

TRUSS: Template-guided Residual Uncertainty State-Space Model

Euichan Moon

Department of Artificial Intelligence
The Catholic University of Korea
Bucheon, Republic of Korea
echan02@catholic.ac.kr

Youjin Shin*

Department of Data Science & Artificial Intelligence
The Catholic University of Korea
Bucheon, Republic of Korea
yj.shinn@catholic.ac.kr

Abstract

Multivariate time series anomaly detection is a challenging task due to noise, uncertainty, and complex correlations among multiple features in real-world time series data. Existing reconstruction-based and prediction-based methods often rely on deterministic outputs, which limits their ability to model noisy environments and diverse normal patterns. To address these limitations, this paper proposes the Template-guided Residual Uncertainty State-Space Model (TRUSS) for multivariate time series anomaly detection.

The proposed model reconstructs each time series patch not as a single deterministic value, but as a probability distribution, thereby explicitly modeling reconstruction uncertainty. In addition, TRUSS extracts trend information from training patches and clusters it to initialize a Template Bank, enabling the reconstruction process to leverage representative temporal patterns that repeatedly appear in normal data. By integrating Recurrent State-Space Model (RSSM)-based latent dynamics, template selection, residual correction, and probabilistic decoding, the model jointly captures temporal dependencies, feature-wise trend structures, and uncertainty in multivariate time series.

To evaluate the performance of the proposed model, we conduct experiments on four benchmark datasets: PSM, SMD, MSL, and SMAP. The results show that the proposed model achieves competitive anomaly detection performance compared with existing methods. These results indicate that TRUSS provides stable and robust anomaly detection performance across diverse multivariate time series datasets.

CCS Concepts

• Information systems → Data mining; • Computing methodologies → Machine learning.

Keywords

Multivariate Time Series, Anomaly Detection, State-Space Model, Uncertainty Modeling, Template Learning

ACM Reference Format:

Euichan Moon and Youjin Shin. 2026. TRUSS: Template-guided Residual Uncertainty State-Space Model. In *Proceedings of the 32nd ACM SIGKDD*

*Corresponding Author

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

KDD '26, Jeju, Republic of Korea

© 2026 Copyright held by the owner/author(s). Publication rights licensed to ACM.

Conference on Knowledge Discovery and Data Mining (KDD '26), August 9–13, 2026, Jeju, Republic of Korea. ACM, New York, NY, USA, 7 pages.

1 Introduction

Time series anomaly detection (TSAD) aims to identify time points or temporal segments that deviate from expected behavior. Unlike conventional outlier detection on independent observations, TSAD must consider temporal context: an observation may appear normal in isolation but become anomalous given its preceding behavior, while a sequence of plausible observations may collectively form an abnormal pattern [3, 4]. In multivariate time series, anomalies may also arise from unexpected relationships among variables. In the common unsupervised setting, models learn normal behavior from unlabeled or predominantly normal training data and score deviations from the learned patterns. Research has consequently progressed from statistical and similarity-based detection to deep learning methods that learn temporal and multivariate patterns directly from data.

Forecasting- and reconstruction-based methods are two major deep learning approaches to TSAD. Forecasting methods predict future observations from preceding data and treat large prediction errors as anomalous. Representative methods include LSTM forecasting with dynamic thresholding and GDN, which models variable dependencies as a graph [7, 11]. MTAD-GAT further combines forecasting and reconstruction while modeling temporal and inter-variable relationships [21]. Reconstruction methods instead learn to reproduce normal inputs and detect poorly reconstructed observations. MSCRED reconstructs correlation-based signature matrices, USAD uses adversarially trained autoencoders, TranAD employs a self-conditioned Transformer, and NPSR combines point- and sequence-based reconstruction [2, 12, 15, 20]. Despite their architectural differences, these methods largely rely on error magnitude, which can be ambiguous when normal variability changes across temporal contexts.

Beyond error magnitude, other methods detect anomalies through temporal or multivariate relationships. Anomaly Transformer measures discrepancies between temporal associations, while DCdetector uses dual-attention contrastive learning to obtain discriminative temporal representations [18, 19]. CARLA uses synthetic anomalies as contrasting examples to separate normal and abnormal representations [6]. SARAD learns and reconstructs spatial associations among variables, while CATCH captures frequency-domain patterns and channel relationships through frequency patching [5, 16]. These approaches capture abnormal relationships that pointwise errors may overlook, but focus primarily on representation or relationship discrepancies rather than reconstruction likelihood under expected uncertainty.

Probabilistic reconstruction addresses normal variability by estimating a reconstruction distribution rather than a single value. Donut uses a variational autoencoder to estimate reconstruction probabilities for seasonal univariate time series, while OmniAnomaly combines stochastic recurrent modeling with reconstruction probability for multivariate time series [14, 17]. SensitiveHUE additionally estimates observation-dependent uncertainty [9]. These methods can assign different anomaly scores to similar reconstruction errors according to expected variability. However, they do not explicitly use recurring patch-level normal structures as a structural component of the reconstruction mean.

