right\|_{\mathcal{L}}}, \tag{15}$$

where $A_h \odot^{\kappa} \mathbf{V}_h$ is denoted as $\mathbf{O}_h$. Then, concatenating the outputs from all heads, we obtain the final attention output embeddings $\mathbf{O} \in \mathbb{L}_{\kappa}^{B\times T\times(D+1)}$.

4.2.3. Hyperbolic non-linear operation

Layer normalization, activation, and dropout are essential components of the Hyperbolic Spatio-Temporal Transformer. We implement all these non-linear operations using the HRC module, which enables curvature-aware refinement within the Lorentzian manifold. The HRC ensures stable and effective manifold embeddings through several hyperbolic-specific operations, including hyperbolic normalization, activation, and dropout. Specifically, the HRC module is defined as follows [15]:

$$\mathrm{HRC}(\mathbf{O}; f_r, \kappa_1, \kappa_2) = \left(\sqrt{\frac{\kappa_1}{\kappa_2}\|f_r(\mathbf{O})\|_2^2 - \frac{1}{\kappa_2}}, \sqrt{\frac{\kappa_1}{\kappa_2}} f_r(\mathbf{O})\right), \tag{16}$$

Specifically, the refining function $f_r(\cdot)$ encapsulates hyperbolic layer normalization $f_{\mathrm{LayerNorm}}(\cdot)$, hyperbolic activation $f_\sigma(\cdot)$, and hyperbolic dropout $f_{\mathrm{Dropout}}(\cdot)$, as formally defined as follows:

$$\begin{aligned} \mathrm{HypLayerNorm}(\mathbf{O}) &= \mathrm{HRC}(\mathbf{O}, f_{\mathrm{LayerNorm}}), \\ \mathrm{HypReLU}(\mathbf{O}) &= \mathrm{HRC}(\mathbf{O}, f_\sigma), \\ \mathrm{HypDropout}(\mathbf{O}) &= \mathrm{HRC}(\mathbf{O}, f_{\mathrm{Dropout}}). \end{aligned} \tag{17}$$

The output of HRC is a refined hyperbolic embedding $\hat{\mathbf{X}} = \mathrm{HRC}(\mathbf{O}; f_r, \kappa_1, \kappa_2) \in \mathbb{L}_{\kappa}^{B\times T\times(D+1)}$, suitable for downstream tasks, where κ_1, κ_2 represent the curvatures before and after the transformation.

4.3. Anomaly detection

We estimate an anomaly score for each frame in a 3D point cloud video by measuring deviations from learned spatio-temporal patterns. Traditional Euclidean metrics, such as mean squared error or cosine similarity, often fail to capture the hierarchical geometry inherent in 3D data. To address this, we adopt the Lorentzian intrinsic distance in hyperbolic space, as defined in Eq. (18), providing a more expressive and geometry-aware measure of deviation. This formulation naturally adapts to diverse scoring settings while preserving geometric consistency.

$$d_{\mathcal{L}}(\hat{\mathbf{X}}, \mathbf{P}_{T+1}) = \operatorname{arcosh}\left(\kappa\langle \mathrm{Decoder}(\hat{\mathbf{X}}), \mathbf{P}_{T+1}^{\mathbb{L}}\rangle_{\mathcal{L}}\right) \tag{18}$$

where Decoder$(\cdot)$ is the multilayer perceptron (MLP). In the classification setting, the decoder head directly outputs an anomaly score, and the Lorentzian intrinsic distance, defined in Eq. (18), is employed as a hyperbolic loss to supervise the learning process.

Table 1
Parameter settings for the foreground segmentation method.

Param.	K	K_{kinect}	$\Delta P_{\max}$	α	T_{W}	N_{H}
Value	1.25	$5 \cdot 10^{-4}$	100	0.4	300	90

4.4. Foreground mask generation

Foreground segmentation is a critical step in video anomaly detection because anomalous activities typically occur within the foreground region of the observed scene. Traditional methods designed for RGB images often struggle with background segmentation, especially under dynamic environmental conditions. Thus, utilizing depth information significantly simplifies and improves the reliability of foreground segmentation [9,13].

