Integrating Sequence and Image Modeling in Irregular Medical Time Series Through Self-Supervised Learning

Liuqing Chen[1,2], **Shuhong Xiao**[1], **Shixian Ding**[1], **Shanhai Hu**[1], **Lingyun Sun**[1,2] *

[1]College of Computer Science and Technology, Zhejiang University, China
[2]International Design Institute, Zhejiang University, China
sunly@zju.edu.cn

Abstract

Medical time series are often irregular and face significant missingness, posing challenges for data analysis and clinical decision-making. Existing methods typically adopt a single modeling perspective, either treating series data as sequences or transforming them into image representations for further classification. In this paper, we propose a joint learning framework that incorporates both sequence and image representations. We also design three self-supervised learning strategies to facilitate the fusion of sequence and image representations, capturing a more generalizable joint representation. The results indicate that our approach outperforms seven other state-of-the-art models in three representative real-world clinical datasets. We further validate our approach by simulating two major types of real-world missingness through leave-sensors-out and leave-samples-out techniques. The results demonstrate that our approach is more robust and significantly surpasses other baselines in terms of classification performance.

Code — https://github.com/zju-d3/AAAI25-Irregular-Medical-Time-Series

Introduction

Multivariate time series are utilized in various real-world applications, particularly in the medical field, where they are used to record vital signs and laboratory test results for diagnosis (Chaudhary et al. 2020; Brizzi et al. 2022). Typically, these time series are irregular, faced with asynchronicity across sensors and nonuniform sampling in the time domain (Chowdhury et al. 2023; Huang et al. 2024). Moreover, significant missing values are usually present in clinical data collection. For example, random missingness can result from patients joining or leaving treatments midway, or complete absence of data from a sensor when specific tests are not conducted (de Jong et al. 2019). Some public clinical datasets, such as PhysioNet2012, take even a 80% missing rate, posing challenges for data analysis and clinical decision-making (Wang et al. 2024).

Deep learning methods have been widely adopted to model irregular time series. Some methods rely on the assumption of time discretization, utilizing LSTMs (Neil, Pfeiffer, and Liu 2016; Weerakody, Wong, and Wang 2023), RNNs (Che et al. 2018; Ma, Li, and Cottrell 2020; Miao et al. 2021), and Transformers (Horn et al. 2020; Huang et al. 2024) to capture characteristics of discrete sequences. Nonetheless, these methods often face difficulties in accumulating errors from missing observations (Ma et al. 2019). Recently, vision models have also shown promising potential in handling irregular sequence data (Li, Li, and Yan 2024). By transforming series into corresponding RGB representations, visual frameworks can effectively capture dynamic trends and inter-sensor relationships within images (Maroor et al. 2024; Li, Li, and Yan 2024). However, such designs perform poorly with sparse series that exhibit heavy missing rate (Li, Li, and Yan 2024).

We recognize that no one has yet integrated both sequence and image representations in handling irregular medical time series. This introduces a pivotal question: *How can we effectively merge these two distinct representations to improve the robustness of classification for irregular medical time series with extensive missing values?*

To investigate this question, we utilize a joint learning framework that incorporates both sequence and image representations. Additionally, we propose different self-supervised learning (SSL) strategies to enhance the integration and capture of supplementary information across these two representations. Specifically, our approach consists of three main components, as shown in Figure 1. For the sequence modeling branch, we employ a generator-discriminator structure and adopt an adversarial strategy (Ma et al. 2019; Miao et al. 2021) for sequence imputation task to minimize the propagation of cumulative errors. In the image branch, we implement different image transformation strategies to improve the performance on sparse series, and utilize a pre-trained Swin Transformer (Liu et al. 2022; Li, Li, and Yan 2024) to obtain the corresponding image representations. Three different SSL losses are designed: (1) an inter-sequence contrastive loss to stabilize the sequence imputation process; (2) a sequence-image contrastive loss with margin to learn a more generalizable joint representation for downstream classification; and (3) a clustering loss on joint representations to push similar cases closer across different batches.

We conduct experiments on three real-world clinical

*Corresponding Author

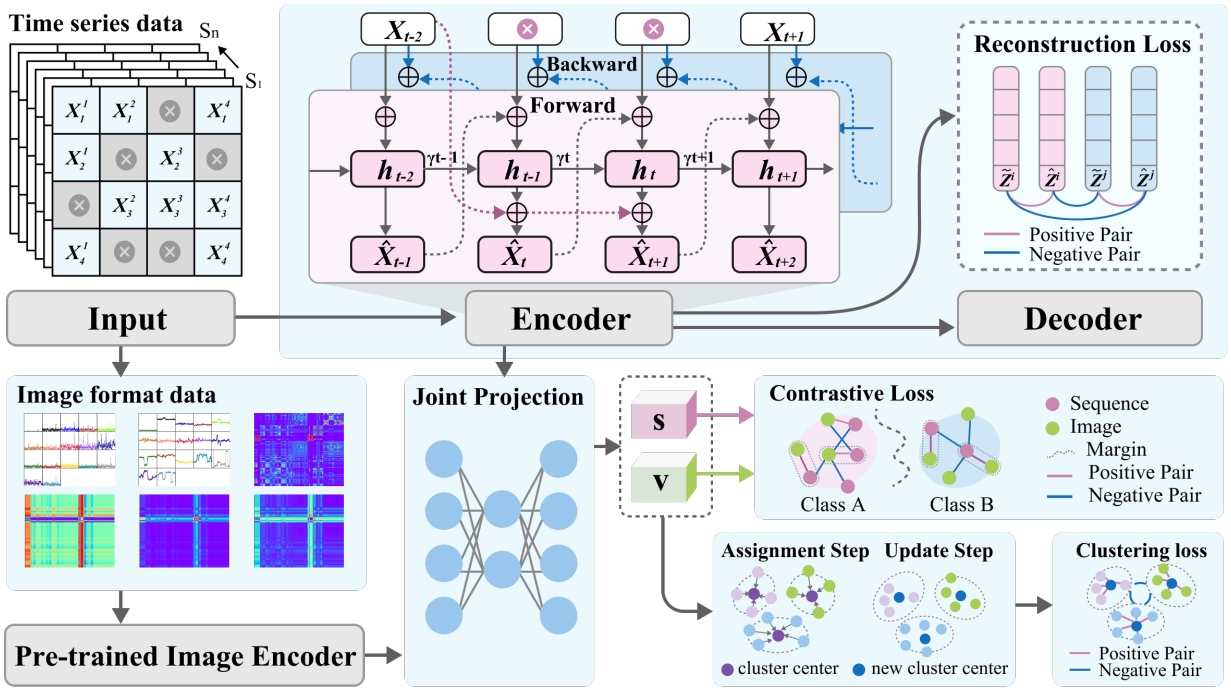

Figure 1: The framework of our approach.

