A More Details on Time Series Line Graph Image Creation

Implementation. The time-series-to-image transformation can be implemented using the Matplotlib package[1] with the following few lines of code.

```python
def TS2Image(t, v, D, colors, image_height, image_width, grid_height, grid_width):
    import matplotlib.pyplot as plt
    plt.figure(figsize=(image_height/100, image_width/100), dpi=100)
    for d in range(D): # enumerate the multiple variables
        plt.subplot(grid_height, grid_width, d+1) # position in the grid
        # plot line graph of variable d
        plt.plot(t[d], v[d], color=colors[d], linestyle="-", marker="*")
```

In addition to the designs mentioned in the main paper for plotting line graph images, we also explore the following aspects.

Axis Limits of Line Graphs. The axis limits determine the plot area of the line graphs and the range of displayed timestamps and values. By default, we set the limits of the x-axis and y-axis as the ranges of all the observed timestamps and values across the dataset. However, we found that some extreme observed values for some variables can make the range of the y-axis very large, causing most plotted points of observations to cluster in a small area and resulting in flat line graphs. Common normalization and standardization methods will not solve this issue, as the relative magnitudes remain unchanged in the created images. We thus tried the following strategies to remove extreme values and narrow the range of the y-axis:

- Interquartile Range (IQR): IQR is one of the most extensively used methods for outlier detection and removal. The interquartile range is calculated based on the first and third quartiles of all the observed values of each variable in the dataset and then used to calculate the upper and lower limits.
- Standard Deviation (SD): The upper and lower boundaries are calculated by taking 3 standard deviations from the mean of observed values for each variable across the dataset. This method usually assumes the data is normally distributed.
- Modified Z-score (MZ): A z-score measures how many standard deviations away a value is from the mean and is similar to the standard deviation method to detect outliers. However, z-scores can be influenced by extreme values, which modified z-scores can better handle. We set the upper and lower limits as the values whose modified z-scores are 3.5 and -3.5.