In our study, we adopt the depth-based pixel-wise background subtraction method proposed by Braham et al. [16]. This physically motivated model leverages the inherent properties of range images, such as their robustness to illumination changes and distinct foreground-background depth discontinuities, to achieve reliable segmentation. Here, we use the same parameter settings as in [9], which evaluates methods on the TIMo dataset under identical sensing conditions. The parameters we used are given in Table 1.

5. Experiments

5.1. Experiments setup

5.1.1. Dataset

We evaluate our method on two public datasets designed for 3D video anomaly detection: TIMo [44] and DAD [10]. Currently, there exists no dedicated dataset for anomaly detection in 3D point cloud videos. To address this limitation, we convert depth video datasets

into 3D point clouds via back-projection and evaluate our method accordingly. All baseline methods used for comparison are originally designed for depth video anomaly detection, as there is currently no publicly available method specifically developed for anomaly detection in 3D point cloud videos.

TIMo contains over 1500 sequences captured by an Azure Kinect depth camera, covering both normal and anomalous activities across tilted camera viewpoint. Invalid or missing depth values are removed. Following previous works [9,13], we use 909 normal sequences for training and 679 sequences for testing, among which 569 are anomalous.

DAD is a large-scale video dataset for monitoring anomalous driver behaviors in a simulated driving environment. In this work, we use only the front-view depth modality, which captures the driver's upper body and head movements. The training set includes recordings from 25 subjects, and the test set comprises 36 sequences from 6 unseen subjects, where 16 types of anomalous actions not present in the training set occur unpredictably.

5.1.2. Implementation details

HyPCV-Former is implemented in PyTorch and trained using distributed data parallel on 4 NVIDIA RTX 4090 GPUs with CUDA 12.2. Each input sequence consists of 3 consecutive frames, each downsampled to $N = 2048$ foreground points. The hyperbolic spatio-temporal transformation consists of 4 hyperbolic spatio-temporal transformer layers with 8 HMHA heads. The model is trained for 400 epochs using the AdamW optimizer with an initial learning rate of 1×10^{-4} and cosine annealing schedule. We set variable curvatures as trainable parameters, following the design principle introduced in [15].

5.1.3. Evaluation metrics

We use the area under the ROC curve (AUROC) to evaluate the performance of HyPCV-Former following the previous methods in video anomaly detection [10,44]. Anomaly scores are computed based on the Lorentzian intrinsic distance in hyperbolic space. On the TIMo dataset, we adopt a prediction-based strategy and measure the distance between predicted and actual point clouds in the final frame. On the DAD dataset, we employ a classification-based approach, where the anomaly score is derived from the distance to normal class prototypes in the embedding space.

To reduce local noise and improve temporal consistency, we apply a simple post-processing step based on a moving average. Specifically, for each frame, the final score is computed as the average of the current and previous $w - 1$ scores, where the window size w is set to 10. This temporal smoothing helps stabilize predictions and better capture sustained abnormal behaviors across consecutive frames.

5.2. Comparisons with state-of-the-art methods

To evaluate the effectiveness of our proposed HyPCV-Former, we compare it with several benchmarks on the TIMo dataset and DAD dataset, respectively. We evaluate our method on the same benchmarks as prior studies for a fair and consistent comparison [9,10]. As TIMo and DAD are originally depth video datasets, all baseline methods are implemented and evaluated on depth image sequences, as shown in Tables 2 and 3. Following the previous work [9], we adopt the categorization of anomalies in TIMo into three types, including aggressive behavior, medical issue, and left-behind objects. We report results for each category as well as the total performance, where all anomaly types are evaluated together as a unified test set. Note that the total score is not an average, but reflects the overall detection ability of each method across diverse anomaly types.