datasets: PAM (Reiss and Stricker 2012), P12 (Goldberger et al. 2000), and P19 (Reyna et al. 2020). Table 1 presents their statistics, which show that all three datasets experience severe missing values. We compare our approach with seven other state-of-the-art (SOTA) methods in terms of classification performance. Specifically, our approach achieves the best performance across all three datasets. For the PAM dataset, we observe improvements of 3.1% in Accuracy, 2.9% in Precision, 2.3% in Recall, and 2.6% in F1 score compared to the second-best method. For the P12 and P19 datasets, we use AUPRC and AUROC as evaluation metrics. Our approach surpasses prior SOTA by 1.1% (AUPRC) and 0.9% (AUROC) on P12, and 5.8% (AUPRC) and 2.3% (AUROC) on P19. Furthermore, we test further missingness through leave-samples-out and leave-sensors-out experiments on the PAM dataset. In the most severe scenario, with an additional 50% missing values, our approach demonstrates better robustness, outperforming the second-best method by 6.1% in Accuracy, 5.9% in Precision, 3.4% in Recall, and 4.6% in F1 score.

The contributions of this paper are summarized as follows:

- We propose a joint representation learning framework for multivariate irregular medical time series. To the best of our knowledge, this is the first approach to incorporate both sequence and image modeling.
- We outline three SSL strategies: inter-sequence contrastive loss, sequence-image contrastive loss, and clustering-based loss. These strategies together enable better integration of sequence and image representations, enhancing the robustness against heavy missingness.
- Our approach outperforms seven other SOTA methods on three real-world clinical datasets. We also simulates two classic types of missingness and experiments show that our method offers better robustness in handling these cases.

Related Work

Irregular Time Series Methods

Early practices for modeling irregular time series with missing values typically relied on fixed-time discretization. In this context, (Choi et al. 2016) ignores the timestamp information by treating all intervals as equal, (Lipton et al. 2016) considers missing data as an effective feature for learning, and (Harutyunyan et al. 2019) segments the data into evenly spaced time intervals. In contrast, GRU-D (Che et al. 2018) employs a gated network and incorporates imputation of missing values into the optimization process. Unlike previous methods, it adopts an additional missing value mask and lag matrix as inputs. Similar strategy have been adopted in (Ma et al. 2019; Ma, Li, and Cottrell 2020; Miao et al. 2021), where adversarial frameworks are utilized to enhance the prediction of imputed values.

Some recent approaches have leveraged attention mechanisms to improve modeling. For instance, SeFT (Horn et al. 2020) introduces a set of differentiable set functions and uses attention mechanisms to aggregate embeddings of different variables. ContiFormer (Chen et al. 2024), on the other hand, combines neural ordinary differential equations (ODEs) with attention mechanisms based on continuous-time dynamics, extending the relationship modeling capabilities of Transformers to the continuous time domain. Be-

sides, DNA-T (Huang et al. 2024) utilizes a deformable attention mechanism to dynamically adjust the receptive field, enabling more effective handling of local features and short-term correlations. Warpformer (Zhang et al. 2023) also considers multi-scale features by applying a warping module to achieve multi-grained representations. Unlike previous methods that adopt a sequence modeling perspective, ViTST (Li, Li, and Yan 2024) transforms the signals into RGB images and utilizes a pre-trained Swin Transformer for further classification and regression.

Modeling Time Series as Images

Transforming time series data into images has gained significant attention with the advancements in visual detection frameworks. Some approaches (Sood et al. 2021; Sangha et al. 2022; Ao and He 2023; Semenoglou, Spiliotis, and Assimakopoulos 2023; Maroor et al. 2024) plot time series directly as time-observation representations and utilize convolutional neural networks (CNNs) for downstream tasks. Generally, they do not apply special processing to the sequences, instead focusing on leveraging visual frameworks to better capture temporal patterns in visualized sequences. ViTST (Li, Li, and Yan 2024) is another similar case that extends further to multivariate sequences and discusses the impact of visualization parameters such as color, markers, and order.

In contrast, other methods emphasize the modeling of time series, which requires more specialized design and expert knowledge. (Tripathy and Acharya 2018) utilizes an iterative filtering (IF) approach to produce different intrinsic mode functions (IMFs) from EEG signals. Empirically, these transformed features often fit the task better than the original signals. Chong et al. (Chong et al. 2011) and Deng et al. (Deng et al. 2023) model sequences based on time segmentation, calculating time-invariant features and transforming them into corresponding RGB images. Similarly, frequency domain modeling, as demonstrated by TimesNet (Wu et al. 2023), has also proven effective. By utilizing fast Fourier transform (FFT) to concatenate signal of different time periods, it constructs a 2D representation optimized for CNNs. Finally, other methods model the relative relationships between points in a time series. Examples include Gramian Angular Field (GAF), Markov Transition Field (MTF), and recurrence plot (Wang and Oates 2015; Hatami, Gavet, and Debayle 2018). Typically, these methods involve applying a reversible time coordinate transformation and calculating the correlations between points, effectively capturing the continuity and periodic characteristics of the sequences.

Approach

Notations

For a given clinical time series dataset D, each sample $X \in \mathbb{R}^{d \times T}$ represents a set of d records over a time $T = \{t_1, ..., t_n\}$, corresponding to a label y. A binary mask $M \in \mathbb{R}^{d \times T}$ is used to indicate the presence of missing observations in X, where $M_i^j = 0$ signifies that the observation of the i^{th} item at time j is missing.