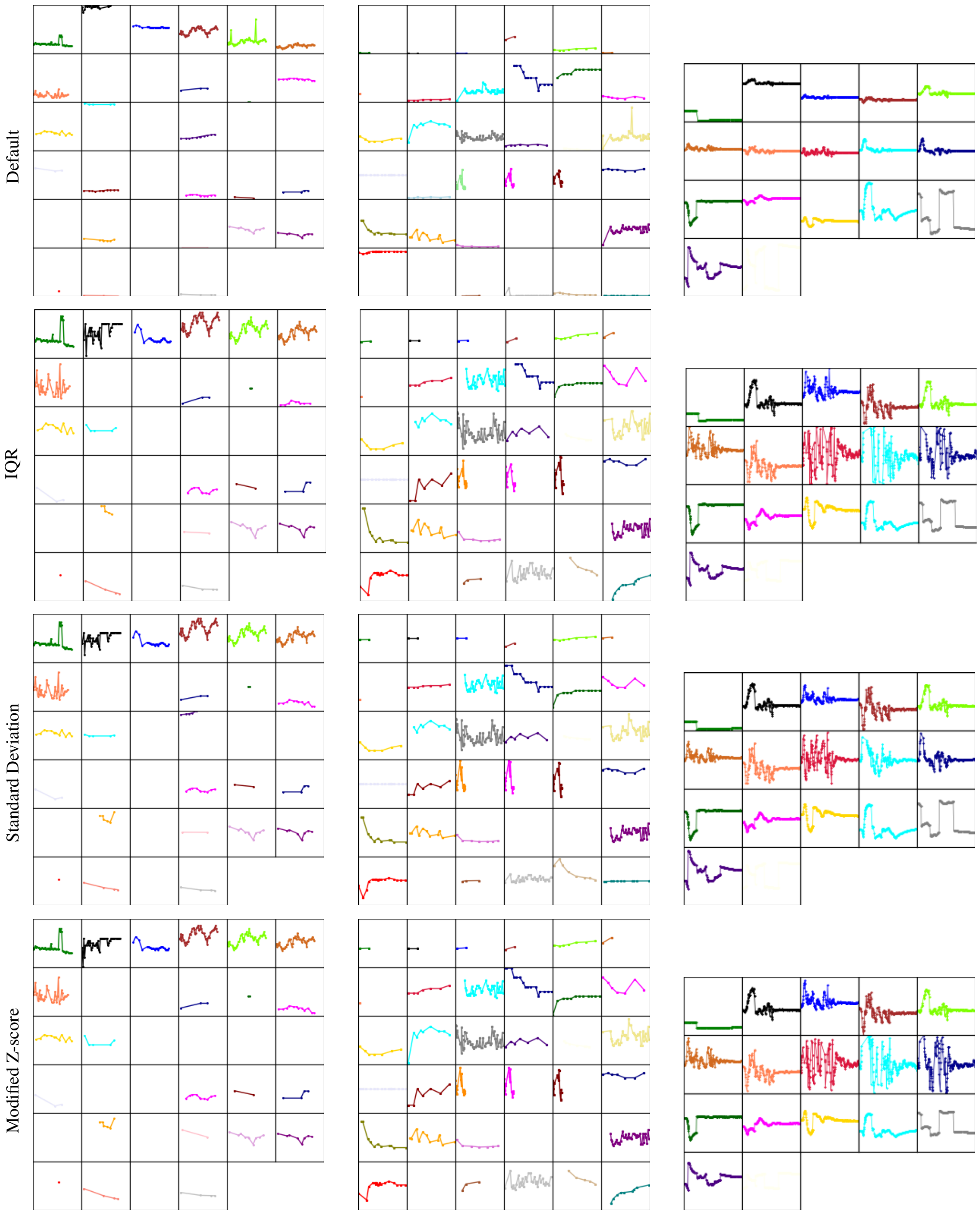
We show examples of the created images with these strategies in Figure 1.

Table 1: Ablation study on different strategies to decide the line graph limit. The default strategy is to directly set the axis limit as the range of all observed values on the dataset. “IQR”, “SD”, and “MZS’ denote three strategies to remove extreme value, *i.e.,* Interqurtile Range, Standard Deviation, and Modified Z-score. The reported numbers are averaged on 5 data splits.

Strategies	P19		P12		PAM			
	AUROC	AUPRC	AUROC	AUPRC	Accuracy	Precision	Recall	F1 score
Default	**89.4** ± 1.9	**52.8** ± 3.8	**85.6** ± 1.1	**49.8** ± 2.5	96.1 ± 0.7	96.8 ± 1.1	96.5 ± 0.7	96.6 ± 0.9
IQR	88.2 ± 0.8	49.6 ± 1.7	84.5 ± 1.1	48.9 ± 2.6	95.9 ± 0.7	96.8 ± 0.7	96.1 ± 0.7	96.4 ± 0.7
SD	87.4 ± 1.6	51.2 ± 3.6	84.6 ± 1.7	47.1 ± 2.9	**96.6** ± 0.9	**97.1** ± 0.8	**97.0** ± 0.6	**97.0** ± 0.7
MZS	87.3 ± 1.0	50.8 ± 3.7	84.3 ± 1.4	47.1 ± 2.1	96.0 ± 1.1	96.8 ± 0.99	96.4 ± 0.9	96.6 ± 0.9

The performance comparison of models trained on images created with different strategies is shown in Table 1. We observe that the methods that remove extreme values hurt the performance, except for SD on the PAM dataset. Although these methods narrow the value range and highlight the dynamic patterns of line graphs, they discard the extreme values which might be informative themselves. This observation suggests that our approach may not require additional data preprocessing on the time series, further demonstrating its advantage in simplicity.

[1] https://matplotlib.org/



(a) P19 (b) P12 (c) PAM

Figure 1: The images created with different strategies for three samples from P19, P12, and PAM dataset, respectively (sample “p000019” for P19, “132548” for P12, and “0” for PAM).

