DualCrossAD: Multivariate Time Series Anomaly Detection with Homogeneous-Heterogeneous Dual-Cross Attention

1st Zeyu Zhang
College of Computer Science and Technology, Inner Mongolia Normal University
Hohhot 010020, China
Inner Mongolia Autonomous Region Key Laboratory for Ecological Environment Collaborative Intelligence
Hohhot 010020, China
Inner Mongolia Autonomous Region Industrial Technology Engineering Center for Intelligent and Data in Ecological Environment
Hohhot 010020, China
15248027304@163.com

2nd Gaizhi Guo
College of Computer Science and Technology, Inner Mongolia Normal University
Hohhot 010020, China
Inner Mongolia Autonomous Region Key Laboratory for Ecological Environment Collaborative Intelligence
Hohhot 010020, China
Inner Mongolia Autonomous Region Industrial Technology Engineering Center for Intelligent and Data in Ecological Environment
Hohhot 010020, China
ciecggz@imnu.edu.cn

***Abstract*—In recent years, research on multivariate time series anomaly detection has increasingly focused on leveraging Transformer models to address the challenges of long-range dependency modeling and complex variable interactions. However, conventional Transformer architectures struggle to simultaneously capture temporal patterns and variable-wise relationships within a uniffed representation space, which limits their expressive capacity. To overcome this, a dual-branch Transformer model was designed that generates temporally dominated and variable-dominated representations in parallel. Yet an effective fusion of these two homogeneous-but-heterogeneous representations remains challenging: single-path fusion often suffers from insufffcient interaction and static combination, while multi-path fusion can lead to coarse-grained interactions and isolated feature ffows.Therefore, we propose a dual cross-attention mechanism to achieve deep fusion of homogeneous-heterogeneous data, and introduce a symmetric KL divergence term to mitigate potential distributional inconsistencies during fusion.Experiments conducted on ffve benchmark datasets validate the effectiveness of the proposed method.**

***Index Terms*—Transformer, dual-branch, dual cross-attention, KL divergence**

I. Introduction

Multivariate time series anomaly detection (MTSAD) [1] is a fundamental task in data analytics and is widely applied in safety-critical domains such as industrial monitoring, financial risk management, and autonomous driving. With the proliferation of IoT and multi-sensor systems, modern infrastructures generate massive, high-dimensional, dynamic time series, posing challenges for accurate and efficient anomaly detection. Transformer architectures [2], including Informer [3], Autoformer [4], and FEDformer [5], have shown strong performance in capturing complex dependencies. However, their single-branch unified encoding limits joint modeling of temporal dynamics and inter-variable interactions.

To address this, recent studies explore dual-branch architectures that extract heterogeneous features via parallel specialized pathways, e.g., MTAD-GAT [6], MSCRED [7], and ENSO-Former [8]. Yet, fusing branch-specific representations remains a bottleneck, as existing methods rely on shallow concatenation or limited directional interaction, insufficient for capturing deep dependencies in "homogeneous–heterogeneous" multivariate data.To overcome these limitations, we propose DualCrossAD, a dual-branch model with dual cross-attention for effective feature fusion. Its main contributions are as follows:

- Analyze the representation bottleneck of single-branch Transformers in MTSAD and introduce a dual-branch architecture to decouple temporal and variable modeling.
- Design a dual cross-attention mechanism for deep, bidi-

This work was supported by the Central Government Guides Local Science and Technology Development Fund (Project No. 2024ZY0144) and the Natural Science Foundation of Inner Mongolia Autonomous Region (Project No. 2025MS06016).

rectional, and adaptive feature interaction, fully leveraging complementary information.

- Incorporate a symmetric KL divergence constraint to enforce distributional consistency of fused features, enhancing robustness and generalization.

II. Related Work

A. Multivariate Time Series Anomaly Detection Methods

The development of multivariate time series anomaly detection has progressed from traditional machine learning methods, such as Isolation Forest [9]and Support Vector Data Description [10], to deep learning frameworks. While classical methods rely on manual feature engineering and struggle to capture complex high-dimensional dependencies, deep models enable end-to-end feature learning. Single-branch architectures, e.g., PatchTST [11] and Anomaly Transformer [12], improve temporal and local feature modeling but suffer from representation conflicts when simultaneously capturing temporal dynamics and inter-variable relationships.

To address this, multi-branch designs have emerged, extracting temporal and variable features through specialized pathways for deeper decoupled representations. Examples include MTAD-GAT [6], which explicitly models variable relationships via a temporal–graph dual-branch structure, and DCdetector [13], which leverages dual-branch contrastive learning to enhance feature discriminability. Despite these advances, effective fusion of "homogeneous-but-heterogeneous" features remains challenging. Simple concatenation or shallow interaction cannot fully exploit complementary information, highlighting the need for bidirectional, fine-grained, and adaptive fusion mechanisms.

B. Feature Fusion and Model Optimization

Feature fusion is essential for enhancing multivariate time series anomaly detection. Early single-path methods, such as gated modulation [14], and convolutional local interactions [15], integrate multi-source features within a single pathway but struggle to capture deep cross-feature dependencies.