To better handle consecutive missing values time, we follow (Miao et al. 2021; Che et al. 2018) to obtain a time-lag matrix $\delta \in \mathbb{R}^{d \times T}$ for each sample X. This matrix quantifies the time elapsed since the most recent non-missing value for each observation, defined as follows.

$$\delta_i^j = \begin{cases} 0, & i = 1 \\ t_i - t_{i-1}, & m_{i-1}^j = 1 \text{ and } i > 1 \\ \delta_{i-1}^j + t_i - t_{i-1}, & m_{i-1}^j = 0 \text{ and } i > 1 \end{cases}$$

For each sample X, the corresponding image I is constructed, where $I \in \mathbb{R}^{3 \times W \times H}$ represent a certain RGB format image. In total, we implement six transformed images as shown in Figure 1. The specific transformation methods applied are as follows: Line Graphs, Frequency Spectrums, Gramian Angular Summation/Difference Fields, Markov Transition Fields, Recurrence Plots.

The Model Overview

In this section, we introduce the overall framework of our model, which comprises three main parts: (a) the sequence encoder, (b) the image encoder, and (c) the joint representation module. The sequence encoder consists of a generator-discriminator pair employing an adversarial strategy for imputation. The generator, G, takes the time series X, the mask M, and the lag matrix δ as inputs. Its objective is to estimate the missing values in X and generate a completed sequence X'. This completed sequence X' is then used to obtain the sequence representation $s \in \mathbb{R}^d$. The discriminator D evaluates these estimations with the goal of distinguishing true observations from the imputed values. It outputs a binary matrix M', which identifies the regions of imputation predicted. For the image encoder, it takes a transformed image I as input and output the corresponding image representation $v \in \mathbb{R}^d$. Finally, the joint representation module is responsible for mapping the sequence representation s and the image representation v into the same space. It then uses the final joint feature $u \in \mathbb{R}^d$ for classification.

Sequence Branch with Imputation

We adopt a modified bidirectional recurrent neural network (BiRNN) as our generator G, which has been widely used in imputation tasks (Che et al. 2018; Ma et al. 2019; Ma, Li, and Cottrell 2020; Miao et al. 2021; Xu et al. 2024). Taking the forward update step as an example, we update the current hidden state as:

$$h_t = \tanh\left(W_h(\gamma_t \odot h_{t-1}) + W_h'(\hat{x}_t + x_\delta) + b_h\right) \quad (1)$$

$$\gamma_t = \exp\left\{-\max(0, W_\gamma \delta_t + b_\gamma)\right\} \quad (2)$$

In this setup, γ_t is derived from the lag matrix to model the dynamics of decay, where a longer duration of missing data leads γ_t closer to 0. It is applied to determine the extent to which the previous hidden state h_{t-1} should be retained. In the updating process of h_t, instead of solely utilizing the previous reconstruction $\hat{x}_t$ as done in prior works, we introduce an additional computation involving x_δ as Eq. 3.

$$x_\delta = x_{t^-} \cdot \exp\left\{-\max(0, W_\delta \delta_t) + b_\delta\right\} \quad (3)$$

This assumes the closest observation x_{t^-} prior to the current missing value influences the reconstruction process, with this influence decreasing as the time gap increases.

Then, using a fully connected layer, the new reconstruction of the next step is obtained as: $\hat{x}_{t+1} = W_{\hat{x}} h_t + b_{\hat{x}}$. And the overall imputed sequence X' is represented as: $X' = M \odot X + (1 - M) \odot avg(\hat{X}_{for} + \hat{X}_{back})$, where we take the average of forward and backward result, and only the missing parts are replaced. Finally, the sequence representation s is obtained as:

$$s = Drop(W_s \cdot LayerNorm(X') + b_s) \tag{4}$$

In particular, W_h, W'_h, W_γ, W_δ, $W_{\hat{x}}$, W_s, b_h, b_x, b_γ, b_δ, $b_{\hat{x}}$, and b_s are learnable parameters of the model and $\odot$ denotes the element-wise multiplication.

We formulate the objective of generator G into two components: adversarial loss and reconstruction loss. The adversarial loss is defined as the standard GAN's (Goodfellow et al. 2020):

$$\mathcal{L}_{adv} = \mathbb{E}[(1 - M) \log(1 - D(X'))] \tag{5}$$

For the reconstruction loss, previous methods often use regression-based metrics such as mean square error (MSE) (Ma, Li, and Cottrell 2020) or mean absolute error (MAE) (Ma et al. 2019) to assess the consistency between the missing and imputed sequences. However, when dealing with severely missing data, these strategies often fail to model the underlying data patterns, force the generator to learn nothing during the adversarial training phase. Inspired by (Raghu et al. 2023), we adopt a self-learning strategy to construct our reconstruction loss, and one choice is the normalized temperature-scaled cross-entropy loss (NT-Xent) (Chen et al. 2020). Given $2B$ pairs (z_i, z_j) totally, it is computed as:

$$\mathcal{L}_{NT} = \frac{1}{2B} \sum_{i=1}^{2B} -\log \frac{\exp(sim(z_i, z_j)/\tau)}{\sum_{k=1}^{2B} 1_{[k \neq i]} \exp(sim(z_i, z_k)/\tau)} \tag{6}$$

where cosine similarity is used as $sim(z_i, z_j)$ and τ is the temperature hyperparameter. We use NT-Xent to enforce consistency between the forward and backward predictions, as well as between the original and imputed sequences. Thus, the reconstruction loss is defined as:

$$\mathcal{L}_{rec} = \mathcal{L}_{NT}(\hat{X}_{for}, \hat{X}_{back}) + \mathcal{L}_{NT}(X, X') \tag{7}$$

We employ the same RNN in (Ma et al. 2019) as our discriminator D, which takes X' as input and determines whether each observation is generated with a binary matrix M'. Therefore, the discriminator is trained by minimizing:

$$\mathcal{L}_{dis} = \mathbb{E}[MlogM' + (1 - M)log(1 - M')] \tag{8}$$

Imaging Time Series

We use a pre-trained Swin Transformer (Liu et al. 2022) as our image encoder. For the given image input I, the Swin Transformer constructs a hierarchical representation to integrate both local and global information. Specifically, at earlier layers, it partitions the input into small patches and progressively merges neighboring patches as depth increases. It employs two types of attention mechanisms: window-based multi-head self-attention (W-MSA) and shifted window multi-head self-attention (SW-MSA). These mechanisms are respectively used to compute self-attention within a fixed window and to calculate dynamic relationships between windows. The vectors from the last stage after layer normalization are used as our image representation $v \in \mathbb{R}^d$.