Table 2: Ablation study on grid layouts and image sizes on P19.

Grid Layout	Image Size	AUROC	AUPRC
4 × 9	256 × 576	87.4 ± 1.9	48.1 ± 4.5
5 × 7	320 × 448	87.9 ± 1.9	49.6 ± 2.7
6 × 6	384 × 384	**89.4** ± 1.9	**52.8** ± 3.8
6 × 6	224 × 224	88.7 ± 1.4	52.3 ± 0.6

Table 3: Ablation study on grid layouts and image sizes on P12.

Grid Layout	Image Size	AUROC	AUPRC
4 × 9	256 × 576	84.0 ± 1.4	47.9 ± 2.6
5 × 8	320 × 512	84.1 ± 1.6	47.2 ± 2.3
6 × 6	384 × 384	**85.6** ± 1.1	**49.8** ± 2.5
6 × 6	224 × 224	85.2 ± 2.1	48.8 ± 3.7

Table 4: Ablation study on grid layouts and image sizes on PAM.

Grid Layout	Image Size	Accuracy	Precision	Recall	F1 score
2 × 9	128 × 576	95.9 ± 1.4	96.5 ± 1.0	**95.9** ± 1.2	96.0 ± 0.5
3 × 6	192 × 384	**96.1** ± 0.8	**96.7** ± 0.5	**95.9** ± 0.9	96.2 ± 0.7
4 × 5	256 × 320	**96.1** ± 0.7	**96.8** ± 1.1	96.5 ± 0.7	**96.6** ± 0.9
4 × 5	224 × 224	95.9 ± 0.6	96.7 ± 0.8	**95.9** ± 0.6	96.3 ± 0.7

Grid Layout and Image Size. We conducted experiments to study the impact of grid layouts and image sizes on the performance of our approach. For a fair comparison of different grid layouts, we fixed the size of each grid cell as 64×64 and altered the grid layouts. The results on the P19, P12, and PAM datasets are listed in Table 2, Table 3, and Table 4, respectively. We observed that the square grid layouts consistently produced good results on all three datasets. We conjecture that this is because the square layout ensures that the distance between any two line graphs is shortest. We also tested the performance with the standard image size of 224×224 and found that the differences were marginal, indicating the robustness of our approach to various image sizes.

B More Experimental Details

B.1 Datasets

We used the datasets processed by [7], whose details are given below.

P19: PhysioNet Sepsis Early Prediction Challenge 2019. [2] The P19 dataset [5] consists of clinical data for 38,803 patients, and aims to predict whether sepsis will occur within the next 6 hours. The dataset includes 34 irregularly sampled sensors with 8 vital signs and 26 laboratory values for each patient, as well as 6 demographic features. To process the static features, we use templates outlined in Table 5, and utilize a pre-trained Roberta-base model to extract textual features. These textual features are then combined with visual features obtained from the vision transformer to perform binary classification. The dataset is highly imbalanced with only 4% of samples being positive, and has a missing ratio of 94.9%.

P12: PhysioNet Mortality Prediction Challenge 2012. [3] P12 dataset [2] includes clinical data from 11,988 ICU patients, with 36 irregularly sampled sensor observations and 6 static demographic features provided for each patient. The goal is to predict patient mortality, which is a binary classification task. The dataset is highly imbalanced, with around 86% of samples being negative. The missing ratio of the dataset is 88.4%.

PAM: PAMAP2 Physical Activity Monitoring. [4] The PAM dataset originally contains data of 18 physical activities with 9 subjects wearing 3 inertial measurement units. However, to make it suitable for irregular time series classification, [7] excluded the ninth subject due to its short length of sensor readouts, and 10 out of the 18 activities that had less than 500 samples were also excluded. As a result, the task is an 8-way classification with 5,333 samples, each with 600 continuous observations. To simulate the irregular time series setting, 60% of the observations are randomly removed. No static features are provided, and the 8 categories are approximately balanced. The missing ratio is 60.0%.