As shown in Tables 2 and 3, HyPCV-Former consistently outperforms prior methods. When using PointNet as the point cloud feature extractor, our method achieves the best results in both category-wise and overall evaluations. We also assess performance using PointMLP and DGCNN, observing that PointNet offers more stable and discriminative representations within our framework. We further examine the effect of the distance metric used to compute prediction errors. Specifically, ES refers to Euclidean space distance using MSE, while HS corresponds to hyperbolic space distance computed using the Lorentzian intrinsic distance. The results show that hyperbolic distance yields consistently better performance, underscoring the benefit of modeling in curved geometry for 3D point cloud video anomaly detection.

It is also worth noting that we do not include the results of F-MSE and W-MSE losses proposed in [9], as these loss functions are specifically designed for depth images and do not generalize to point cloud representations. For a fair comparison, all baseline methods and our variants are evaluated using L_2-based loss.

5.3. Additional results and analysis

5.3.1. Ablation study

To evaluate the effectiveness of different components of HyPCV-Former, we conduct ablation studies focusing on the choice of geometric space and loss function. The results are summarized in Table 4. We first analyze the effect of using different geometric spaces. When applying only Euclidean space with MSE loss, the AUROC reaches 75.4%. Switching to hyperbolic space while keeping MSE as the loss function improves the performance to 77.0%, demonstrating the benefit of hyperbolic representation in capturing complex spatio-temporal structures. We then replace MSE with the Lorentzian intrinsic distance. This results in a further increase to 77.3%, showing that using a loss function aligned with the hyperbolic geometry leads to better anomaly discrimination.

To validate the effectiveness of the key components in our proposed HMHA architecture, we conduct ablation studies on both the hyperbolic positional encoding and the HTC module, as shown in Table 5. The HyPCV-Former using our hyperbolic positional encoding achieves the highest AUC of 77.3%, outperforming both standard sinusoidal encoding and the variant without any positional encoding. The HTC-based HyPCV-Former consistently outperforms the standard variant, confirming that the curvature-aware transformation in hyperbolic space helps retain geometric consistency and provides a more expressive representation for spatiotemporal modeling.

5.3.2. Parameter analysis

We further investigate the influence of several key hyperparameters on the performance of HyPCV-Former, including the number of hyperbolic spatio-temporal transformer layers, feature channels, the input frame window size, and the number of sampling points. As shown in the left subfigure of Fig. 4, we conduct a grid search over different combinations of transformer layers and channel dimensions. We observe that the model achieves the best performance when using 4 layers and 256 channels, reaching an AUROC of 77.3%. The right subfigure of Fig. 4 illustrates the impact of varying the number of input frames from 2 to 10. We find that the model performs best when using 3 frames for temporal modeling. While increasing the frame length initially improves performance due to more temporal context, excessive length introduces noise and redundancy, leading to a decline in accuracy.

Table 6 presents a quantitative comparison of model performance and computational complexity across different numbers of input points per frame. As shown, increasing the number of points from 512 to 8192 results in a steady improvement in AUC, indicating enhanced anomaly detection capability. However, this performance gain comes at the cost of significantly increased model size and floating point operations (FLOP). To strike a balance between performance and computational cost, we select 2048 points per frame. This setting provides a favorable trade-off and aligns with previous method [13], ensuring consistency and comparability in experimental evaluation.

Table 2
Frame-level AUROC (%) performance comparison of TIMo dataset. For each column, the top-performing method is marked in **bold**.

Method	Point Cloud Extractor	Aggressive Behavior	Medical Issue	Left-Behind Objects	Total	
					ES	HS
CAE	–	–	–	–	66.4	–
ConvLSTM	–	–	–	–	62.8	–
R-CAE	–	76.8	48.0	66.6	66.4	–
P-CAE	–	79.3	59.9	73.4	71.4	–
R-ViT-AE	–	68.3	53.2	71.8	64.9	–
P-ViT-AE	–	68.9	53.6	72.6	65.1	–
P-ConvLSTM	–	62.2	50.9	64.9	62.8	–
HyPCV-Former	PointNet	**80.4**	**75.9**	**77.5**	**75.6**	**77.3**
	PointMLP	79.7	72.7	75.4	73.9	74.9
	DGCNN	80.0	72.7	76.1	74.4	75.5