Multi-path fusion has emerged to address this limitation, processing homogeneous-but-heterogeneous features through dedicated pathways and structured interactions. For instance, SlowFast [16] uses cross-path connections for multi-level feature complementarity, while CFNet [17] employs cross-attention and multi-scale modules to enhance inter-modal representations. Nevertheless, coarse interaction granularity and distributional discrepancies among heterogeneous features remain challenges. Distribution consistency constraints, including statistical moment matching [18], adversarial learning [19], can alleviate these issues. In this work, symmetric KL divergence is applied to align dual-branch outputs, ensuring effective fusion of complementary information while improving consistency and stability.

III. The Proposed Method

Figure 1 illustrates DualCrossAD. The input time series is normalized and patched, then processed by parallel dual-branch Transformers to capture variable dependencies and temporal features. Dual cross-attention fuses these features, and the prediction layer outputs the anomaly scores.

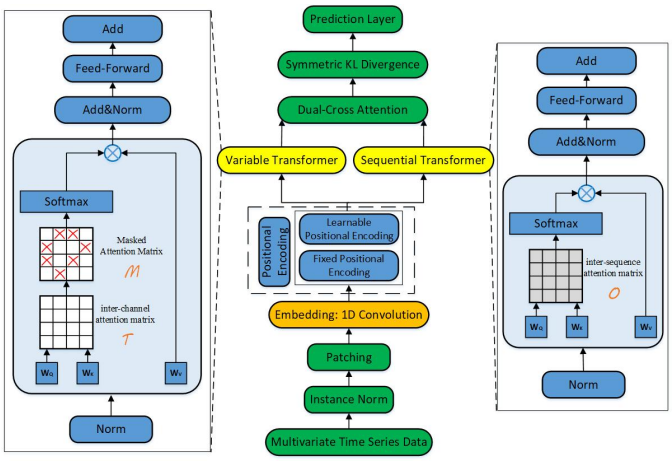

Fig. 1. Overall Pipeline of the DualCrossAD Model

A. Data Preprocessing, Embedding, and Positional Encoding

We denote the multivariate time series as $X \in \mathbb{R}^{N\times T}$, where N is the number of variables and T is the sequence length.The i-th variable sequence, starting at time index 1 with length L, is represented as $x_{1:L}^{(i)} = \left(x_1^{(i)}, \ldots, x_L^{(i)}\right), \quad i = 1, \ldots, N$.In multivariate time series anomaly detection, a "channel" corresponds directly to a "variable." To remove scale discrepancies across variables and improve training stability, each variable sequence $X^{(i)} \in \mathbb{R}^T$ is normalized independently using instance normalization, formulated as:

$$\hat{x}_t^{(i)} = \frac{x_t^{(i)} - \mu^{(i)}}{\sqrt{(\sigma^{(i)})^2 + \epsilon}}, \quad t = 1, 2, \ldots, L \tag{1}$$

Here, $\mu^{(i)}$ and $\sigma^{(i)}$ denote the mean and standard deviation of the i-th variable, respectively, and ϵ is a small constant for numerical stability. The normalized sequence $\hat{x}^{(i)}$ is then divided into M local patches using a predefined block length P and stride S:

$$M = \left\lfloor \frac{L - P}{S} \right\rfloor + 1 \tag{2}$$

To cover the end of the sequence, padding is applied at the tail before patching, yielding the patch representation $x_{\text{patch}}^{(i)} \in \mathbb{R}^{P\times M}$.Each patch is then embedded into a D-dimensional vector via a 1D convolution:

$$E^{(i)} = \text{Conv1d}\left(x_{\text{patch}}^{(i)}\right) \tag{3}$$

where the convolution kernel weights $W_{\text{c}} \in \mathbb{R}^{D\times P}$, producing an output $E^{(i)} \in \mathbb{R}^{M\times D}$. To preserve temporal order information, both fixed positional encoding P_{pos} and learnable positional encoding W_{pos} are added:

$$x_d^{(i)} = E^{(i)} + W_{\text{pos}} + P_{\text{pos}} \tag{4}$$

Finally, each variable is represented by $x_{\text{d}}^{(i)} \in \mathbb{R}^{M\times D}$, which serves as the input to the subsequent dual-branch Transformer.

B. Dual-Branch Attention Mechanism

After obtaining the embeddings and positional encodings, a feature was acquired set corresponding to the N variables,$\{x_d^{(1)}, x_d^{(2)}, \ldots, x_d^{(N)}\}$.These features are stacked along the variable dimension and reshaped into a three-dimensional tensor $X_{\mathrm{d}} \in \mathbb{R}^{N\times M\times D}$. After adding the batch dimension, the input tensor $X \in \mathbb{R}^{B\times N\times M\times D}$ is reshaped separately into $X_{\mathrm{seq}} \in \mathbb{R}^{(B\times N)\times M\times D}$ and $X_{\mathrm{var}} \in \mathbb{R}^{(B\times M)\times N\times D}$, which are fed into the sequence Transformer and the variable Transformer, respectively.To stabilize the training process and accelerate convergence, layer normalization is applied to X_{seq} and X_{var}:

$$Z_{\mathrm{seq}}^{(i)\prime} = \mathrm{LayerNorm}\big(Z_{\mathrm{seq}}^{(i)}\big) = \frac{Z_{\mathrm{seq}}^{(i)} - \mu^{(i)}}{\sqrt{(\sigma^{(i)})^2 + \epsilon}} \tag{5}$$

where $Z_{\mathrm{seq}}^{(i)} \in \mathbb{R}^D$ denotes the i-th feature vector in X_{seq} . After independently normalizing all feature vectors, we obtain $X_{\mathrm{seq}}^{'} \in \mathbb{R}^{(B\times N)\times M\times D}$ and $X_{\mathrm{var}}^{'} \in \mathbb{R}^{(B\times M)\times N\times D}$.

In the variable Transformer branch, attention is computed at the block level.Each block $P^{(j)} \in \mathbb{R}^{N\times D}$ is extracted from the corresponding sample–time slice of $X_{\mathrm{var}}^{'}$.Linear projections are then applied to generate Q,K and V:

$$Q_h^{(j)} = W_h^Q P^{(j)}, K_h^{(j)} = W_h^K P^{(j)}, V_h^{(j)} = W_h^V P^{(j)} \tag{6}$$

To enhance the model's ability to selectively capture inter-variable dependencies, a learnable binary mask $M^{(j)} \in \mathbb{R}^{N\times N}$ is introduced.It is generated from a probability matrix $D^{(j)} = \sigma(\mathrm{Linear}(P^{(j)}))$ using Gumbel-Softmax [20] resampling.The masked attention computation is formulated as:

$$C_h^{(j)} = Q_h^{(j)}(K_h^{(j)})^T \tag{7}$$

$$S_h^{(j)} = C_h^{(j)} \odot M^{(j)} + (1 - M^{(j)}) \odot (-\infty) \tag{8}$$

$$T_h^{(j)} = \mathrm{Softmax}\Big(\frac{S_h^{(j)}}{\sqrt{d_k}}\Big)V_h^{(j)} \tag{9}$$

In the sequence Transformer branch, attention is computed per channel. For the i-th variable, linear projections are applied to $Z_{\mathrm{seq}}^{(i)'}$, and standard scaled dot-product attention is performed:

$$O_h^{(i)\prime} = \mathrm{Softmax}\Big(\frac{Q_h^{(i)\prime}(K_h^{(i)\prime})^T}{\sqrt{d_k}}\Big)V_h^{(i)\prime} \tag{10}$$

Outputs from all attention heads are concatenated and projected back to the original dimension:

$$S_1^{(i)} = O^{(i)\prime}W_o \tag{11}$$

Then, layer normalization, residual connections, and a feed-forward network are applied:

$$\hat{S}_1^{(i)} = \mathrm{LayerNorm}(S_1^{(i)}) \tag{12}$$

$$\bar{S}_1^{(i)} = \hat{S}_1^{(i)} + S_1^{(i)} \tag{13}$$

$$S_1^{\prime(i)} = \mathrm{FFN}(\bar{S}_1^{(i)}) + \bar{S}_1^{(i)} \tag{14}$$

Finally, the outputs of the N variables from all samples are stacked to obtain the final output of the sequence Transformer branch,$S_1^{'} \in \mathbb{R}^{(B\times N)\times M\times D}$.The output of the variable Transformer branch is $S_2^{'} \in \mathbb{R}^{(B\times M)\times N\times D}$.

C. Dual Cross-Attention

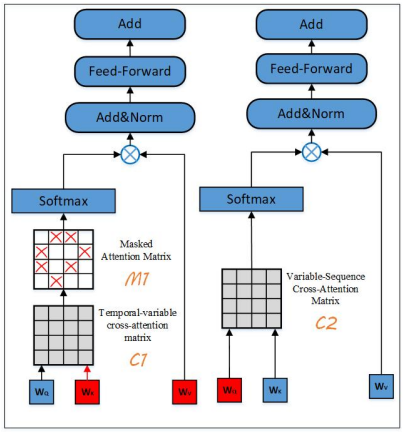

Fig. 2. Dual-Cross Attention Workflow

The structure of the dual cross-attention mechanism is illustrated in Fig. 2, in which the temporal and variable branches are designed symmetrically. Each branch employs its own set of learnable weight matrices to project the input features into the Q, K and V spaces. In the diagram, the weight matrices of the temporal branch are highlighted in red, while those of the variable branch are shown in blue. Since the fundamental attention mechanism has been introduced earlier, this section focuses on the cross-attention module centered on the variable Transformer, which enhances the modeling of complex inter-variable dependencies through a learnable masking matrix.