[2] https://physionet.org/content/challenge-2019/1.0.0/

[3] https://physionet.org/content/challenge-2012/1.0.0/

[4] https://archive.ics.uci.edu/ml/datasets/pamap2+physical+activity+monitoring

Table 5: Templates for transforming static features to natural language sentences.

Dataset	Static features	Template	Example
P19	*Age*, *Gender*, *Unit1 (medical ICU)*, *Unit2 (surgery ICU)*, *HospAdmTime*; *ICULOS (ICU length-of-stay)*	A patient is {*Age*} years old, {*Gender*}, went to {*Unit1&Unit2*} {*HospAdmTime*} hours after hospital admit, had stayed there for {*ICULOS*} hours.	A patient is 65 years old, female, went to the medical ICU 10 hours after hospital admit, had stayed there for 20 hours.
P12	*RecordID*, *Age*, *Gender*, *Height* (cm), *ICUType*, *Weight* (kg)	A patient is {*Age*} years old, {*Gender*}, {*Height*} cm, {*Weight*} kg, stayed in {*ICUType*}.	A patient is 48 years old, male, 171 cm, 78 kg, stayed in surgical ICU.

Table 6: Ablation studies on different methods to encode static features.

Methods	P19		P12	
	AUROC	AUPRC	AUROC	AUPRC
Raindrop	87.0 ± 2.3	51.8 ± 5.5	82.8 ± 1.7	44.0 ± 3.0
Swin	89.4 ± 1.8	50.2 ± 3.0	84.3 ± 0.6	49.3 ± 3.7
Swin-MLP	88.6 ± 1.3	51.4 ± 3.7	84.6 ± 0.9	48.7 ± 3.2
Swin-Roberta	$\mathbf{89.4} \pm 1.9$	$\mathbf{52.8} \pm 3.8$	85.6 ± 1.1	$\mathbf{49.8} \pm 2.5$

B.2 Experiments on Static Features

Time series data is often associated with information from other modalities, such as the textual clinical notes in electronic health records (EHRs) in the healthcare domain. Our approach is naturally suitable for incorporating such information since we convert time series data to images, and thus various vision-language and multi-modal techniques can be utilized to incorporate the visual (time series) information and information from other modalities. For example, the CLIP [4] learns a shared hidden feature space where the paired image and text stay close. Under our framework, such a shared space can also be learned for the paired visual time series images and textual clinical notes, which is our future direction. It also paves the way for the application of multi-modal models such as GPT-4 [3] to handle the visualized time series data and the clinical notes simultaneously. In our current experiments, we used a text encoder, Roberta-base, to encode textual demographic information in the P19 and P12 datasets. We also experimented with normalizing the original categorical features and encoding them using an MLP as in previous work, and compare with the strong baseline, Raindrop. The results are shown in Table 6. We observe that even without using static features, our method has already outperformed Raindrop. In addition, utilizing Roberta to encode and incorporate the textual feature is more effective than applying MLP over categorical features.

Table 7: The statistics and hyperparameter settings of the evaluated regular multivariate time series datasets.

Datasets	Variables	Classes	Length	Train size	Grid layout	Image size	Learning rate	Epochs
EC	3	4	1,751	261	2 × 2	256 × 256	1e-4	20
UW	3	8	315	120	2 × 2	256 × 256	1e-4	100
SCP1	6	2	896	268	2 × 3	256 × 384	1e-4	100
SCP2	7	2	1,152	200	3 × 3	384 × 384	5e-5	100
JV	12	9	29	270	4 × 4	384 × 384	1e-4	100
SAD	13	10	93	6599	4 × 4	384 × 384	1e-5	20
HB	61	2	405	204	4 × 4	384 × 384	1e-4	100
FD	144	2	62	5890	12 × 12	384 × 384	5e-4	100
PS	**963**	7	144	267	32 × 32	384 × 384	5e-4	100
EW	6	5	**17984**	128	2 × 3	256 × 384	2e-5	100