Overall, we implement six types of images for representation learning and a detailed description is presented in Appendix A.

- **Line Graphs** are constructed as (Li, Li, and Yan 2024), with each variable represented by a line image of uniform size.
- **Frequency Spectrums** are generated based on the Fourier transform, considering that frequency domain signals tend to be more robust in cases of extreme data missingness.
- **Gramian Angular Fields** (Wang and Oates 2015) transform time series into polar coordinates, constructing trigonometric sums/ differences between any two time points to represent temporal correlation.
- **Markov Transition Fields** record the Markov transition probabilities between any two time observations (Wang and Oates 2015). They are insensitive to the distribution of the time series and temporal step information, allowing them to effectively capture correlations between observations with substantial missing data.
- **Recurrence Plots** (Hatami, Gavet, and Debayle 2018), based on phase space reconstruction, transform time series data into trajectories within phase space and analyze their recurrences. They are designed to capture the inherent repetitiveness and periodicity within the time series.

Joint Representations Through Contrast and Clustering

The joint representation module includes a transformation function $f : s, v \rightarrow \mathbb{R}^D$, which projects and concatenates the sequence features s and image features v into a joint space R^D, and the fused feature is obtained as $u = [s, v]$. To ensure both the quality and consistency of the joint representation, we implement contrastive learning within each batch to maximize the mutual information between corresponding pairs. A simple choice is to use the NT-Xent in Eq. 6, where only sequence and image features corresponding to the same sample are treated as positive pairs (Sangha et al. 2024). Through this approach, NT-Xent ensures that the similarity between representations from the same sample is higher than that of other pairs. However, it also misses opportunities to learn from a wider set of potential pairs (Li, Torr, and Lukasiewicz 2022).

In this case, a step forward is to treat s and v from different samples within the different category as a special form of negative pairs, thereby enhancing the model's ability to distinguish inter-class differences. Specifically, we introduce an additional margin m for these special negative pairs, enforce the model to exert greater effort to distinguish them:

$$-\frac{1}{B} \sum_{i=1}^{B} \left[\log \left(\frac{\exp((v_i \cdot s_i)/\tau)}{\sum_{j \in P(i)} \exp((v_i \cdot s_j)/\tau) + \sum_{j \notin P(i)} \exp((v_i \cdot s_j + m)/\tau)} \right) \right.$$

$$+\log\left(\frac{\exp((v_i\cdot s_i)/\tau)}{\sum_{k\in P(i)}\exp((v_k\cdot s_i)/\tau)+\sum_{k\notin P(i)}\exp((v_k\cdot s_i+m)/\tau)}\right] \quad (9)$$

Here, for the i^{th}sample, $P(i)$ represents the set of all sample index that are in the same category.

In contrastive learning, the formation of positive and negative pairs is confined to each batch. However, this approach lacks control over the semantic relationships between samples across different batches. As a result, similar samples from separate batches may not receive similar representations. In this case, we incorporate clustering learning into the training process to push semantically similar samples together across batches.

Specifically, we applied the K-means algorithm to the fused feature u. We begin with the assignment step: during each training epoch, we select a set of k (k $\ll$ N) representative features $[C_{u_1}, \ldots, C_{u_k}]$ as the cluster centers for that round. Each fused feature u_i is assigned to a set S_k with center C_{u_k} by minimizing the overall distance as defined in Eq. 10.

$$\underset{S}{\operatorname{argmin}} \sum_{j=1}^{k} \sum_{u_i \in S_j} \|u_i - C_{u_j}\|^2 \quad (10)$$

We then use these cluster centers, $[C_{u_1}, \ldots, C_{u_k}]$, as contrastive loss reference targets to construct the clustering loss:

$$\mathcal{L}_{cluster} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(\cos(u_i, C_{u_i})/\tau)}{\sum_{j=1}^{k}\exp(\cos(u_i, C_{u_j})/\tau)} \quad (11)$$

where a cluster center C_{u_k} and all elements within set S_k are treated as positive pairs, and elements from different clusters are considered negative pairs. To ensure sufficient samples for optimizing clustering, we perform the update step at the end of each epoch: we iteratively update the cluster centers using Eq. 12 and calculate new assignments with Eq. 10, until the total distance is less than a predefined threshold τ_c.

$$C_k = \underset{u\in S_k}{\operatorname{argmin}} \sum_{u'\in S_k} \|u - u'\|^2 \quad (12)$$

Overall Training Process

The overall training process is divided into three steps as follows:

- Firstly, we fix the generator G and update the discriminator D based on Eq. 8 .
- Next, we update the parameters of G based on the new D, with the objective function $\mathcal{L}_{adv} + \alpha\mathcal{L}_{rec}$.
- Finally, we compute the forward pass of all three components, utilizing the joint feature u to perform classification. For the PAM dataset, we use the Cross Entropy Loss as the classification loss $\mathcal{L}_{clf}$. For the more imbalanced P12 and P19 datasets, we opt for the Focal Loss. The final objective is expressed as $\mathcal{L}_{clf} + \beta_1\mathcal{L}_{cont} + \beta_2\mathcal{L}_{cluster}$.

Dataset	Features	Time	Classes	Missing Ratio	Samples
PAM	17	600	8	60%	5,333
P12	36	215	2	88.4%	11,988
P19	34	60	2	94.9%	38,803

Table 1: Statistics of datasets utilized.

Experiments

Datasets and Metrics

In the experiments, we consider three real-world irregular clinical datasets as shown in Table 1. The physical activity monitoring (PAM) dataset (Reiss and Stricker 2012) focuses on tracking human activities, containing data from eight person who performed nine different actions. This dataset comprises 5,333 samples and captures data from four types of sensors placed at three distinct body locations, encompassing a total of 17 observational variables. The P12 dataset (Goldberger et al. 2000) includes 11,988 patient samples from ICU stays, with 36 measurements each. The binary labels indicate the prognosis for each sample as either survival or not. Finally, the P19 dataset (Reyna et al. 2020) contains data from 38,803 sepsis patients, each with 34 measurements, and a high missing rate of 94.9%. Approximately 90% of these patients died due to sepsis.