Memory-based methods complement this direction by providing explicit references for diverse normal patterns. MEMTO stores representative normal patterns in a learnable memory, constrains reconstruction using retrieved representations, and measures deviations in input and latent spaces [13]. However, its memory representations do not directly form a patch-shaped reconstruction base that is separately refined through residual correction.

These studies highlight the complementary roles of temporal dynamics, multivariate relationships, uncertainty, and representative normal patterns, motivating a reconstruction process that uses a context-dependent structural reference while separately modeling local variation and uncertainty.

To this end, we propose the Template-guided Residual Uncertainty State-Space Model (TRUSS). Following the general Recurrent State-Space Model (RSSM) design [10], TRUSS uses a deterministic state to summarize preceding patches and a stochastic state to capture patch-level variation. From this temporal context, the Template Composer combines patch-shaped patterns from a learnable Template Bank to construct an expected normal structure. The Residual Uncertainty Decoder then estimates a residual correction and observation-dependent uncertainty from the temporal states and composed template.

TRUSS reconstructs each patch as a Gaussian distribution whose mean combines the normal template and residual correction, while its scale represents reconstruction uncertainty. The template captures representative structure, and the residual accounts for unexplained local variation. TRUSS is trained on normal data using Gaussian negative log-likelihood with latent-state and template-selection regularization. During inference, negative log-likelihood serves as the anomaly score, reflecting reconstruction deviation relative to its estimated uncertainty.

The main contributions of this paper are summarized as follows:

- We propose TRUSS, which integrates recurrent state modeling, template-guided reconstruction, residual correction, and uncertainty estimation for multivariate time series anomaly detection.
- We introduce a structured probabilistic reconstruction whose mean combines a representative normal template and a residual correction, with anomalies detected using uncertainty-aware negative log-likelihood.
- We evaluate TRUSS on PSM, SMD, MSL, and SMAP and conduct ablation studies to examine the contributions of the proposed components.

2 Method

2.1 Overall Architecture

The overall architecture of the Template-guided Residual Uncertainty State-Space Model (TRUSS) is shown in Figure 1. TRUSS divides a multivariate time series into patches and processes them in temporal order. At each step t , the three colored regions form a sequential reconstruction flow: the Recurrent State-Space Model captures temporal context, the Template Composer constructs the expected normal structure, and the Residual Uncertainty Decoder reconstructs the current patch as a probability distribution. The temporal state is then updated for the next patch.

The Recurrent State-Space Model in Figure 1(a) maintains a deterministic state for temporal context and a stochastic state for patch-level variation. The posterior estimates the stochastic state using the current patch and context, whereas the prior uses only the context. The decoder uses the posterior during training and the prior during inference, with KL divergence aligning the two distributions. The posterior also updates the deterministic state through the GRU.

The Template Composer in Figure 1(b) maps the deterministic state to weights over the Template Bank and combines the templates into an expected normal structure. Reference weights derived from the standardized current patch supervise this process during training, whereas inference uses only the deterministic state.

The Residual Uncertainty Decoder in Figure 1(c) receives the deterministic state, the stochastic state selected for training or inference, and the composed template. It predicts a residual correction and reconstruction uncertainty, which define the Gaussian reconstruction together with the template. The model is trained jointly using reconstruction, latent-state, and template-selection objectives, and uses negative log-likelihood as the anomaly score at inference. The following subsections describe this sequence in detail.

2.2 Recurrent State-Space Model

The reconstruction process begins with the Recurrent State-Space Model, which relates each patch to its preceding temporal context. Let $X = [x_1, \dots, x_N]^T \in \mathbb{R}^{N \times C}$ denote a multivariate time series of length N , where $x_n \in \mathbb{R}^C$ contains measurements from C channels at time n . Using patch length L and stride S , TRUSS constructs $P = \lfloor (N - L)/S \rfloor + 1$ ordered patches $o_t \in \mathbb{R}^{L \times C}$, where the t -th patch starts at $i_t = 1 + (t - 1)S$.

Following the general RSSM formulation [10], the model represents the patch sequence using a deterministic state $h_t \in \mathbb{R}^{D_h}$ and a stochastic state $s_t \in \mathbb{R}^{D_s}$, where D_h and D_s are their respective dimensions. The deterministic state h_t summarizes the preceding patches, while s_t captures patch-level variation. For the current patch, the encoder produces $e_t = e_\alpha(o_t)$, where e_α is parameterized by α .

Using only the preceding context, the prior defines

$$p_\psi(s_t | h_t) = \mathcal{N}(\mu_t^p, \text{diag}((\sigma_t^p)^2)), \quad (1)$$

where μ_t^p and σ_t^p are the mean and standard deviation predicted by the prior network parameterized by ψ . This distribution represents the patch-level state expected before observing o_t .