B.3 Experiment on Regular Time Series

We selected ten representative multivariate time series datasets from the UEA Time Series Classification Archive [1] with diverse characteristics, including the number of classes, variables, and time series length. The datasets we chose are EthanolConcentration (EC), Handwriting (HW), UWaveGestureLibrary (UW), SelfRegulationSCP1 (SCP1), SelfRegulationSCP2 (SCP2), JapaneseVowels (JV), SpokenArabicDigits (SAD), Heartbeat (HB), FaceDetection (FD), PEMS-SF (PS), and EigenWorms

(EW). Notably, the PS dataset has an exceptionally high number of variables (963), while the EW dataset has extremely long time series (17984). These two datasets allow us to assess the effectiveness of our approach when dealing with large numbers of variables and long time series. We applied different image sizes according to the grid layouts for these datasets. The hyperparameter settings are provided in Table 7, and we applied cutout data augmentation methods to SCP1, SCP2, and JV datasets due to the small size of their training sets.

B.4 Self-supervised Learning

We preliminary explored masked image modeling self-supervised pre-training on the time series line graph images. We randomly mask columns of patches with a width of 32 on each line graph within a grid cell. The masking ratio is set as 50%. We finetuned the Swin Transformer model for 10 epochs with batch size 48. The learning rate is 2e-5. Following [6], we use a linear layer to reconstruct the pixel values and employ an ℓ_1 loss on the masked pixels:

$$\mathcal{L} = \frac{1}{\Omega(\mathbf{p}_{\mathrm{M}})} \left\| \hat{\mathbf{p}_{\mathrm{M}}} - \mathbf{p}_{\mathrm{M}} \right\|_1 , \tag{1}$$

where $\mathbf{p}_{\mathrm{M}}$ and $\hat{\mathbf{p}_{\mathrm{M}}}$ are the masked and reconstructed pixels, respectively; $\Omega(\cdot)$ denotes the number of elements.

B.5 Full Experimental Results

We presented the full experimental results in the leave-sensors-out settings in Table 8, and the full results of ablation studies on backbone vision models are presented in Table 9.

Table 8: Full results in the leave-sensors-out settings on PAM dataset. The "missing ratio" denotes the ratio of masked variables.