In this pathway, the output of the temporal branch $S_1^{'} \in \mathbb{R}^{(B\times N)\times M\times D}$, serves as the source of Q, whereas the output of the variable branch $S_2^{'} \in \mathbb{R}^{(B\times M)\times N\times D}$ provides the K and V. To align the features of the two branches,$S_1^{'}$ is reshaped via $S_1 = \mathrm{Reshape}(S_1^{\prime})$,yielding $S_1 \in \mathbb{R}^{(B\times M)\times N\times D}$.Subsequently,$S_1$ and $S_2^{'}$ are linearly projected to generate the Q, K and V matrices for the h-th attention head$(h = 1, 2, \ldots, H)$:

$$Q_h = W_h^Q S_1, K_h = W_h^K S_2^{'}, V_h = W_h^V S_1^{'} \tag{15}$$

The attention computation proceeds as follows:

$$\mathbf{C1}_h = Q_h K_h^{\top} \tag{16}$$

where $C1_{\mathrm{h}} \in \mathbb{R}^{(B\times M)\times N\times N}$ captures the interaction strength between temporal and variable information.To enhance the modeling of inter-variable relations, a learnable mask matrix $M_1 \in \mathbb{R}^{N\times N}$(as introduced earlier and shown in Fig. 2) is incorporated into the attention computation:

$$S_h = \mathbf{C1}_h \odot M_1 + (1 - M_1) \odot (-\infty) \tag{17}$$

Here,$S_{\text{h}} \in \mathbb{R}^{(B\times M)\times N\times N}$.The masked attention scores are then fed into the scaled dot-product attention:

$$\mathbf{A1}_h = \text{Softmax}\Big(\frac{S_h}{\sqrt{d_k}}\Big)V_h \tag{18}$$

Outputs from all attention heads are concatenated along the feature dimension and projected by $W_{\text{O}} \in \mathbb{R}^{(D\times H)\times D}$ to obtain the output of this branch, denoted as $A1^{'}$.

After residual connection, layer normalization, and the feed-forward network, the variable-dominant enhanced representation $\tilde{A}_1 \in \mathbb{R}^{(B\times M)\times N\times D}$ is obtained.Symmetrically, in the other pathway,$S_2^{'}$ serves as the source of queries, while $S_1^{'}$ provides the keys and values.Applying the sequence-wise attention mechanism described previously, followed by the same post-processing modules, yields the temporal-dominant enhanced representation $\tilde{A}_2 \in \mathbb{R}^{(B\times N)\times M\times D}$.

D. Symmetric KL Divergence

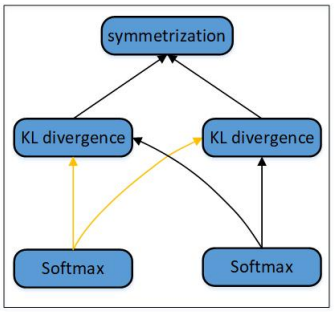

Fig. 3. Symmetric KL Divergence

To quantify the distributional consistency between the dual-branch outputs, the outputs of the dual-cross attention,$\tilde{A}_1$ and $\tilde{A}_2$,are first converted into probability distributions P and Q using the Softmax function:

$$P = \text{Softmax}(\tilde{A}_1), \quad Q = \text{Softmax}(\tilde{A}_2) \tag{19}$$

The bidirectional KL divergence is then computed as:

$$\text{KL}(P\|Q) = \sum_{i=1}^{D} P_i \log Q_i \tag{20}$$

$$\text{KL}(Q\|P) = \sum_{i=1}^{D} Q_i \log P_i \tag{21}$$

Finally, the symmetric KL divergence loss is obtained by taking the arithmetic mean of the two terms, serving as the consistency regularization term L_{con}:

$$L_{\text{con}} = \frac{1}{2}\left[\text{KL}(P\|Q) + \text{KL}(Q\|P)\right] \tag{22}$$

This loss is used as a regularizer alongside the main loss to jointly optimize the model.

E. Output Representation and Multi-Objective Loss Function

After obtaining the enhanced feature representations $\tilde{A}_1 \in \mathbb{R}^{(B\times M)\times N\times D}$ and $\tilde{A}_2 \in \mathbb{R}^{(B\times N)\times M\times D}$ via the dual cross-attention mechanism and symmetric KL-divergence constraint, the model first integrates the complementary information from both branches into a unified tensor $A^{'} \in \mathbb{R}^{B\times N\times 2D\times M}$ through dimension transformation, concatenation, and reshaping. The final prediction is then obtained through a linear projection layer prediction $= \text{Linear}(\text{Flatten}(A')) \in \mathbb{R}^{B\times N\times \text{target}}$.To optimize model performance, a multi-level loss function framework is constructed. First, to guide the mask generator in learning effective variable dependencies, a dual-supervision mechanism is designed, combining a clustering loss and a regularization term. The clustering loss:

$$\text{ClusteringLoss} = -\frac{1}{N}\sum_{n=1}^{N} \log \frac{\sum_{m=1}^{N} \exp\left(\frac{S_{k,m}^{i}}{\tau}\right)}{\sum_{l=1}^{N} \exp\left(\frac{T_{k,l}^{i}}{\tau}\right)} \tag{23}$$

preserves consistency between the attention distribution after masking and the original semantics, while the regularization loss:

$$\text{RegularLoss} = \frac{1}{N}\left\|I - M^{i}\right\|_F \tag{24}$$

suppresses deviations of the mask matrix from the identity matrix. The overall mask generator loss is then formulated as a weighted combination:

$$L_{\text{Mask}} = \text{ClusteringLoss} + k \cdot \text{RegularLoss} \tag{25}$$

Second, to improve robustness under the presence of outliers, the main prediction task is optimized using the Huber loss:

$$L_{\text{Huber}} = \frac{1}{N}\sum_{i=1}^{N}\sum_{t=L+1}^{L+T} l_\delta\left(\text{prediction}_t^{(i)} - x_t^{(i)}\right) \tag{26}$$

where l_δ is the Huber loss function and δ is a threshold parameter controlling its smoothness.