The posterior additionally conditions on e_t :

$$q_\phi(s_t | e_t, h_t) = \mathcal{N}(\mu_t^q, \text{diag}((\sigma_t^q)^2)), \quad (2)$$

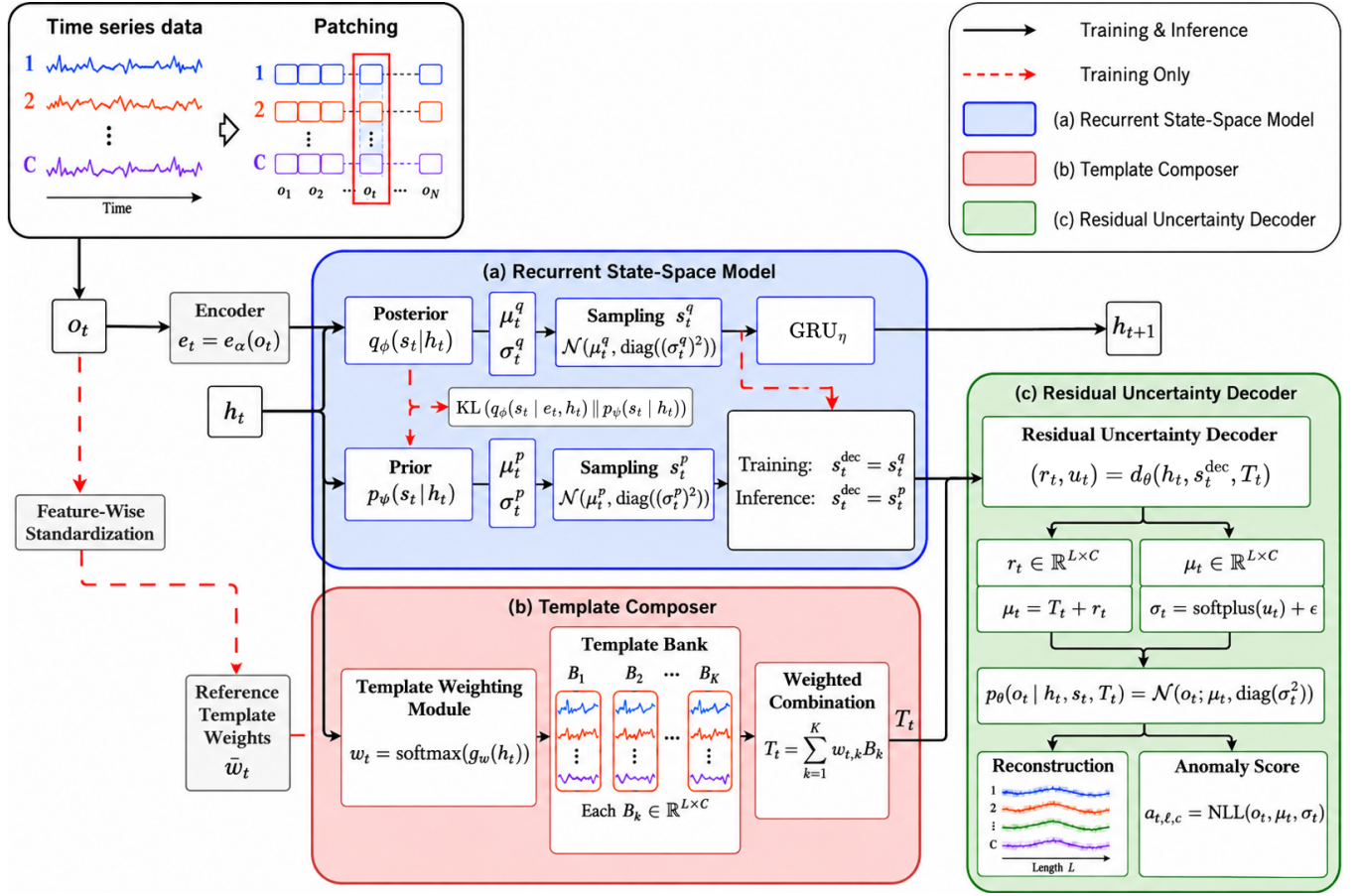


Figure 1: Overall architecture of TRUSS: (a) a Recurrent State-Space Model that captures temporal dynamics, (b) a Template Composer that constructs a context-dependent normal template, and (c) a Residual Uncertainty Decoder that estimates a residual correction and reconstruction uncertainty. Solid arrows are used in both training and inference and red dashed arrows only during training.

where μ_t^q and σ_t^q are predicted by the posterior network parameterized by ϕ . Samples from the prior and posterior are denoted by s_t^p and s_t^q , respectively.

During training, a posterior sample is used for reconstruction, and the posterior distribution is regularized toward the prior through

$$\mathcal{L}_{KL}^{(t)} = D_{KL}(q_\phi(s_t | e_t, h_t) \| p_\psi(s_t | h_t)), \quad (3)$$

where D_{KL} denotes the Kullback–Leibler divergence. This encourages the prior to approximate the latent state inferred from normal observations.