To maintain consistency across all experiments, we follow the same data partition as (Zhang et al. 2022; Li, Li, and Yan 2024), dividing the datasets into training, validation, and testing sets in an 8:1:1 ratio. For the PAM dataset, we use Accuracy, Precision, Recall, and F1 score as evaluation metrics. For the more imbalanced P12 and P19 datasets, we report the Area Under the ROC Curve (AUROC) and the Area Under the Precision-Recall Curve (AUPRC). For more experimental results that are not included in this section, we present them in Appendix D.

Implementation and Training

We use Gated Recurrent Units (GRU) (Dey and Salem 2017) in both our generator and discriminator. The generator has 4 layers, with the number of units fixed at 128. The discriminator is a 5-layer RNN and the number of units is set to {128, 64, 16, 64, 128}, respectively. A checkpoint pre-trained on ImageNet-21K dataset are utilized for our image encoder. The patch size and window size are 4 and 7. For the P12 and P19 datasets, all images are set to a size of 384 $\times$ 384 pixels. While for the PAM dataset, line graph and frequency spectrum are configured to 256$\times$ 320, while all other images are set to 320 $\times$ 320. We use a 3-layer MLP as our joint projection, with the number of units set to {1024, 512, 1024}.

For the P12 and P19 datasets, the total epoch is set to 8 and we apply upsampling of the minority class to mitigate imbalance. For the PAM dataset, we set the total epoch to 40. The batch sizes used for training are 32 for P19 and P12, and 48 for PAM. For each dataset, we discuss the learning rate as well as more hyperparameter settings in Appendix B. All experiments are performed on a server with NVIDIA GeForce RTX 3090 24GB and PyTorch 2.4.0+cu124.

Methods	PAM				P12		P19	
	Accuracy	Precision	Recall	F1 score	AUROC	AUPRC	AUROC	AUPRC
GRU-D	83.3 ± 1.6	84.6 ± 1.2	85.2 ± 1.6	84.8 ± 1.2	81.7 ± 1.8	41.3 ± 3.5	83.6 ± 2.1	45.7 ± 4.2
SeFT	63.3 ± 2.2	66.7 ± 2.4	65.3 ± 1.5	65.1 ± 1.8	73.3 ± 2.5	29.1 ± 4.1	84.5 ± 2.3	46.7 ± 3.1
CARD	71.9 ± 2.9	75.5 ± 2.8	73.5 ± 3.1	73.8 ± 3.0	71.4 ± 0.9	26.1 ± 1.2	80.7 ± 1.0	36.7 ± 6.0
Raindrop	89.2 ± 1.3	90.8 ± 1.0	90.4 ± 1.3	90.5 ± 1.2	82.0 ± 2.4	44.3 ± 3.3	82.7 ± 3.9	52.3 ± 3.9
PrimeNet	85.5 ± 1.5	87.8 ± 1.2	87.1 ± 1.1	87.1 ± 1.2	<u>85.1</u> ± 0.8	<u>49.3</u> ± 1.9	80.3 ± 0.5	31.6 ± 0.9
ContiFormer	66.6 ± 1.8	68.6 ± 1.7	69.7 ± 1.5	67.4 ± 1.7	72.1 ± 0.4	29.6 ± 0.8	80.7 ± 0.3	34.7 ± 1.9
ViTST	<u>95.2</u> ± 1.4	<u>95.8</u> ± 1.3	<u>96.1</u> ± 1.1	<u>95.9</u> ± 1.2	84.2 ± 1.1	43.2 ± 2.4	<u>89.3</u> ± 0.2	<u>53.8</u> ± 1.1
Ours	**98.3** ± 0.3	**98.7** ± 0.6	**98.4** ± 1.0	**98.5** ± 0.7	**86.0** ± 0.3	**50.4** ± 2.1	**91.6** ± 0.9	**59.6** ± 1.3

Table 2: Comparison with state-of-the-art baselines on irregularly sampled time series classification. We use **bold** to indicate the best results and <u>underline</u> for the second best one.

Results

Comparison with state-of-the-art methods. We compare our approach against seven state-of-the-art methods for irregularly sampled time series, including GRU-D (Che et al. 2018), SeFT (Horn et al. 2020), CARD (Han, Zheng, and Zhou 2022), Raindrop (Zhang et al. 2022), PrimeNet (Chowdhury et al. 2023), ContiFormer (Chen et al. 2024), and ViTST (Li, Li, and Yan 2024). For each baseline, we introduce our implementation and hyperparameter settings in Appendix C. To ensure a fair evaluation, we average the performance of each method across five individual tests, using the same data splits and settings provided in (Li, Li, and Yan 2024).

Table 2 presents the comparison results, highlighting that our approach outperforms the other seven state-of-the-art methods across all three datasets. Specifically, we achieve a significant improvement on the PAM datasets, with an increase of 3.1% in Accuracy, 2.9% in Precision, 2.3% in recall, and 2.6% in F1 score. For the P12 and P19 datasets, our approach shows improved performance in predicting minority classes, with an increase of 0.9%, 2.3% in absolute AUROC points, and 1.1%, 5.8% in absolute AUPRC, respectively.

Performance under increased missing rates. To further validate the robustness of our approach, we conduct additional experiments to compare the performance under increased levels of missing rate. Given that the P12 and P19 datasets have already faced very high missing rates—88.4% and 94.9% respectively, we conduct all the tests on the PAM dataset, which originally has a missing rate of 60%. We conducted two types of tests: the leave-sensors-out setting, simulating scenarios where certain medical tests are not performed, and the leave-samples-out setting, reflecting situations where patients join or leave treatments midway. We follow the approach in (Zhang et al. 2022), applying all modifications only to the test set by randomly masking the original observations.