Finally, the symmetric KL-divergence-based distribution consistency loss L_{con} (defined in Section 2.5) is introduced to explicitly align the output distributions of the dual branches. The overall optimization objective of the model is defined as the weighted sum of these three loss terms:

$$L_{\text{loss}} = L_{\text{Huber}} + \lambda_1 \cdot L_{\text{Mask}} + \lambda_2 \cdot L_{\text{con}} \tag{27}$$

where λ_1 and λ_2 are hyperparameters. This multi-objective loss framework ensures collaborative optimization of prediction accuracy, structural learning, and distributional consistency.

IV. Experimental Results

A. Datasets and Evaluation Metrics

The experiments selected five publicly available benchmark datasets covering five domains: passenger flow, finance, water treatment, machinery, and transportation. As shown in Table 1, these datasets exhibit significant differences in the number of variables, sequence lengths, and anomaly rates, providing a

comprehensive benchmark for evaluating the model's performance under varying complexities and scenarios. Precision, Recall, F1-score, and Accuracy (Acc) are used for quantitative analysis. This multi-dimensional evaluation framework comprehensively assesses the model's precision, completeness, and overall discriminative ability, avoiding the limitations of relying on a single metric.

TABLE I
STATISTICS OF THE MULTIVARIATE TIME-SERIES DATASETS.

Dataset	Domain	Variables	AR(%)	ASL	ATL
CalIt2	Traffic	2	4.09	5,040	2,520
Creditcard	Finance	29	0.17	284,807	142,404
GECCO	Water	9	1.25	138,524	69,261
Genesis	Manufacturing	18	0.31	16,220	12,616
NYC	Transport	3	0.57	17,520	4,416

To evaluate our model, el, it is compared with nine representative baselines from 2023–2024:

Transformer variants:iTransformer (iTrans) [21] uses an inverted architecture for long-term forecasting;Anomaly Transformer (ATrans) bt16 detects anomalies via an association discrepancy mechanism;PatchTST (Patch) [11] employs blockwise input and channel-independent design.

Dedicated anomaly detectors:DCdetector (DC) [13] uses dual-branch contrastive learning;DualTF [22] adopts a dual-branch design for heterogeneous temporal features.

Temporal convolution and linear models:ModernTCN (Modern) [23] integrates advanced temporal convolution;TimesNet (TsNet) [24] leverages 2D temporal transformations for periodic patterns;DLinear (DLin) and NLinear (NLin) [25] showcase the effectiveness of linear models in time series.

B. Experimental Results

This section presents a comprehensive evaluation of DualCrossAD against mainstream baseline models on five representative datasets, using four core metrics:F1, Precision(P), Recall(R), and Accuracy(Acc).To ensure comparability and objectivity of the results. All baseline experimental data are taken from the original publications to guarantee reliability. Tables 2 and 3 summarize the results, where boldface indicates the best performance and underlined values denote the second-best.

Tables 2 and 3 show that DualCrossAD outperforms baselines relying on single-sequence modeling (e.g., Anomaly Transformer, PatchTST) or local linear decomposition (e.g., TimesNet, DLinear) across most datasets and metrics. Its dual-branch design, separating temporal and variable dependencies, captures complex multivariate patterns more effectively. Notably, in low-anomaly (Genesis) or structurally complex (NYC) scenarios, DualCrossAD consistently detects anomalies while other models degrade. These results demonstrate that the dual-cross attention mechanism enables deep, robust cross-branch feature fusion, enhancing generalization and robustness. Overall, the experiments validate the effectiveness of DualCrossAD in disentangling features and improving anomaly detection under challenging conditions.

TABLE II
ACC AND F1 SCORES OF DIFFERENT MODELS ON MULTIPLE DATASETS.

Model	CalIt2		Creditcard		GECCO		Genesis		NYC	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1
Modern	0.889	0.114	0.981	0.107	0.984	0.504	0.965	0.027	0.972	0.000
iTrans	0.930	0.161	0.951	0.051	0.981	0.189	0.986	0.095	0.797	0.061
DualTF	0.910	0.103	0.688	0.006	0.612	0.018	0.970	0.116	0.976	0.019
ATrans	0.944	0.090	0.847	0.003	0.983	0.010	0.931	0.100	**0.977**	0.020
DC	0.841	0.056	0.716	0.003	0.979	0.008	0.991	0.018	0.977	0.019
TsNet	0.946	0.106	0.981	0.107	0.984	0.516	0.991	0.149	0.977	0.000
Patch	0.889	0.141	0.980	0.107	0.989	0.524	0.987	0.109	0.976	0.000
DLin	0.887	0.128	0.981	0.105	0.981	0.466	0.987	0.079	0.976	0.000
NLin	0.844	0.109	0.980	0.107	0.987	0.418	0.972	0.022	0.977	0.000
DCross	**0.948**	**0.177**	**0.994**	**0.159**	**0.990**	**0.585**	**0.992**	**0.160**	0.955	**0.153**

TABLE III
P AND R OF DIFFERENT MODELS ON MULTIPLE DATASETS.