The decoder therefore uses

$$s_t^{\text{dec}} = \begin{cases} s_t^q, & \text{during training,} \\ s_t^p, & \text{during inference.} \end{cases} \quad (4)$$

In both phases, the posterior sample updates the deterministic state as $h_{t+1} = \text{GRU}_\eta(s_t^q, h_t)$, where η denotes the GRU parameters. Thus, inference reconstructs the current patch from the prior, while the posterior incorporates the observation into subsequent context; the KL loss is used only during training.

2.3 Template Composer

The temporal state h_t is next converted into an explicit structural reference by combining patterns from a Template Bank. The bank contains K patch-shaped templates:

$$\mathcal{B} = \{B_1, \dots, B_K\}, \quad B_k \in \mathbb{R}^{L \times C}. \quad (5)$$

Template Bank initialization. To obtain diverse initial normal shapes, each training patch is standardized independently by channel:

$$\tilde{o}_{t,:c} = \frac{o_{t,:c} - \bar{o}_{t,c}}{\hat{\sigma}_{t,c} + \epsilon}, \quad (6)$$

where $\bar{o}_{t,c}$ and $\hat{\sigma}_{t,c}$ are the within-patch mean and standard deviation for channel c , and $\epsilon > 0$ ensures numerical stability. Each standardized channel is approximated by a cubic polynomial:

$$\tilde{o}_{t,\ell,c} \approx \sum_{m=0}^3 a_{t,\ell,c}^{(m)} \xi_\ell^m, \quad (7)$$

where ξ_ℓ is the temporal coordinate at patch position ℓ . The $4C$ coefficients are concatenated as

$$v_t = \left[a_{t,1}^{(0)}, \dots, a_{t,1}^{(3)}, \dots, a_{t,C}^{(0)}, \dots, a_{t,C}^{(3)} \right]^\top \in \mathbb{R}^{4C}. \quad (8)$$

K-means clustering is applied to $\{v_t\}_{t=1}^P$, and each cluster center is evaluated at the L temporal positions to obtain an initial template B_k . The templates are subsequently optimized as learnable parameters.

Template composition. The Template Weighting Module uses h_t to select templates appropriate for the current context:

$$w_t = \text{softmax}(g_\omega(h_t)), \quad w_t \in \mathbb{R}^K, \quad (9)$$

where g_ω has parameters ω , $w_{t,k} \geq 0$, and $\sum_{k=1}^K w_{t,k} = 1$. The template for the current patch is

$$T_t = \sum_{k=1}^K w_{t,k} B_k. \quad (10)$$

The weighting module is supervised by a reference distribution derived from the standardized current patch. The distance from the current patch to template B_k is

$$d_{t,k} = \frac{1}{LC} \|\tilde{o}_t - B_k\|_F^2, \quad (11)$$

and the corresponding reference weights are

$$\bar{w}_t = \text{softmax}\left(-\frac{[d_{t,1}, \dots, d_{t,K}]}{\tau}\right), \quad (12)$$

where τ controls their concentration. The template selection loss is

$$\mathcal{L}_{temp}^{(t)} = -\sum_{k=1}^K \bar{w}_{t,k} \log w_{t,k}. \quad (13)$$

The reference weights are used only for training supervision; at inference, w_t and T_t are computed solely from h_t .

2.4 Residual Uncertainty Decoder

The composed template T_t provides an expected normal shape but cannot represent all patch-specific variation. The Residual Uncertainty Decoder therefore combines it with h_t and s_t^{dec} to refine the reconstruction and estimate its uncertainty:

$$(r_t, u_t) = d_\theta(h_t, s_t^{\text{dec}}, T_t), \quad (14)$$

where d_θ has parameters θ and $r_t, u_t \in \mathbb{R}^{L \times C}$. Here, r_t corrects variation not represented by the template, while u_t determines reconstruction uncertainty. The Gaussian reconstruction parameters are

$$\mu_t = T_t + r_t, \quad (15)$$

$$\sigma_t = \text{softplus}(u_t) + \epsilon. \quad (16)$$

The softplus transformation keeps every element of σ_t positive. TRUSS therefore defines

$$p_\theta(o_t | h_t, s_t^{\text{dec}}, T_t) = \mathcal{N}(\text{vec}(o_t); \text{vec}(\mu_t), \text{diag}(\text{vec}(\sigma_t^2))). \quad (17)$$

Here, $\text{vec}(\cdot)$ stacks the patch entries into a vector.

The element-wise negative log-likelihood is

$$\ell_{t,\ell,c} = \frac{1}{2} \left[\left(\frac{o_{t,\ell,c} - \mu_{t,\ell,c}}{\sigma_{t,\ell,c}} \right)^2 + 2 \log \sigma_{t,\ell,c} + \log(2\pi) \right], \quad (18)$$

and the reconstruction loss for patch t is

$$\mathcal{L}_{rec}^{(t)} = \sum_{\ell=1}^L \sum_{c=1}^C \ell_{t,\ell,c}. \quad (19)$$

The standardized residual measures reconstruction mismatch relative to the predicted scale, while the logarithmic term prevents arbitrarily large uncertainty.