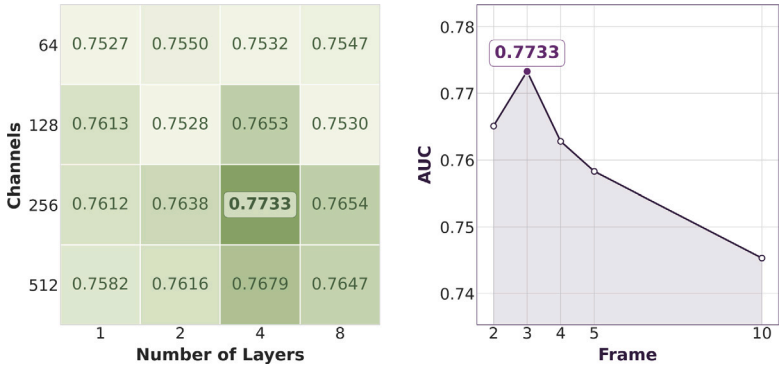


Fig. 4. Parameter analysis of HyPCV-Former on TIMo dataset. Left: AUROC heatmap across different hyperbolic spatio-temporal transformer depths and channel widths. Right: Effect of varying the number of input frames on AUROC.

Table 3
Frame-level AUROC (%) performance comparison of DAD dataset. For each column, the top-performing method is marked in **bold**.

Method	Point Cloud Extractor	Total	
		ES	HS
MobileNetV1 2.0×	–	90.18	–
MobileNetV2 1.0×	–	88.99	–
ShuffleNetV1 2.0×	–	88.69	–
ShuffleNetV2 2.0×	–	90.02	–
ResNet-18 (from scratch)	–	89.96	–
ResNet-18 (pre-trained)	–	90.20	–
ResNet-18 (post-processed)	–	90.20	–
HyPCV-Former	PointNet	**94.89**	**95.55**
	PointMLP	88.60	90.57
	DGCNN	93.92	95.08

Finally, we investigate the effect of the window size used in the post-processing stage. As shown in Fig. 5 , the performance improves when increasing w from small values, indicating that temporal smoothing effectively reduces local score noise. However, overly large window sizes lead to performance degradation due to excessive smoothing, which may blur short abnormal events. The best performance is achieved at $w = 10$, which provides a good balance between noise suppression and sensitivity to sustained anomalies.

5.4. Visualization and qualitative analysis

5.4.1. Feature discrimination visualization

To further demonstrate the discriminative capability of the learned features under different training settings, we visualize the representations of normal and anomalous frames using Isomap projection. Fig. 6 illustrates the feature distributions under three settings: (1) direct output of the point cloud extractor, (2) features learned with MSE loss in the Euclidean space, and (3) features learned with Lorentzian intrinsic distance in the hyperbolic space.

As shown in the left plot, when using the point cloud encoder alone, the extracted features for normal and anomalous samples show significant overlap, indicating limited separation in the latent space. Introducing MSE loss slightly improves the feature separability, yet the two classes still exhibit considerable mixing. In contrast, our method, fully operating in hyperbolic space with geometry-consistent distance, yields a much clearer separation. The boundary between normal and abnormal features becomes more distinct, validating that the hierarchical structures of temporal dynamics are better captured when learning is performed intrinsically in hyperbolic geometry.

5.4.2. Qualitative visualizations

We visualize anomaly scores from three representative videos, each from a different anomaly category in the TIMo dataset. Fig. 7(a) shows an aggressive behavior case, (b) is a medical issue, and (c) is luggage left behind. The qualitative results of three different video recordings in DAD dataset are visualized in Fig. 8. In each plot, the green dashed line is the raw anomaly score, and the blue solid line is the post-processed score. The red shaded areas represent ground truth anomalies. Post-processing clearly reduces noise and improves temporal consistency.