Model	CalIt2		Creditcard		GECCO		Genesis		NYC	
	P	R	P	R	P	R	P	R	P	R
Modern	0.074	0.243	0.058	0.740	0.373	0.779	0.015	0.120	0.000	0.000
iTrans	0.124	0.230	0.026	**0.843**	0.173	0.207	0.065	0.180	0.034	**0.293**
DualTF	0.073	0.176	0.003	0.596	0.009	0.340	0.066	0.500	0.111	0.010
ATrans	0.025	0.095	0.001	0.139	0.012	0.008	0.053	**0.960**	**0.333**	0.010
DC	0.034	0.162	0.001	0.238	0.008	0.008	0.016	0.020	0.250	0.010
TsNet	0.104	0.108	0.058	0.740	0.379	**0.804**	0.119	0.200	0.000	0.000
Patch	0.091	0.311	0.058	0.749	0.475	0.585	0.075	0.200	0.000	0.000
DLin	0.083	0.284	0.057	0.726	0.332	0.781	0.055	0.140	0.000	0.000
NLin	0.066	**0.324**	0.057	0.744	0.384	0.460	0.013	0.080	0.000	0.000
DCross	**0.166**	0.189	**0.100**	0.376	**0.532**	0.650	**0.145**	0.180	0.123	0.202

C. Ablation Study

To systematically evaluate the contribution of each core component in DualCrossAD, we conduct a set of ablation studies that decompose the model into its key design elements. Figure 4 summarizes the ablation results across three critical aspects: (a) dual-branch architecture, comparing the performance of the variable branch, the temporal branch, and the complete dual-branch design; (b) dual cross-attention mechanism, analyzing the effects of variable-wise cross attention, temporal cross attention, and the full dual cross-attention configuration; and (c) symmetric KL divergence regularization, which examines the impact of this constraint. Overall, the results demonstrate that the complete model consistently achieves the best or near-best F1 scores across all datasets, providing preliminary evidence for the effectiveness and necessity of each proposed component. We next discuss these results in detail.

1) Effectiveness of the Dual-Branch Framework: To validate the effectiveness of the dual-branch framework,this section designs two single-branch models as baselines for ablation studies: the Variable model (retaining only the variable Transformer branch) and the Temporal model (retaining only the temporal Transformer branch).

Table 4 shows that DualCrossAD consistently outperforms the single-branch variants across all metrics. While the Vari-

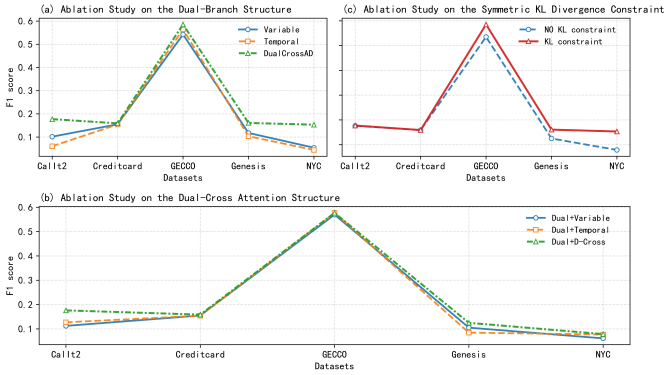

Fig. 4. Ablation Results of the Dual-Branch Component (F1 Score)

TABLE IV
ABLATION STUDY OF THE DUAL-BRANCH STRUCTURE.

Model	Variable		Temporal		DCross	
	Acc	F1	Acc	F1	Acc	F1
CalIt2	0.9369	0.1016	0.9380	0.0602	**0.9484**	**0.1772**
Creditcard	0.9938	0.1548	0.9938	0.1543	**0.9940**	**0.1590**
GECCO	0.9293	0.5422	0.9898	0.5641	**0.9902**	**0.5853**
Genesis	0.9916	0.1176	0.9917	0.1034	**0.9925**	**0.1607**
NYC	0.9445	0.0540	0.9413	0.0442	**0.9545**	**0.1532**

Model	Variable		Temporal		DCross	
	P	R	P	R	P	R
CalIt2	0.0873	0.1216	0.0543	0.0675	**0.1666**	**0.1891**
Creditcard	0.0987	0.3587	0.0982	0.3587	**0.1007**	**0.3766**
GECCO	0.4965	0.5972	0.5171	0.6205	**0.5319**	**0.6506**
Genesis	0.1014	0.1400	0.0909	0.1200	**0.1451**	**0.1800**
NYC	0.0437	0.0707	0.0348	0.0606	**0.1234**	**0.2020**

able branch sometimes achieves high F1, its Recall is low, and the Temporal branch suffers from low Precision. In contrast, DualCrossAD improves F1 by over 50% on complex datasets (CalIt2, NYC) and achieves the highest Recall and F1 on low-anomaly datasets (Genesis), demonstrating the effectiveness and robustness of the dual-branch design.