2.5 Training Objective and Anomaly Scoring

The reconstruction, KL, and template-selection losses are jointly optimized over the training patches as

$$\mathcal{L} = \sum_{t=1}^P \left(\lambda_{rec} \mathcal{L}_{rec}^{(t)} + \lambda_{KL} \mathcal{L}_{KL}^{(t)} + \lambda_{temp} \mathcal{L}_{temp}^{(t)} \right), \quad (20)$$

where λ_{rec} , λ_{KL} , and λ_{temp} weight the reconstruction, KL, and template selection losses, respectively.

At inference, the channel-averaged negative log-likelihood at position ℓ of patch t is used as the anomaly score:

$$a_{t,\ell} = \frac{1}{C} \sum_{c=1}^C \ell_{t,\ell,c}. \quad (21)$$

For overlapping patches, scores mapped to the same original time point are aggregated.

3 Experiment

3.1 Datasets

We evaluate our proposed model, Template-guided Residual Uncertainty State-Space Model (TRUSS), on four widely used benchmark datasets for time series anomaly detection: PSM, SMD, MSL, and SMAP.

- **PSM (Public Server Machine)** [1]: Server and system monitoring data with multivariate operational metrics collected from real service environments.
- **SMD (Server Machine Dataset)** [14]: Server machine data collected from multiple machines, including normal operating states and failure-related abnormal patterns.
- **MSL (Mars Science Laboratory)** [11]: Mars Science Laboratory telemetry data containing multivariate spacecraft signals, complex temporal patterns, and anomalous intervals.
- **SMAP (Soil Moisture Active Passive)** [11]: Soil Moisture Active Passive satellite telemetry data containing multivariate system signals and abnormal intervals.

3.2 Experimental Setup

For all datasets, each multivariate time series is divided into patches using a fixed patch size of 64 and a stride of 10. TRUSS is trained using the same training objective defined in Section 2.5. The loss weighting coefficients are set to $\lambda_{rec} = 0.45$, $\lambda_{KL} = 0.20$, and $\lambda_{temp} = 0.35$ for all datasets.

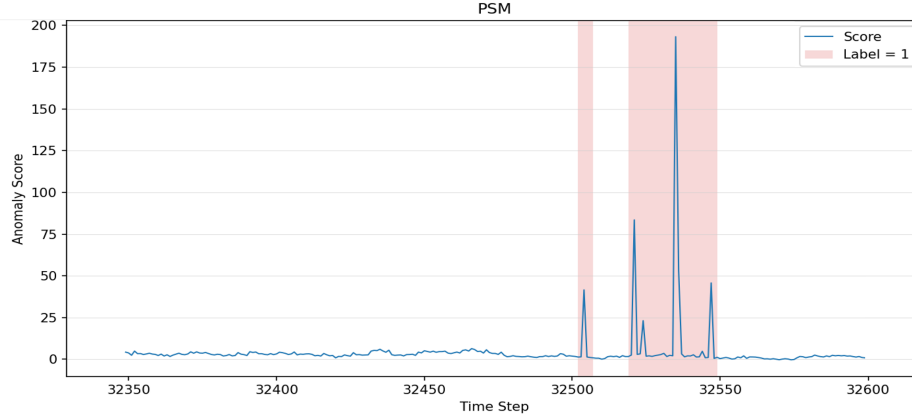
During inference, the latent state is sampled from the prior distribution instead of the posterior distribution, and the template weights are predicted from the hidden state h_t using the weighting module. Each patch is reconstructed using the prior latent state and the composed template, and anomaly scores are computed using the Gaussian reconstruction negative log-likelihood.

Table 1: Anomaly detection performance on PSM, SMD, MSL, and SMAP. Best and second-best results are determined based on the F1-score and are marked in bold and underlined, respectively.

Model	PSM			SMD			MSL			SMAP		
	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
DCDetector [19]	0.4022	0.3857	0.3938	0.1159	0.7248	0.1999	0.1633	0.2604	0.2007	0.1340	0.1470	0.1402
CARLA [6]	0.3861	1.0000	0.5571	0.6757	0.8465	0.7515	0.6617	0.7143	<u>0.6870</u>	0.6342	0.8935	<u>0.7418</u>
MEMTO [13]	0.2859	0.0305	0.0551	0.0598	0.0565	0.0581	0.0885	0.0136	<u>0.0237</u>	0.0261	0.0102	<u>0.0147</u>
SARAD [5]	0.5274	0.9119	0.6683	0.3430	0.6399	0.4466	0.2440	0.8917	0.3831	0.1768	0.9442	0.2978
SensitiveHUE [9]	0.6050	0.9540	<u>0.7404</u>	0.6833	0.7832	0.7299	0.5202	1.0000	0.6843	0.4605	0.9676	0.6241
NPSR [12]	0.9020	0.4376	0.5893	0.3054	0.2793	0.2918	0.1760	0.7777	0.2870	0.3265	0.4755	0.3871
CATCH [16]	0.7114	0.7004	0.7059	0.3009	0.5147	0.3798	0.2317	0.8687	0.3659	0.2380	0.4356	0.3078
CGAD [8]	0.5276	0.9555	0.6798	0.5538	0.8172	0.6602	0.3960	0.9878	0.5654	0.3775	0.9511	0.5405
Ours	0.6776	0.9586	0.7940	0.6295	0.8942	<u>0.7389</u>	0.6446	0.7513	0.6939	0.7008	0.7941	0.7445
Improvement	+7.23%			-			+1.01%			+0.36%		

Table 2: Ablation study results on four benchmark datasets. $\Delta F1$ indicates the relative F1 improvement over Vanilla RSSM.