As shown in Figure 2, our approach consistently achieves the best performance in all settings. For the leave-sensors-out tests, as the missing ratio increase from 10% to 50%, our approach exhibit the least performance decline. Even in the most extreme scenario, where 50% of the sensors (9 sensors) are masked, all our metrics remain above 80%. Compared to the second-best method, ViTST, our approach outperform by 6.1%, 5.9%, 3.4%, and 4.6% in Accuracy, Precision, Recall, and F1 score, respectively. The margins are even more significant compared to the third-ranked Raindrop, with improvements of 27.4%, 39.8%, 29.9%, and 37.5% in the same metrics. For the leave-samples-out setting, we randomly sampled and masked time steps. Overall, only CARD experienced significant decline as missing rate increases, while most models shows relatively minor decline, indicating that they effectively capture the temporal relationships between time steps. In terms of absolute performance, our model outperform the second-best, ViTST, by 5.7% in Accuracy, 3.8% in Precision, 5.4% in Recall, and 4.8% in F1 score at a 50% missing rate.

Clinical Turing tests. To ensure that the learned representations align with clinically meaningful patterns rather than statistical artifacts, we conducted a clinical Turing test on the generated signals, as described in (Gillette et al. 2023). Specifically, we select 60 samples from the P19 (ICU) dataset, with half imputed using linear interpolation as real measured samples and the other half imputed using our model as generated samples. Five ICU-experienced clinicians (3 chief physicians and 2 attending physicians) attempt to distinguish between the two types. As shown in Table 3, the experts achieve prediction accuracy of 50.0%, 48.3%, 48.3%, 58.3%, and 60.0%, resulting in a kappa score of -0.03. These results are close to random guessing, suggesting that the experts generally struggle to differentiate between the samples. A brief interview further revealed why experts struggled to identify clear patterns to distinguish real from generated samples. One reason is that the complex events in the ICU environment make the data distribution more tolerant. For example, sedation or anesthesia can cause body temperature to fall below the usual range.

Ablation study. In this section, we present the results of our ablation study in Table 4. The "default" one is our standard setup, which includes the sequence encoder, the image encoder, and the joint representation module, along with three self-supervised learning strategies. In the first part of

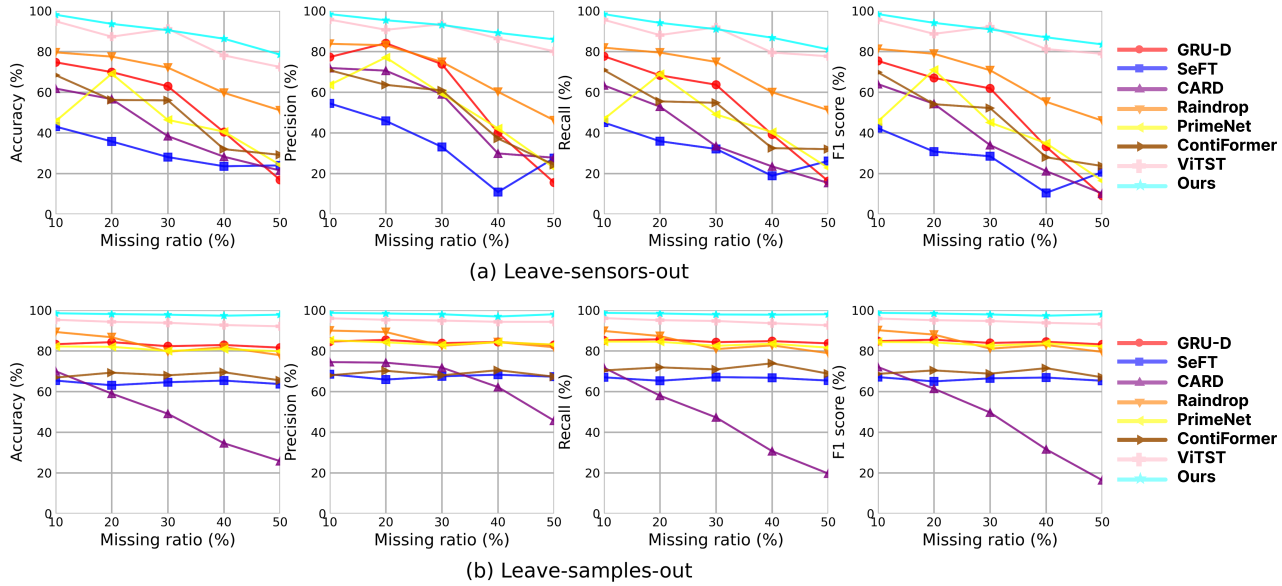

Figure 2: Performance under increased missingness: (a) leave-sensors-out and (b) leave-samples-out on the PAM dataset. Tests are conducted with 10%-50% extra missing values.

Experts	P19				
	Accuracy	Precision	Recall	F1 score	Specificity
P1	50.0	47.4	64.3	54.5	0.38
P2	48.3	44.8	46.4	45.6	0.50
P3	48.3	45.5	53.6	49.2	0.44
P4	58.3	55.2	57.1	56.1	0.59
P5	60.0	57.1	57.1	57.1	0.63

Table 3: Performance metrics of five ICU-experienced medical experts.

Methods	PAM			
	Accuracy	Precision	Recall	F1 score
image	95.4 ± 0.6	96.5 ± 0.6	95.4 ± 0.4	95.9 ± 0.5
sequence	93.3 ± 1.1	94.4 ± 0.7	93.6 ± 0.6	94.0 ± 0.7
sequence-MSE	92.5 ± 0.4	93.8 ± 0.4	93.4 ± 0.4	93.5 ± 0.4
concatenation	95.7 ± 0.7	96.7 ± 0.5	96.1 ± 0.4	96.5 ± 0.5
contrastive	96.8 ± 0.7	97.6 ± 0.5	97.4 ± 0.7	97.5 ± 0.5
clustering	96.9 ± 0.3	97.4 ± 0.6	97.0 ± 0.5	97.3 ± 0.6
default	98.3 ± 0.3	98.7 ± 0.6	98.4 ± 1.0	98.5 ± 0.7

Table 4: Ablation studies on different strategies.