6. Conclusions

In this paper, we propose HyPCV-Former, a novel hyperbolic spatio-temporal transformer-based framework for anomaly detection in 3D point cloud videos. Our method leverages the rich spatio-temporal structure of point cloud sequences by first extracting per-frame features using point-based networks, and subsequently projecting them into a Lorentzian hyperbolic space. We introduce an HMHA mechanism that effectively captures temporal dependencies within the hyperbolic manifold, enabling more discriminative representations for anomaly prediction. Extensive experiments show that HyPCV-Former achieves state-of-the-art performance across multiple anomaly categories. In

Table 4
Ablation on the choice of space and loss function on TIMo dataset. The top-performing choice is marked in **bold**.

Space		Loss function		Params	GFLOPS	HyPCV-Former (%)
Euclidean	Hyperbolic	MSE	Lorentzian intrinsic distance			
✓		✓		5.37M	1.534	75.4
✓			✓	5.37M	1.534	77.0
	✓	✓		5.92M	0.765	77.1
	✓		✓	5.92M	0.765	**77.3**

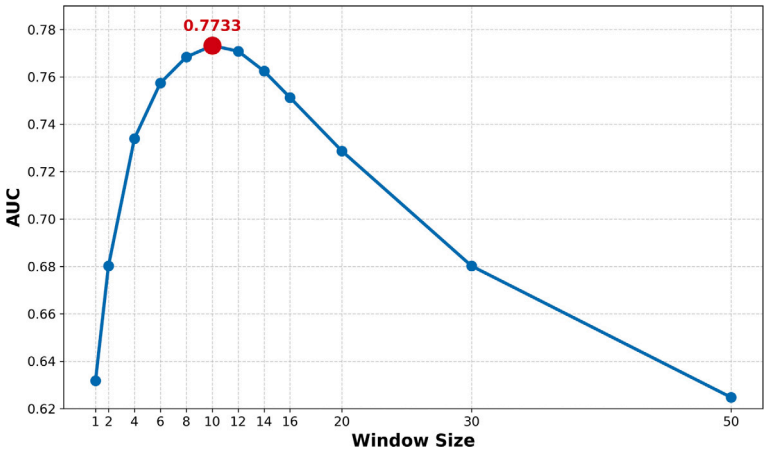


Fig. 5. Sensitivity analysis of the window size.

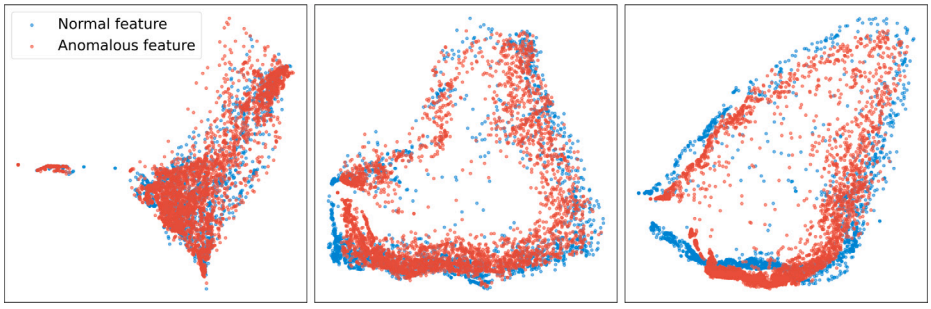
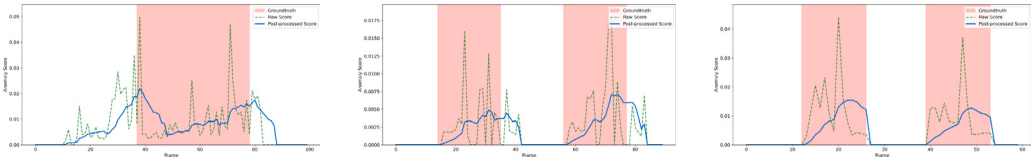


Fig. 6. Isomap projection of feature distributions on TIMo dataset. Left: Raw point cloud features. Middle: Features trained with MSE in Euclidean space. Right: Features trained with hyperbolic geometry and Lorentzian distance.