Missing ratio	Methods	PAM (Leave-**fixed**-sensors-out)				PAM (Leave-**random**-sensors-out)			
		Accuracy	Precision	Recall	F1 score	Accuracy	Precision	Recall	F1 score
10%	Transformer	60.3 ± 2.4	57.8 ± 9.3	59.8 ± 5.4	57.2 ± 8.0	60.9 ± 12.8	58.4 ± 18.4	59.1 ± 16.2	56.9 ± 18.9
	Trans-mean	60.4 ± 11.2	61.8 ± 14.9	60.2 ± 13.8	58.0 ± 15.2	62.4 ± 3.5	59.6 ± 7.2	63.7 ± 8.1	62.7 ± 6.4
	GRU-D	65.4 ± 1.7	72.6 ± 2.6	64.3 ± 5.3	63.6 ± 0.4	68.4 ± 3.7	74.2 ± 3.0	70.8 ± 4.2	72.0 ± 3.7
	SeFT	58.9 ± 2.3	62.5 ± 1.8	59.6 ± 2.6	59.6 ± 2.6	40.0 ± 1.9	40.8 ± 3.2	41.0 ± 0.7	39.9 ± 1.5
	mTAND	58.8 ± 2.7	59.5 ± 5.3	64.4 ± 2.9	61.8 ± 4.1	53.4 ± 2.0	54.8 ± 2.7	57.0 ± 1.9	55.9 ± 2.2
	Raindrop	77.2 ± 2.1	82.3 ± 1.1	78.4 ± 1.9	75.2 ± 3.1	76.7 ± 1.8	79.9 ± 1.7	77.9 ± 2.3	78.6 ± 1.8
	ViTST	**92.7** ± 0.9	**94.2** ± 0.9	**93.2** ± 0.4	**93.6** ± 0.6	**88.4** ± 1.4	**92.3** ± 0.5	**88.6** ± 1.9	**89.8** ± 1.5
20%	Transformer	63.1 ± 7.6	71.1 ± 7.1	62.2 ± 8.2	63.2 ± 8.7	62.3 ± 11.5	65.9 ± 12.7	61.4 ± 13.9	61.8 ± 15.6
	Trans-mean	61.2 ± 3.0	74.2 ± 1.8	63.5 ± 4.4	64.1 ± 4.1	56.8 ± 4.1	59.4 ± 3.4	53.2 ± 3.9	55.3 ± 3.5
	GRU-D	64.6 ± 1.8	73.3 ± 3.6	63.5 ± 4.6	64.8 ± 3.6	64.8 ± 0.4	69.8 ± 0.8	65.8 ± 0.5	67.2 ± 0.0
	SeFT	35.7 ± 0.5	42.1 ± 4.8	38.1 ± 1.3	35.0 ± 2.2	34.2 ± 2.8	34.9 ± 5.2	34.6 ± 2.1	33.3 ± 2.7
	mTAND	33.2 ± 5.0	36.9 ± 3.7	37.7 ± 3.7	37.3 ± 3.4	45.6 ± 1.6	49.2 ± 2.1	49.0 ± 1.6	49.0 ± 1.0
	Raindrop	66.5 ± 4.0	72.0 ± 3.9	67.9 ± 5.8	65.1 ± 7.0	71.3 ± 2.5	75.8 ± 2.2	72.5 ± 2.0	73.4 ± 2.1
	ViTST	**88.4** ± 1.0	**90.4** ± 1.4	**89.3** ± 0.8	**89.7** ± 1.0	**85.1** ± 1.2	**91.1** ± 1.0	**85.6** ± 1.0	**87.0** ± 1.0
30%	Transformer	31.6 ± 10.0	26.4 ± 9.7	24.0 ± 10.0	19.0 ± 12.8	52.0 ± 11.9	55.2 ± 15.3	50.1 ± 13.3	48.4 ± 18.2
	Trans-mean	42.5 ± 8.6	45.3 ± 9.6	37.0 ± 7.9	33.9 ± 8.2	65.1 ± 1.9	63.8 ± 1.2	67.9 ± 1.8	64.9 ± 1.7
	GRU-D	45.1 ± 2.9	51.7 ± 6.2	42.1 ± 6.6	47.2 ± 3.9	58.0 ± 2.0	63.2 ± 1.7	58.2 ± 3.1	59.3 ± 3.5