Table 4, we evaluate the performance of individual components: "image" signifies that only the image encoder is used for classification, whereas "sequence" denotes the use of only the sequence encoder. As a result, we verify that incorporating both sequence and image information significantly improves classification performance, with F1 scores increasing by 2.6% and 4.5%. "sequence-MSE" denotes the use of MSE loss as the reconstruction loss. In contrast, by replacing it with NT-Xent, we achieved improvements in Accuracy, Precision, Recall, and F1 score by 0.8%, 0.6%, 0.2%, and 0.5%, respectively.

The second part of Table 4 focuses on our joint representation module. In the "concatenation" setting, we simply concatenate sequence and image representations for further downstream classification, and the performance is slightly higher than either "sequence" and "image". The "contrastive" setting shows the improvement from contrastive learning strategy, with 1.1%, 0.9%, 1.3%, and 1.0% in Accuracy, Precision, Recall, and F1 score. "Clustering" strategy also shows positive performance, with 1.2% in Accuracy, 0.7% in Precision, 0.9% in Recall and 0.8% in F1 score.

Conclusion

In this paper, we propose a joint learning approach of leveraging both sequence and image representations to tackle the classification of irregularly sampled clinical time series. By employing our three self-supervised learning strategies, we are able to effectively learn more generalized joint representations. The effectiveness of our approach is verified on three real-world clinical datasets, where it demonstrates superior performance compared to seven state-of-the-art methods. Additionally, we test our approach under more severe missing rates using leave-sensors-out and leave-samples-out techniques. Our approach consistently achieved strong results, demonstrating its robustness in these scenarios. Our code and data will be made publicly available later.

Acknowledgements

We express our sincere gratitude to all the anonymous reviewers for their valuable guidance and suggestions. We also

thank Doctor Weihang Hu, Lin Zhang, and all the colleagues from the Intensive Care Unit at Zhejiang Hospital for their contributions to the expert evaluation and for providing us with valuable clinical advice.

References

Ao, R.; and He, G. 2023. Image based deep learning in 12-lead ECG diagnosis. *Frontiers in Artificial Intelligence*, 5: 1087370.

Brizzi, A.; Whittaker, C.; Servo, L. M.; Hawryluk, I.; Prete Jr, C. A.; de Souza, W. M.; Aguiar, R. S.; Araujo, L. J.; Bastos, L. S.; Blenkinsop, A.; et al. 2022. Spatial and temporal fluctuations in COVID-19 fatality rates in Brazilian hospitals. *Nature medicine*, 28(7): 1476–1485.

Chaudhary, K.; Vaid, A.; Duffy, Á.; Paranjpe, I.; Jaladanki, S.; Paranjpe, M.; Johnson, K.; Gokhale, A.; Pattharanitima, P.; Chauhan, K.; et al. 2020. Utilization of deep learning for subphenotype identification in sepsis-associated acute kidney injury. *Clinical Journal of the American Society of Nephrology*, 15(11): 1557–1565.

Che, Z.; Purushotham, S.; Cho, K.; Sontag, D.; and Liu, Y. 2018. Recurrent neural networks for multivariate time series with missing values. *Scientific reports*, 8(1): 6085.

Chen, T.; Kornblith, S.; Swersky, K.; Norouzi, M.; and Hinton, G. E. 2020. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33: 22243–22255.

Chen, Y.; Ren, K.; Wang, Y.; Fang, Y.; Sun, W.; and Li, D. 2024. Contiformer: Continuous-time transformer for irregular time series modeling. *Advances in Neural Information Processing Systems*, 36.

Choi, E.; Bahadori, M. T.; Schuetz, A.; Stewart, W. F.; and Sun, J. 2016. Doctor ai: Predicting clinical events via recurrent neural networks. In *Machine learning for healthcare conference*, 301–318. PMLR.

Chong, U.-P.; et al. 2011. Signal model-based fault detection and diagnosis for induction motors using features of vibration signal in two-dimension domain. *Strojniški vestnik*, 57(9): 655–666.

Chowdhury, R. R.; Li, J.; Zhang, X.; Hong, D.; Gupta, R. K.; and Shang, J. 2023. Primenet: Pre-training for irregular multivariate time series. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 7184–7192.

de Jong, J.; Emon, M. A.; Wu, P.; Karki, R.; Sood, M.; Godard, P.; Ahmad, A.; Vrooman, H.; Hofmann-Apitius, M.; and Fröhlich, H. 2019. Deep learning for clustering of multivariate clinical patient trajectories with missing values. *GigaScience*, 8(11): giz134.

Deng, Y.; Hua, M.; Yingjun, R.; Fanyue, Q.; and Di, P. 2023. An image characterisation method for AHU fault diagnosis based on residual neural networks. In *Proceedings of Building Simulation 2023: 18th Conference of IBPSA*, volume 18 of *Building Simulation*, 3827–3834. Shanghai, China: IBPSA.

Dey, R.; and Salem, F. M. 2017. Gate-variants of gated recurrent unit (GRU) neural networks. In *2017 IEEE 60th international midwest symposium on circuits and systems (MWSCAS)*, 1597–1600. IEEE.

Gillette, K.; Gsell, M. A.; Nagel, C.; Bender, J.; Winkler, B.; Williams, S. E.; Bär, M.; Schäffter, T.; Dössel, O.; Plank, G.; et al. 2023. MedalCare-XL: 16,900 healthy and pathological synthetic 12 lead ECGs from electrophysiological simulations. *Scientific Data*, 10(1): 531.

Goldberger, A. L.; Amaral, L. A.; Glass, L.; Hausdorff, J. M.; Ivanov, P. C.; Mark, R. G.; Mietus, J. E.; Moody, G. B.; Peng, C.-K.; and Stanley, H. E. 2000. PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals. *circulation*, 101(23): e215–e220.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2020. Generative adversarial networks. *Communications of the ACM*, 63(11): 139–144.

Han, X.; Zheng, H.; and Zhou, M. 2022. Card: Classification and regression diffusion models. *Advances in Neural Information Processing Systems*, 35: 18100–18115.

Harutyunyan, H.; Khachatrian, H.; Kale, D. C.; Ver Steeg, G.; and Galstyan, A. 2019. Multitask learning and benchmarking with clinical time series data. *Scientific data*, 6(1): 96.