(a) Anomaly scores for sequence A0403

(b) Anomaly scores for sequence A0387

(c) Anomaly scores for sequence A0311

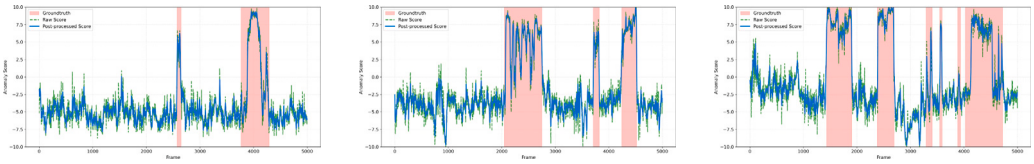
Fig. 7. Qualitative results for three anomaly categories in TIMo. (a) Anomaly scores for sequence A0403 (aggressive behavior), (b) Anomaly scores for sequence A0387 (medical issue), (c) Anomaly scores for sequence A0311 (left-behind object).

addition, ablation studies and qualitative visualizations further verify the advantages of hyperbolic modeling, the choice of loss functions, and our spatial–temporal design.

In future work, we plan to explore adaptive curvature learning to further enhance the geometric expressiveness of our model. We also aim to extend our framework to multi-view or multi-modal 3D video understanding tasks, such as action recognition and behavior forecasting.

CRediT authorship contribution statement

Jiaping Cao: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Kangkang Zhou:** Visualization, Validation, Software, Conceptualization. **Juan Du:** Writing – review & editing, Supervision, Resources, Project administration, Funding acquisition.



(a) Anomaly scores for sequence val05-rec6

(b) Anomaly scores for sequence val05-rec3

(c) Anomaly scores for sequence val04-rec6

Fig. 8. Qualitative results for three anomaly records in DAD. (a) Video recording #6 of subject #5, (b) Video recording #3 of subject #5, (c) Video recording #6 of subject #4.

Table 5
Ablation study on positional encoding and linear transformation on TIMo dataset. The top-performing choice is marked in **bold**.

Module	Type	AUC (%)
Positional Encoding	Hyperbolic-based	**77.3**
	Standard-based	77.0
	None	76.4
Linear Transformation	HTC-based	**77.3**
	Standard Linear-based	76.8

Table 6
Model performance and computational cost under different point numbers.

Points num	AUC (%)	Params	FLOP
512	74.2	4.741M	197.832M
1024	75.7	5.136M	386.576M
2048	77.3	5.925M	765.636M
4096	77.8	7.504M	1.524G
8192	78.1	10.662M	3.040G

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grant No. 72371219, Guangdong Project under Grant No. 2024TQ08A432, and Guangzhou Municipal Science and Technology Project under Grant No. 2025A04J5288.

Data availability

Data is public available.

References

[1] J. Leng, Z. Wu, M. Tan, Y. Liu, J. Gan, H. Chen, X. Gao, Beyond euclidean: Dual-space representation learning for weakly supervised video violence detection, 2024, arXiv preprint arXiv:2409.19252.

[2] H. Du, S. Zhang, B. Xie, G. Nan, J. Zhang, J. Xu, H. Liu, S. Leng, J. Liu, H. Fan, et al., Uncovering what why and how: A comprehensive benchmark for causation understanding of video anomaly, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 18793–18803.

[3] Y. Liu, D. Yang, Y. Wang, J. Liu, J. Liu, A. Boukerche, P. Sun, L. Song, Generalized video anomaly event detection: Systematic taxonomy and comparison of deep models, ACM Comput. Surv. 56 (7) (2024) 1–38.

[4] R. Nayak, U.C. Pati, S.K. Das, A comprehensive review on deep learning-based methods for video anomaly detection, Image Vis. Comput. 106 (2021) 104078.

[5] H. Karim, K. Doshi, Y. Yilmaz, Real-time weakly supervised video anomaly detection, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2024, pp. 6848–6856.

[6] Z. Yang, J. Liu, P. Wu, Text prompt with normality guidance for weakly supervised video anomaly detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 18899–18908.