2) Effectiveness of the Dual Cross-Attention Structure: To evaluate the effectiveness of the dual cross-attention structure in information fusion, this section constructs three variant models for ablation experiments: the Dual+Variable model (retaining only cross-attention with variables as queries), the Dual+Temporal model (retaining only cross-attention with temporal sequences as queries), and the Dual+D-Cross model (using the full dual cross-attention structure).

Table 5 shows that Dual+D-Cross outperforms the single-direction variants on most metrics, highlighting the importance of dual cross-attention for deep cross-branch fusion. Single-direction models struggle on F1 and Recall (e.g., Dual+Variable on Genesis, Dual+Temporal on NYC), while Dual+D-Cross consistently improves both metrics across all datasets, particularly on CalIt2 and Creditcard, and maintains strong performance on low-anomaly datasets, demonstrating its sensitivity and robustness for sparse anomaly detection.

TABLE V
ABLATION STUDY OF DUAL-CROSS ATTENTION STRUCTURE.

Model	Dual+Variable		Dual+Temporal		Dual+ D-Cross	
	P	R	P	R	P	R
CalIt2	0.1046	0.1216	0.1111	0.1486	**0.1647**	**0.1891**
Creditcard	0.0985	0.3632	0.0990	0.3632	**0.1005**	**0.3766**
GECCO	0.5267	0.6205	**0.5284**	**0.6356**	0.5270	0.6305
Genesis	0.0937	0.1200	0.0735	0.1000	**0.1129**	**0.1400**
NYC	0.0538	0.0707	0.0666	0.0909	**0.0692**	**0.0910**

Model	Dual + Variable		Dual + Temporal		Dual + D-Cross	
	Acc	F1	Acc	F1	Acc	F1
CalIt2	0.9436	0.1125	0.9400	0.1271	**0.9480**	**0.1761**
Creditcard	0.9937	0.1550	**0.9938**	0.1543	0.9937	**0.1587**
GECCO	0.9901	0.5698	**0.9902**	**0.5771**	0.9901	0.5770
Genesis	0.9919	0.1052	0.9914	0.0847	**0.9922**	**0.1250**
NYC	0.9513	0.0611	0.9510	0.0769	**0.9522**	**0.0786**

3) Effectiveness of the Symmetric KL Divergence: To validate the effectiveness of symmetric KL divergence as a loss component, this section conducts an ablation study based on the constructed Dual+D-Cross model, comparing the model's performance with and without the inclusion of symmetric KL divergence.

TABLE VI
ABLATION STUDY OF SYMMETRIC KL DIVERGENCE.

Dataset	Dual+D-Cross		DCross		Dual+D-Cross		DCross	
	P	R	P	R	Acc	F1	Acc	F1
CalIt2	0.1647	0.1891	**0.1666**	**0.1891**	0.9480	0.1772	**0.9484**	**0.1772**
Creditcard	0.1005	0.3766	**0.1007**	**0.3766**	0.9937	0.1587	**0.9940**	**0.1590**
GECCO	0.5270	0.6305	**0.5319**	**0.6506**	0.9901	0.5770	**0.9902**	**0.5853**
Genesis	0.1129	0.1400	**0.1451**	**0.1800**	0.9922	0.1250	**0.9925**	**0.1607**
NYC	0.0692	0.0910	**0.1234**	**0.2020**	0.9522	0.0786	**0.9545**	**0.1532**

Table 6 shows that DualCrossAD outperforms Dual+D-Cross without symmetric KL divergence across all metrics, confirming the importance of the distributional consistency constraint. On challenging or low-anomaly datasets (GECCO, Genesis, NYC), the absence of KL reduces F1 and Recall (e.g., GECCO Recall drops from 0.6506 to 0.5849). DualCrossAD consistently improves Precision and Recall, with NYC F1 increasing over 95%, demonstrating that symmetric KL effectively aligns dual-branch features, mitigates conflicts, and enhances anomaly detection and generalization.

CONCLUSION

In this paper, DualCrossAD is proposed, a dual-branch model with dual cross-attention for multivariate time series anomaly detection. The model decouples temporal and variable dependencies via separate Transformer branches and achieves bidirectional, multi-level feature interaction through cross-attention. A symmetric KL divergence constraint is introduced to enforce distributional consistency, enhancing robustness and generalization. Experiments on multiple benchmark datasets demonstrate that DualCrossAD consistently outperforms existing methods, validating its architecture and fusion strategy. While the model achieves high detection accuracy, its decision process remains less interpretable. Future work

will focus on integrating explainability techniques to provide more transparent and actionable anomaly insights, improving reliability in real-world applications.

REFERENCES

[1] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," ACM Comput. Surv., vol. 41, no. 3, pp. 1–58, 2009.