Model	PSM				SMD				MSL				SMAP			
	Pre	Rec	F1	$\Delta F1$	Pre	Rec	F1	$\Delta F1$	Pre	Rec	F1	$\Delta F1$	Pre	Rec	F1	$\Delta F1$
Vanilla RSSM	0.5223	0.9126	0.6644	-	0.6117	0.7362	0.6682	-	0.4591	0.9913	0.6276	-	0.4519	1.0000	0.6225	-
+ Template Composer	0.5246	0.9684	0.6805	+2.42%	0.6325	0.8053	0.7085	+6.03%	0.4730	0.9803	0.6382	+1.69%	0.5243	0.9642	0.6792	+9.11%
+ Residual Uncertainty Decoder	0.5054	0.9601	0.6622	-0.33%	0.6661	0.7034	0.6842	+2.39%	0.4832	0.9772	0.6466	+3.03%	0.5246	0.9779	0.6829	+9.70%
Full Model	0.6776	0.9586	0.7940	+19.51%	0.6295	0.8942	0.7389	+10.58%	0.6446	0.7513	0.6939	+10.56%	0.7008	0.7941	0.7445	+19.60%

**Figure 2: Anomaly score visualization on a selected PSM segment. The blue line denotes the anomaly score, and shaded regions indicate ground-truth anomalies.**

3.3 Baselines

We compare TRUSS with eight representative baselines for multivariate time series anomaly detection. DCDetector [19] and CARLA [6] are included as contrastive-learning and representation-learning approaches, which aim to learn discriminative temporal representations for anomaly detection. MEMTO [13], NPSR [12], and SensitiveHUE [9] are representative normal-pattern modeling approaches, where anomalies are identified based on deviations from learned normal temporal behaviors. In addition, SARAD [5], CATCH [16], and CGAD [8] cover spatial association modeling, frequency-domain patching, and graph-based causal modeling, respectively.

3.4 Evaluation Metric

We evaluate anomaly detection performance using PA%K-adjusted precision, recall, and F1-score. PA%K (Point Adjustment at K%) reflects the segment-level nature of time series anomalies by applying point adjustment only when a sufficient portion of an anomaly segment is detected.

For each ground-truth anomaly segment, the ratio of correctly detected anomaly points is computed. If at least $K\%$ of the points within the segment are detected as anomalies, the segment is regarded as correctly detected, and all points in the segment are

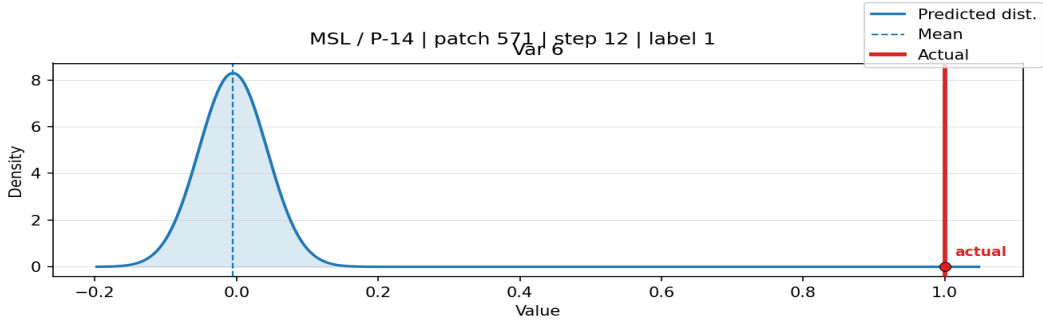


Figure 3: Predicted reconstruction distributions for selected variable in an anomalous MSL segment. The dashed line denotes the predicted mean, and the red line denotes the actual value.

adjusted as true positives. Otherwise, no adjustment is applied, and the original point-wise predictions are used for evaluation.

This criterion reduces overly optimistic performance estimates and penalizes fragmented or partial detections. In our experiments, we use PA%K with $K = 20$.

4 Results

To evaluate the anomaly detection performance of the Template-guided Residual Uncertainty State-Space Model (TRUSS), we compare TRUSS with eight recent baseline methods on four multivariate time series datasets: PSM, SMD, MSL, and SMAP.

TRUSS achieves competitive performance across all evaluated benchmarks, obtaining the best F1-score on PSM, MSL, and SMAP and the second-best score on SMD. Compared with the strongest baseline on each dataset, TRUSS achieves relative F1 improvements of 7.23%, 1.01%, and 0.36% on PSM, MSL, and SMAP, respectively. On SMD, it achieves an F1-score of 0.7389, close to CARLA’s best result of 0.7515. TRUSS also records strong recall on PSM and SMD and the highest precision on SMAP, demonstrating consistent performance across different evaluation metrics.