Hatami, N.; Gavet, Y.; and Debayle, J. 2018. Classification of time-series images using deep convolutional neural networks. In *Tenth international conference on machine vision (ICMV 2017)*, volume 10696, 242–249. SPIE.

Horn, M.; Moor, M.; Bock, C.; Rieck, B.; and Borgwardt, K. 2020. Set functions for time series. In *International Conference on Machine Learning*, 4353–4363. PMLR.

Huang, J.; Yang, B.; Yin, K.; and Xu, J. 2024. DNA-T: Deformable Neighborhood Attention Transformer for Irregular Medical Time Series. *IEEE Journal of Biomedical and Health Informatics*.

Li, B.; Torr, P. H.; and Lukasiewicz, T. 2022. Clustering generative adversarial networks for story visualization. In *Proceedings of the 30th ACM International Conference on Multimedia*, 769–778.

Li, Z.; Li, S.; and Yan, X. 2024. Time series as images: Vision transformer for irregularly sampled time series. *Advances in Neural Information Processing Systems*, 36.

Lipton, Z. C.; Kale, D. C.; Wetzel, R.; et al. 2016. Modeling missing data in clinical time series with rnns. *Machine Learning for Healthcare*, 56(56): 253–270.

Liu, Z.; Hu, H.; Lin, Y.; Yao, Z.; Xie, Z.; Wei, Y.; Ning, J.; Cao, Y.; Zhang, Z.; Dong, L.; Wei, F.; and Guo, B. 2022. Swin Transformer V2: Scaling Up Capacity and Resolution. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ma, Q.; Li, S.; and Cottrell, G. W. 2020. Adversarial joint-learning recurrent neural network for incomplete time series classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(4): 1765–1776.

Ma, Q.; Zheng, J.; Li, S.; and Cottrell, G. W. 2019. Learning representations for time series clustering. *Advances in neural information processing systems*, 32.

Maroor, J. P.; Sahu, D. N.; Nijhawan, G.; Karthik, A.; Shrivastav, A.; and Chakravarthi, M. K. 2024. Image-Based Time Series Forecasting: A Deep Convolutional Neural Network Approach. In *2024 4th International Conference on Innovative Practices in Technology and Management (ICIPTM)*, 1–6. IEEE.

Miao, X.; Wu, Y.; Wang, J.; Gao, Y.; Mao, X.; and Yin, J. 2021. Generative semi-supervised learning for multivariate time series imputation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, 8983–8991.

Neil, D.; Pfeiffer, M.; and Liu, S.-C. 2016. Phased lstm: Accelerating recurrent network training for long or event-based sequences. *Advances in neural information processing systems*, 29.

Raghu, A.; Chandak, P.; Alam, R.; Guttag, J.; and Stultz, C. 2023. Sequential multi-dimensional self-supervised learning for clinical time series. In *International Conference on Machine Learning*, 28531–28548. PMLR.

Reiss, A.; and Stricker, D. 2012. Introducing a new benchmarked dataset for activity monitoring. In *2012 16th international symposium on wearable computers*, 108–109. IEEE.

Reyna, M. A.; Josef, C. S.; Jeter, R.; Shashikumar, S. P.; Westover, M. B.; Nemati, S.; Clifford, G. D.; and Sharma, A. 2020. Early prediction of sepsis from clinical data: the PhysioNet/Computing in Cardiology Challenge 2019. *Critical care medicine*, 48(2): 210–217.

Sangha, V.; Khunte, A.; Holste, G.; Mortazavi, B. J.; Wang, Z.; Oikonomou, E. K.; and Khera, R. 2024. Biometric contrastive learning for data-efficient deep learning from electrocardiographic images. *Journal of the American Medical Informatics Association*, 31(4): 855–865.

Sangha, V.; Mortazavi, B. J.; Haimovich, A. D.; Ribeiro, A. H.; Brandt, C. A.; Jacoby, D. L.; Schulz, W. L.; Krumholz, H. M.; Ribeiro, A. L. P.; and Khera, R. 2022. Automated multilabel diagnosis on electrocardiographic images and signals. *Nature communications*, 13(1): 1583.

Semenoglou, A.-A.; Spiliotis, E.; and Assimakopoulos, V. 2023. Image-based time series forecasting: A deep convolutional neural network approach. *Neural Networks*, 157: 39–53.

Sood, S.; Zeng, Z.; Cohen, N.; Balch, T.; and Veloso, M. 2021. Visual time series forecasting: an image-driven approach. In *Proceedings of the Second ACM International Conference on AI in Finance*, 1–9.

Tripathy, R.; and Acharya, U. R. 2018. Use of features from RR-time series and EEG signals for automated classification of sleep stages in deep neural network framework. *Biocybernetics and Biomedical Engineering*, 38(4): 890–902.

Wang, J.; Du, W.; Cao, W.; Zhang, K.; Wang, W.; Liang, Y.; and Wen, Q. 2024. Deep learning for multivariate time series imputation: A survey. *arXiv preprint arXiv:2402.04059*.

Wang, Z.; and Oates, T. 2015. Imaging time-series to improve classification and imputation. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, 3939–3945. AAAI Press. ISBN 9781577357384.

Weerakody, P. B.; Wong, K. W.; and Wang, G. 2023. Policy gradient empowered LSTM with dynamic skips for irregular time series data. *Applied Soft Computing*, 142: 110314.

Wu, H.; Hu, T.; Liu, Y.; Zhou, H.; Wang, J.; and Long, M. 2023. TimesNet: Temporal 2D-Variation Modeling for General Time Series Analysis. In *International Conference on Learning Representations*.

Xu, S.; Xu, T.; Yang, Y.; and Chen, X. 2024. Learning metabolic dynamics from irregular observations by Bidirectional Time-Series State Transfer Network. *mSystems*, e00697–24.

Zhang, J.; Zheng, S.; Cao, W.; Bian, J.; and Li, J. 2023. Warpformer: A multi-scale modeling approach for irregular clinical time series. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 3273–3285.

Zhang, X.; Zeman, M.; Tsiligkaridis, T.; and Zitnik, M. 2022. Graph-Guided Network For Irregularly Sampled Multivariate Time Series. In *International Conference on Learning Representations, ICLR*.