	SeFT	32.7 ± 2.3	27.9 ± 2.4	34.5 ± 3.0	28.0 ± 1.4	31.7 ± 1.5	31.0 ± 2.7	32.0 ± 1.2	28.0 ± 1.6
	mTAND	27.5 ± 4.5	31.2 ± 7.3	30.6 ± 4.0	30.8 ± 5.6	34.7 ± 5.5	43.4 ± 4.0	36.3 ± 4.7	39.5 ± 4.4
	Raindrop	52.4 ± 2.8	60.9 ± 3.8	51.3 ± 7.1	48.4 ± 1.8	60.3 ± 3.5	68.1 ± 3.1	60.3 ± 3.6	61.9 ± 3.9
	ViTST	**84.1** ± 1.3	**86.5** ± 0.4	**83.1** ± 0.8	**84.9** ± 1.0	**80.6** ± 1.2	**89.5** ± 1.3	**80.9** ± 1.1	**82.6** ± 1.1
40%	Transformer	23.0 ± 3.5	7.4 ± 6.0	14.5 ± 2.6	6.9 ± 2.6	43.8 ± 14.0	44.6 ± 23.0	40.5 ± 15.9	40.2 ± 20.1
	Trans-mean	25.7 ± 2.5	9.1 ± 2.3	18.5 ± 1.4	9.9 ± 1.1	48.7 ± 2.7	55.8 ± 2.6	54.2 ± 3.0	55.1 ± 2.9
	GRU-D	46.4 ± 2.5	64.5 ± 6.8	42.6 ± 7.4	44.3 ± 7.9	47.7 ± 1.4	63.4 ± 1.6	44.5 ± 0.5	47.5 ± 0.0
	SeFT	26.3 ± 0.9	29.9 ± 4.5	27.3 ± 1.6	22.3 ± 1.9	26.8 ± 2.6	2.41 ± 3.4	28.0 ± 1.2	23.3 ± 3.0
	mTAND	19.4 ± 4.5	15.1 ± 4.4	20.2 ± 3.8	17.0 ± 3.4	23.7 ± 1.0	33.9 ± 6.5	26.4 ± 1.6	29.3 ± 1.9
	Raindrop	52.5 ± 3.7	53.4 ± 5.6	48.6 ± 1.9	44.7 ± 3.4	57.0 ± 3.1	65.4 ± 2.7	56.7 ± 3.1	58.9 ± 2.5
	ViTST	**76.5** ± 1.9	**83.5** ± 0.9	**76.7** ± 2.4	**78.3** ± 2.1	**73.7** ± 2.2	**86.4** ± 1.1	**74.0** ± 2.2	**75.8** ± 1.8
50%	Transformer	21.4 ± 1.8	2.7 ± 0.2	12.5 ± 0.4	4.4 ± 0.3	43.2 ± 2.5	52.0 ± 2.5	36.9 ± 3.1	41.9 ± 3.2
	Trans-mean	21.3 ± 1.6	2.8 ± 0.4	12.5 ± 0.7	4.6 ± 0.2	46.4 ± 1.4	59.1 ± 3.2	43.1 ± 2.2	46.5 ± 3.1
	GRU-D	37.3 ± 2.7	29.6 ± 5.9	32.8 ± 4.6	26.6 ± 5.9	49.7 ± 1.2	52.4 ± 0.3	42.5 ± 1.7	47.5 ± 1.2
	SeFT	24.7 ± 1.7	15.9 ± 2.7	25.3 ± 2.6	18.2 ± 2.4	26.4 ± 1.4	23.0 ± 2.9	27.5 ± 0.4	23.5 ± 1.8
	mTAND	16.9 ± 3.1	12.6 ± 5.5	17.0 ± 1.6	13.9 ± 4.0	20.9 ± 3.1	35.1 ± 6.1	23.0 ± 3.2	27.7 ± 3.9
	Raindrop	46.6 ± 2.6	44.5 ± 2.6	42.4 ± 3.9	38.0 ± 4.0	47.2 ± 4.4	59.4 ± 3.9	44.8 ± 5.3	47.6 ± 5.2
	ViTST	**70.0** ± 2.7	**79.9** ± 2.2	**70.5** ± 3.1	**72.2** ± 3.0	**70.9** ± 1.2	**83.6** ± 2.4	**71.5** ± 1.4	**73.3** ± 2.1