Figure 2 presents anomaly scores for a selected PSM segment. The scores increase clearly in ground-truth anomalous regions while remaining relatively stable in normal regions. For sustained anomalies, they remain elevated throughout the anomalous interval, showing close alignment with the ground-truth labels.

Figure 3 shows the predicted reconstruction distributions for selected variable in an anomalous MSL segment. Observed values deviate substantially from the predicted distributions in anomalous regions, resulting in lower reconstruction likelihoods. This example illustrates how uncertainty-aware probabilistic reconstruction distinguishes anomalous observations from expected normal patterns.

5 Ablation Study

We conduct an ablation study to analyze the effectiveness of the Template Composer and the Residual Uncertainty Decoder. As shown in Table 2, adding the Template Composer improves the F1-score across all datasets, with gains of +2.42% on PSM, +6.03% on SMD, +1.69% on MSL, and +9.11% on SMAP. This result indicates that representative temporal templates provide useful structural guidance for reconstruction.

The Residual Uncertainty Decoder also improves performance on most datasets, achieving F1 gains of +2.39% on SMD, +3.03% on MSL, and +9.70% on SMAP, although a slight decrease is observed on PSM. When both components are combined, the full model achieves the largest improvements over Vanilla RSSM on all datasets, with gains of +19.51%, +10.58%, +10.56%, and +19.60% on PSM, SMD, MSL, and SMAP, respectively. These results show that the Template Composer and the Residual Uncertainty Decoder provide complementary contributions to the overall performance.

6 Conclusion

In this paper, we proposed the Template-guided Residual Uncertainty State-Space Model (TRUSS) for multivariate time series anomaly detection. TRUSS integrates template-guided reconstruction and residual uncertainty modeling into the RSSM framework, enabling the model to capture representative normal temporal structures and probabilistic deviations from them.

Experiments on four benchmark datasets show that TRUSS achieves stable anomaly detection performance, obtaining the best F1-score on PSM, MSL, and SMAP and the second-best F1-score on SMD. The ablation study further confirms that the Template Composer and Residual Uncertainty Decoder contribute to template-guided structural reconstruction and residual uncertainty modeling, respectively, and their combination yields the largest performance gains.

Overall, TRUSS provides an effective framework for modeling complex multivariate temporal patterns by integrating template-based structural reconstruction with probabilistic uncertainty estimation.

7 Limitations & Future Work

TRUSS has two main limitations. First, cubic polynomial fitting and K-means initialization capture broad temporal shapes but may overlook abrupt, local, or multi-scale patterns. Multi-resolution or frequency-domain representations could improve the expressiveness of the Template Bank.

Second, TRUSS is trained offline on predominantly normal data and therefore cannot directly adapt to changing operating conditions, seasonality, or concept drift. Online or continual learning could enable adaptive updates of the Template Bank and latent dynamics while limiting contamination from anomalies.

GenAI Usage Disclosure

The authors used GPT-5.5 to assist with grammar correction, word-refinement, and improving the readability of the manuscript. No generative AI tool was used to generate experimental results, analyze data, or draw scientific conclusions. All AI-assisted outputs were carefully reviewed and edited by the authors, who take full responsibility for the final manuscript.