[2] A. Vaswani, N. Shazeer, N. Parmar, et al., "Attention is all you need," Adv. Neural Inf. Process. Syst., vol. 30, 2017.

[3] H. Zhou, S. Zhang, J. Peng, et al., "Informer: Beyond efficient transformer for long sequence time-series forecasting," in Proc. AAAI Conf. Artif. Intell., vol. 35, no. 12, 2021, pp. 11106–11115.

[4] H. Wu, J. Xu, J. Wang, et al., "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," Adv. Neural Inf. Process. Syst., vol. 34, pp. 22419–22430, 2021.

[5] T. Zhou, Z. Ma, Q. Wen, et al., "Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting," in Proc. Int. Conf. Mach. Learn., 2022, pp. 27268–27286.

[6] H. Zhao, Y. Wang, J. Duan, et al., "Multivariate time-series anomaly detection via graph attention network," in Proc. IEEE Int. Conf. Data Mining, 2020, pp. 841–850.

[7] C. Zhang, D. Song, Y. Chen, et al., "A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data," in Proc. AAAI Conf. Artif. Intell., vol. 33, no. 01, 2019, pp. 1409–1416.

[8] W. Fang and Y. Sha, "ENSO-Former: Spatiotemporal fusion network based on multivariate and dual-branch transformer for ENSO prediction," Clim. Dyn., vol. 63, no. 2, p. 131, 2025.

[9] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation Forest," in Proc. IEEE Int. Conf. Data Mining, 2008, pp. 413–422.

[10] B. Schölkopf, R. C. Williamson, A. Smola, et al., "Support vector method for novelty detection," Adv. Neural Inf. Process. Syst., vol. 12, 1999.

[11] Y. Nie, N. H. Nguyen, P. Sinthong, et al., "A Time Series is Worth 64 Words: Long-term Forecasting with Transformers," arXiv preprint arXiv:2211.14730, 2022.

[12] J. Xu, H. Wu, J. Wang, et al., "Anomaly Transformer: Time Series Anomaly Detection with Association Discrepancy," arXiv preprint arXiv:2110.02642, 2021.

[13] Y. Yang, C. Zhang, T. Zhou, et al., "DCdetector: Dual Attention Contrastive Representation Learning for Time Series Anomaly Detection," in Proc. ACM SIGKDD Conf. Knowl. Discov. Data Min., 2023, pp. 3033–3045.

[14] Z. Yu, J. Yu, J. Fan, et al., "Multi-Modal Factorized Bilinear Pooling with Co-Attention Learning for Visual Question Answering," in Proc. IEEE Int. Conf. Comput. Vis., 2017, pp. 1821–1830.

[15] Y. Kim, "Convolutional Neural Networks for Sentence Classification," arXiv preprint arXiv:1408.5882, 2014.

[16] C. Feichtenhofer, H. Fan, J. Malik, et al., "SlowFast Networks for Video Recognition," in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 6202–6211.

[17] Y. Cheng, Y. Zheng, and J. Wang, "CFNet: Automatic Multi-Modal Brain Tumor Segmentation through Hierarchical Coarse-to-Fine Fusion and Feature Communication," Biomed. Signal Process. Control, vol. 99, p. 106876, 2025.

[18] Z. Peng, X. Wang, S. Chen, et al., "Federated Learning for Diffusion Models," IEEE Trans. Cogn. Commun. Netw., 2025.

[19] Y. Ganin and V. Lempitsky, "Unsupervised Domain Adaptation by Backpropagation," in Proc. Int. Conf. Mach. Learn., 2015, pp. 1180–1189.

[20] E. Jang, S. Gu, and B. Poole, "Categorical Reparameterization with Gumbel-Softmax," arXiv preprint arXiv:1611.01144, 2016.

[21] Y. Liu, T. Hu, H. Zhang, et al., "iTransformer: Inverted Transformers are Effective for Time Series Forecasting," arXiv preprint arXiv:2310.06625, 2023.

[22] Y. Nam, S. Yoon, Y. Shin, et al., "Breaking the Time-Frequency Granularity Discrepancy in Time-Series Anomaly Detection," in Proc. ACM Web Conf., 2024, pp. 4204–4215.

[23] D. Luo and X. Wang, "ModernTCN: A Modern Pure Convolution Structure for General Time Series Analysis," in Proc. Int. Conf. Learn. Represent., 2024, pp. 1–43.

[24] H. Wu, T. Hu, Y. Liu, et al., "TimesNet: Temporal 2D-Variation Modeling for General Time Series Analysis," arXiv preprint arXiv:2210.02186, 2022.

[25] A. Zeng, M. Chen, L. Zhang, et al., "Are Transformers Effective for Time Series Forecasting?" in Proc. AAAI Conf. Artif. Intell., vol. 37, no. 9, 2023, pp. 11121–11128.

[26] X. Wu, X. Qiu, Z. Li, et al., "Catch: Channel-aware Multivariate Time Series Anomaly Detection via Frequency Patching," arXiv preprint arXiv:2410.12261, 2024.