[7] L. Zhang, K. Zhou, F. Lu, Z. Li, X. Shao, X.-D. Zhou, Y. Shi, ESMformer: Error-aware self-supervised transformer for multi-view 3D human pose estimation, Pattern Recognit. 158 (2025) 110955.

[8] L. Zhang, K. Zhou, F. Lu, X.-D. Zhou, Y. Shi, Deep semantic graph transformer for multi-view 3d human pose estimation, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 38, 2024, pp. 7205–7214.

[9] P. Schneider, J. Rambach, B. Mirbach, D. Stricker, Unsupervised anomaly detection from time-of-flight depth images, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 231–240.

[10] O. Kopuklu, J. Zheng, H. Xu, G. Rigoll, Driver anomaly detection: A dataset and contrastive learning approach, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2021, pp. 91–100.

[11] P. Bergmann, X. Jin, D. Sattlegger, C. Steger, The mvtec 3d-ad dataset for unsupervised 3d anomaly detection and localization, 2021, arXiv preprint arXiv:2112.09045.

[12] C. Zhang, M. Fiore, I. Murray, P. Patras, Cloudlstm: A recurrent neural model for spatiotemporal point-cloud stream forecasting, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, 2021, pp. 10851–10858.

[13] T. He, W. Wang, G. Zeng, Point cloud video anomaly detection based on point spatio-temporal auto-encoder, IEEE Sensors J. (2024).

[14] X. Peng, H. Wen, Y. Luo, X. Zhou, K. Yu, P. Yang, Z. Wu, Learning weakly supervised audio-visual violence detection in hyperbolic space, 2023, arXiv preprint arXiv:2305.18797.

[15] M. Yang, H. Verma, D.C. Zhang, J. Liu, I. King, R. Ying, Hypformer: Exploring efficient transformer fully in hyperbolic space, in: Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2024, pp. 3770–3781.

[16] M. Braham, A. Lejeune, M. Van Droogenbroeck, A physically motivated pixel-based model for background subtraction in 3D images, in: 2014 International Conference on 3D Imaging (IC3D), IEEE, 2014, pp. 1–8.

[17] A. Xiao, J. Huang, D. Guan, X. Zhang, S. Lu, L. Shao, Unsupervised point cloud representation learning with deep neural networks: A survey, IEEE Trans. Pattern Anal. Mach. Intell. 45 (9) (2023) 11321–11339.

[18] A. Hornung, K.M. Wurm, M. Bennewitz, C. Stachniss, W. Burgard, OctoMap: An efficient probabilistic 3D mapping framework based on octrees, Auton. Robots 34 (2013) 189–206.

[19] M. Nießner, M. Zollhöfer, S. Izadi, M. Stamminger, Real-time 3D reconstruction at scale using voxel hashing, ACM Trans. Graph. (ToG) 32 (6) (2013) 1–11.

[20] C.R. Qi, H. Su, K. Mo, L.J. Guibas, Pointnet: Deep learning on point sets for 3d classification and segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 652–660.

[21] C.R. Qi, L. Yi, H. Su, L.J. Guibas, Pointnet++: Deep hierarchical feature learning on point sets in a metric space, Adv. Neural Inf. Process. Syst. 30 (2017).

[22] Y. Wang, Y. Sun, Z. Liu, S.E. Sarma, M.M. Bronstein, J.M. Solomon, Dynamic graph cnn for learning on point clouds, ACM Trans. Graph. (Tog) 38 (5) (2019) 1–12.

[23] X. Ma, C. Qin, H. You, H. Ran, Y. Fu, Rethinking network design and local geometry in point cloud: A simple residual MLP framework, in: International Conference on Learning Representations, 2022, URL https://openreview.net/forum?id=3Pbra-_u76D.

[24] A. Montanaro, D. Valsesia, E. Magli, Rethinking the compositionality of point clouds through regularization in the hyperbolic space, Adv. Neural Inf. Process. Syst. 35 (2022) 33741–33753.