References

- [1] Ahmed Abdulaal, Zhuanghua Liu, and Tomer Lancewicki. 2021. Practical Approach to Asynchronous Multivariate Time Series Anomaly Detection and Localization. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2485–2494. doi:10.1145/3447548.3467174
- [2] Julien Audibert, Pietro Michiardi, Frédéric Guyard, Sébastien Marti, and Maria A. Zuluaga. 2020. USAD: UnSupervised Anomaly Detection on Multivariate Time Series. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 3395–3404. doi:10.1145/3394486.3403392
- [3] Ane Blázquez-García, Angel Conde, Usue Mori, and Jose A. Lozano. 2021. A Review on Outlier/Anomaly Detection in Time Series Data. *Comput. Surveys* 54, 3, Article 56 (2021), 33 pages. doi:10.1145/3444690
- [4] Varun Chandola, Arindam Banerjee, and Vipin Kumar. 2009. Anomaly Detection: A Survey. *Comput. Surveys* 41, 3, Article 15 (2009), 58 pages. doi:10.1145/1541880.1541882
- [5] Zhihao Dai, Ligang He, Shuang-Hua Yang, and Matthew Leeke. 2024. SARAD: Spatial Association-Aware Anomaly Detection and Diagnosis for Multivariate Time Series. In *Advances in Neural Information Processing Systems*, Vol. 37. 48371–48410. doi:10.52202/079017-1533
- [6] Zahra Zamanzadeh Darban, Geoffrey I. Webb, Shirui Pan, Charu C. Aggarwal, and Mahsa Salehi. 2025. CARLA: Self-Supervised Contrastive Representation Learning for Time Series Anomaly Detection. *Pattern Recognition* 157 (2025), 110874. doi:10.1016/j.patcog.2024.110874
- [7] Ailin Deng and Bryan Hooi. 2021. Graph Neural Network-Based Anomaly Detection in Multivariate Time Series. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 4027–4035. doi:10.1609/aaai.v35i5.16523
- [8] Falih Gozi Febrinanto, Kristen Moore, Chandra Thapa, Mujie Liu, Vidya Saikrishna, Jiangang Ma, and Feng Xia. 2025. Entropy Causal Graphs for Multivariate Time Series Anomaly Detection. *ACM Transactions on Intelligent Systems and Technology* 16, 6, Article 125 (2025), 25 pages. doi:10.1145/3757922
- [9] Yuye Feng, Wei Zhang, Yao Fu, Weihao Jiang, Jiang Zhu, and Wenqi Ren. 2024. SensitiveHUE: Multivariate Time Series Anomaly Detection by Enhancing the Sensitivity to Normal Patterns. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 782–793. doi:10.1145/3637528.3671919
- [10] Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. 2019. Learning Latent Dynamics for Planning from Pixels. In *Proceedings of the 36th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 97)*. PMLR, 2555–2565. <https://proceedings.mlr.press/v97/hafner19a.html>
- [11] Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, and Tom Soderstrom. 2018. Detecting Spacecraft Anomalies Using LSTMs and Non-parametric Dynamic Thresholding. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 387–395. doi:10.1145/3219819.3219845
- [12] Chih-Yu Andrew Lai, Fan-Keng Sun, Zhengqi Gao, Jeffrey H. Lang, and Duane S. Boning. 2023. Nominality Score Conditioned Time Series Anomaly Detection by Point/Sequential Reconstruction. In *Advances in Neural Information Processing Systems*, Vol. 36. 76637–76655. doi:10.52202/075280-3348
- [13] Junho Song, Keonwoo Kim, Jeonglyul Oh, and Sungzoon Cho. 2023. MEMTO: Memory-Guided Transformer for Multivariate Time Series Anomaly Detection. In *Advances in Neural Information Processing Systems*, Vol. 36. 57947–57963. doi:10.52202/075280-2525
- [14] Ya Su, Youjian Zhao, Chenhao Niu, Rong Liu, Wei Sun, and Dan Pei. 2019. Robust Anomaly Detection for Multivariate Time Series through Stochastic Recurrent Neural Network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2828–2837. doi:10.1145/3292500.3330672
- [15] Shreshth Tuli, Giuliano Casale, and Nicholas R. Jennings. 2022. TranAD: Deep Transformer Networks for Anomaly Detection in Multivariate Time Series Data. *Proceedings of the VLDB Endowment* 15, 6 (2022), 1201–1214. doi:10.14778/3514061.3514067
- [16] Xingjian Wu, Xiangfei Qiu, Zhengyu Li, Yihang Wang, Jilin Hu, Chenjuan Guo, Hui Xiong, and Bin Yang. 2025. CATCH: Channel-Aware Multivariate Time Series Anomaly Detection via Frequency Patching. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=m08aK3xxdJ>
- [17] Haowen Xu, Wenxiao Chen, Nengwen Zhao, Zeyan Li, Jiahao Bu, Zhihan Li, Ying Liu, Youjian Zhao, Dan Pei, Yang Feng, Jie Chen, Zhaogang Wang, and Honglin Qiao. 2018. Unsupervised Anomaly Detection via Variational Auto-Encoder for Seasonal KPIs in Web Applications. In *Proceedings of the 2018 World Wide Web Conference*. 187–196. doi:10.1145/3178876.3185996
- [18] Jiehui Xu, Haixu Wu, Jianmin Wang, and Mingsheng Long. 2022. Anomaly Transformer: Time Series Anomaly Detection with Association Discrepancy. In *International Conference on Learning Representations*. https://openreview.net/forum?id=LzQQ89U1qm_
- [19] Yiyuan Yang, Chaoli Zhang, Tian Zhou, Qingsong Wen, and Liang Sun. 2023. DCdetector: Dual Attention Contrastive Representation Learning for Time Series Anomaly Detection. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 3033–3045. doi:10.1145/3580305.3599295
- [20] Chuxu Zhang, Dongjin Song, Yuncong Chen, Xinyang Feng, Cristian Lumezanu, Wei Cheng, Jingchao Ni, Bo Zong, Haifeng Chen, and Nitesh V. Chawla. 2019. A Deep Neural Network for Unsupervised Anomaly Detection and Diagnosis in Multivariate Time Series Data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 1409–1416. doi:10.1609/aaai.v33i01.33011409
- [21] Hang Zhao, Yujing Wang, Juanyong Duan, Congrui Huang, Defu Cao, Yunhai Tong, Bixiong Xu, Jing Bai, Jie Tong, and Qi Zhang. 2020. Multivariate Time-Series Anomaly Detection via Graph Attention Network. In *2020 IEEE International Conference on Data Mining*. 841–850. doi:10.1109/ICDM50108.2020.00093