Table 9: Full results of our approach with different backbone vision models and the compared baselines. **Bold** indicates the best performer, while underline represents the second best.

Methods	P19		P12		PAM			
	AUROC	AUPRC	AUROC	AUPRC	Accuracy	Precision	Recall	F1 score
Transformer	80.7 ± 3.8	42.7 ± 7.7	83.3 ± 0.7	47.9 ± 3.6	83.5 ± 1.5	84.8 ± 1.5	86.0 ± 1.2	85.0 ± 1.3
Trans-mean	83.7 ± 1.8	45.8 ± 3.2	82.6 ± 2.0	46.3 ± 4.0	83.7 ± 2.3	84.9 ± 2.6	86.4 ± 2.1	85.1 ± 2.4
GRU-D	83.9 ± 1.7	46.9 ± 2.1	81.9 ± 2.1	46.1 ± 4.7	83.3 ± 1.6	84.6 ± 1.2	85.2 ± 1.6	84.8 ± 1.2
SeFT	81.2 ± 2.3	41.9 ± 3.1	73.9 ± 2.5	31.1 ± 4.1	67.1 ± 2.2	70.0 ± 2.4	68.2 ± 1.5	68.5 ± 1.8
mTAND	84.4 ± 1.3	50.6 ± 2.0	84.2 ± 0.8	48.2 ± 3.4	74.6 ± 4.3	74.3 ± 4.0	79.5 ± 2.8	76.8 ± 3.4
IP-Net	84.6 ± 1.3	38.1 ± 3.7	82.6 ± 1.4	47.6 ± 3.1	74.3 ± 3.8	75.6 ± 2.1	77.9 ± 2.2	76.6 ± 2.8
DGM^2-O	86.7 ± 3.4	44.7 ± 11.7	84.4 ± 1.6	47.3 ± 3.6	82.4 ± 2.3	85.2 ± 1.2	83.9 ± 2.3	84.3 ± 1.8
MTGNN	81.9 ± 6.2	39.9 ± 8.9	74.4 ± 6.7	35.5 ± 6.0	83.4 ± 1.9	85.2 ± 1.7	86.1 ± 1.9	85.9 ± 2.4
Raindrop	87.0 ± 2.3	51.8 ± 5.5	82.8 ± 1.7	44.0 ± 3.0	88.5 ± 1.5	89.9 ± 1.5	89.9 ± 0.6	89.8 ± 1.0
ResNet	76.3 ± 3.3	34.7 ± 4.1	72.9 ± 1.0	28.8 ± 2.4	73.1 ± 0.9	82.4 ± 5.6	69.7 ± 0.9	71.4 ± 1.8
ViT	87.9 ± 2.5	51.6 ± 3.7	84.8 ± 1.3	48.1 ± 3.8	93.4 ± 0.7	94.7 ± 0.9	94.1 ± 0.7	94.3 ± 0.7
Swin	**89.4** ± 1.9	**52.8** ± 3.8	**85.6** ± 1.1	**49.8** ± 2.5	**96.1** ± 0.7	**96.8** ± 1.1	**96.5** ± 0.7	**96.6** ± 0.9
Swin-scratch	74.6 ± 2.5	29.9 ± 4.6	66.9 ± 1.6	26.5 ± 2.6	84.5 ± 0.5	86.6 ± 0.6	87.1 ± 1.2	86.6 ± 0.6

References

[1] Bagnall, A., Dau, H. A., Lines, J., Flynn, M., Large, J., Bostrom, A., Southam, P., and Keogh, E. The uea multivariate time series classification archive, 2018. *arXiv preprint arXiv:1811.00075*, 2018.

[2] 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. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *circulation*, 101(23): e215–e220, 2000.

[3] OpenAI. Gpt-4 technical report, 2023.

[4] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

[5] Reyna, M. A., Josef, C., Seyedi, S., Jeter, R., Shashikumar, S. P., Westover, M. B., Sharma, A., Nemati, S., and Clifford, G. D. Early prediction of sepsis from clinical data: the physionet/computing in cardiology challenge 2019. In *2019 Computing in Cardiology (CinC)*, pp. Page–1. IEEE, 2019.

[6] Xie, Z., Zhang, Z., Cao, Y., Lin, Y., Bao, J., Yao, Z., Dai, Q., and Hu, H. Simmim: A simple framework for masked image modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9653–9663, 2022.

[7] Zhang, X., Zeman, M., Tsiligkaridis, T., and Zitnik, M. Graph-guided network for irregularly sampled multivariate time series. In *International Conference on Learning Representations*, 2022.