[25] O. Ganea, G. Bécigneul, T. Hofmann, Hyperbolic neural networks, Adv. Neural Inf. Process. Syst. 31 (2018).

[26] I. Chami, Z. Ying, C. Ré, J. Leskovec, Hyperbolic graph convolutional neural networks, Adv. Neural Inf. Process. Syst. 32 (2019).

[27] Q. Liu, M. Nickel, D. Kiela, Hyperbolic graph neural networks, Adv. Neural Inf. Process. Syst. 32 (2019).

[28] P. Mettes, M. Ghadimi Atigh, M. Keller-Ressel, J. Gu, S. Yeung, Hyperbolic deep learning in computer vision: A survey, Int. J. Comput. Vis. 132 (9) (2024) 3484–3508.

[29] V. Khrulkov, L. Mirvakhabova, E. Ustinova, I. Oseledets, V. Lempitsky, Hyperbolic image embeddings, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 6418–6428.
[30] A. Ermolov, L. Mirvakhabova, V. Khrulkov, N. Sebe, I. Oseledets, Hyperbolic vision transformers: Combining improvements in metric learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 7409–7419.
[31] R. Shimizu, Y. Wang, M. Kimura, Y. Hirakawa, T. Wada, Y. Saito, J. McAuley, A fashion item recommendation model in hyperbolic space, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 8377–8383.
[32] Y. Tan, C. Yang, X. Wei, C. Chen, L. Li, X. Zheng, Enhancing recommendation with automated tag taxonomy construction in hyperbolic space, in: 2022 IEEE 38th International Conference on Data Engineering, ICDE, IEEE, 2022, pp. 1180–1192.
[33] J. Dai, Y. Wu, Z. Gao, Y. Jia, A hyperbolic-to-hyperbolic graph convolutional network, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 154–163.
[34] L. Sun, Z. Zhang, J. Zhang, F. Wang, H. Peng, S. Su, P.S. Yu, Hyperbolic variational graph neural network for modeling dynamic graphs, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, 2021, pp. 4375–4383.
[35] H. Du, C. Liu, H. Liu, X. Ding, H. Huo, An efficient federated learning framework for graph learning in hyperbolic space, Knowl.-Based Syst. 289 (2024) 111438.
[36] C. Gulcehre, M. Denil, M. Malinowski, A. Razavi, R. Pascanu, K.M. Hermann, P. Battaglia, V. Bapst, D. Raposo, A. Santoro, et al., Hyperbolic attention networks, 2018, arXiv preprint arXiv:1805.09786.
[37] W. Chen, X. Han, Y. Lin, H. Zhao, Z. Liu, P. Li, M. Sun, J. Zhou, Fully hyperbolic neural networks, 2021, arXiv preprint arXiv:2105.14686.
[38] R. Shimizu, Y. Mukuta, T. Harada, Hyperbolic neural networks++, 2020, arXiv preprint arXiv:2006.08210.
[39] S. Cho, S. Cho, S. Park, H. Lee, H. Lee, M. Lee, Curve your attention: Mixed-curvature transformers for graph representation learning, 2023, arXiv preprint arXiv:2309.04082.
[40] M. Nickel, D. Kiela, Learning continuous hierarchies in the lorentz model of hyperbolic geometry, in: International Conference on Machine Learning, PMLR, 2018, pp. 3779–3788.
[41] W. Li, Z. Yang, W. Han, H. Man, X. Wang, X. Fan, Hyperbolic-constraint point cloud reconstruction from single RGB-d images, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 39, 2025, pp. 4959–4967.
[42] M. Law, R. Liao, J. Snell, R. Zemel, Lorentzian distance learning for hyperbolic representations, in: International Conference on Machine Learning, PMLR, 2019, pp. 3672–3681.
[43] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Adv. Neural Inf. Process. Syst. 30 (2017).
[44] P. Schneider, Y. Anisimov, R. Islam, B. Mirbach, J. Rambach, D. Stricker, F. Grandidier, Timo—a dataset for indoor building monitoring with a time-of-flight camera, Sensors 22 (11) (2022) 3992.