doi:10.3969/j.issn.1001-4616.2025.02.010

# 基于 IFS-LCT-ViT 的时间序列分类方法

杨思栋[1],王　珂[1],刘　兵[1],苏　冰[2]

(1.中国矿业大学计算机科学与技术学院,江苏 徐州 221000)
(2.河北广电网络集团秦皇岛有限公司,河北 秦皇岛 066000)

[**摘要**]　目前针对时间序列分类问题,大多采用一维视角进行分析. 二维视角下的时间序列具有更高量级的数据,但相关的研究较少且基本为格拉姆角场法(Gramian Angular Field,GAF)和卷积神经网络模型的组合. 文中将对图像视角下的时间序列分类进行深入研究,对目前方法存在的相关问题进行优化. 首先解决 GAF 算法的计算冗余问题,提出不平衡因子法(imbalance factor subtraction,IFS),以基础运算替换 GAF 的三角运算,在不损失分类精度的情况下,减少了图像生成过程的运算. 其次针对卷积类模型存在局部偏好的问题,文中将图像识别的任务交给视觉全自注意力网络(Vision Transformer,ViT),通过对时序转换图分割,再对分割后的子块以全局并行计算的方式分配注意力权重,得到图像的整体特征. 最后,提出适配 ViT 模型的轻量卷积令牌(lightweight convolutional token,LCT),通过一维卷积提取原始序列的局部特征,来弥补 ViT 模型对图像简单硬分割所带来的信息损失. 结合以上所有提出了 IFS-LCT-ViT 模型,为了验证模型的有效性,在 UCR 官网中的 11 个数据集上进行了实验. 结果表明,该模型与 GRU-FCN、TST、GAF-CNN、XCM、OSCNN、MultiRocket 相比,在 6 个数据集上获得了 85.9%、80.2%、68.2%、63.0、85.3%和 84.0%的最高准确率,证明了该模型在时间序列分类任务上的有效性.

[**关键词**]　时间序列分类,图像视角,不平衡因子,视觉自注意力网络,轻量卷积令牌
[**中图分类号**]TP391　[**文献标志码**]A　[**文章编号**]1001-4616(2025)02-0091-11

# Time Series Classification Method Based on IFS-LCT-ViT

Yang Sidong[1],Wang Ke[1],Liu Bing[1],Su Bing[2]

(School of Computer Science and Technology,China University of Mining and Technology,Xuzhou 221000,China)
(2.Hebei Television Broadcasts and Information Network Group Qinhuangdao Co,Ltd,Qinhuangdao 066000,China)

**Abstract**:Currently,most of time series classification problems are analyzed from a one-dimensional perspective. Time series in a two-dimensional perspective have higher levels of data,which brings more space for research exploration,but there are fewer related studies. And the studies are basically a combination of Gramian Angular Field(GAF) and convolutional neural network models. In this paper,time series classification from the image perspective will be studied in depth. The related problems existing in current method will be optimized. Firstly,the Imbalance Factor Subtraction(IFS) method is proposed to solve the computational redundancy problem of the GAF algorithm. It replaces the trigonometric operation of GAF with the basic operation,which reduces the operation of the image generation process without losing the classification accuracy. Secondly,aiming at the problem of local preference in convolutional class models,the task of image recognition is given to the Vision Transformer(ViT). The overall features of the image are obtained by segmenting the temporal transition graph and then assigning the attention weight in the same way to all segmented subblocks. Finally, a lightweight convolutional token(LCT) adapted to ViT is proposed to extract the local features of the original sequence through one-dimensional convolution to compensate for the information loss caused by the simple hard segmentation of the image by ViT. Combining all of the above,IFS-LCT-ViT is proposed,and to verify the validity of the model,experiments were carried out on 11 datasets on the UCR website. Experimental results show that compared with GRU-FCN,TST,GADF-CNN,XCM,OSCNN and MultiRocket,the model obtains the highest accuracy of 85.9%,80.2%,68.2%,63.0,85.3% and 84.0% on six datasets,which proves the effectiveness of the model in time series classification tasks.

**Key words**: time series classification, image perspective, imbalance factor subtraction, vision transformer, lightweight convolution token

**收稿日期**:2023-05-03.
**基金项目**:国家自然科学基金项目(62276266).
**通讯作者**:王珂,博士,副教授,研究方向:虚拟现实矿山技术、无线传感器网络. E-mail:wangke@ cumt.edu.cn

目前,时间序列分类问题可分为基于传统机器学习的分类方法和基于神经网络的深度学习方法. 前者大多以人为方式构造特征,在表征时序系统状态时具备很强的可解释性,而后者以多层神经网络进行特征提取,随着网络的加深提取到高层特征,此种特征通常无法解释,但可以识别到数据中的额外特征.

传统机器学习依赖于最近邻[1](K nearest neighbors,KNN)聚类算法,关键在于时间序列之间的距离度量. 针对时间序列距离度量的有欧式距离[2](euclidean distance,ED)和动态规整距离[3](dynamic time warping,DTW),UCR 官方将此类方法作为基线方法.

其次是深度学习领域的研究,2018 年,Elsayed 等[4]提出的门控循环全卷积网络(gate recurrent unit-fully convolutional network,GRU-FCN),利用循环神经网络对时间序列的记忆信息进行提取,同时使用全卷积神经网络获取时间序列的局部特征,二者结合的方式使得模型把握了更加丰富的信息. 2021 年,Zerveas 等[5]提出基于全自注意力的时序分类模型(transformer-based time,TST),利用注意力机制对重点片段进行捕捉,在 Heartbeat、InsectWingbeat 等数据集上效果明显. 2021 年,Fauvel 等[6]提出了一种可解释的卷积神经网络(Explainable Convolutional Neural Network,XCM)用于时间序列分类,能够更加精确地识别出对分类贡献大的输入区. 2022 年,Tang 等[7]提出了全方位卷积神经网络(OMNI-SCALE CNNS,OSCNN),有效解决了卷积神经网络模型对于卷积核大小的选择问题. MultiRocket 是 Tan 等[8]在 2022 年对 Rocket 模型[9]的改进,与 Rocket 使用随机卷积核的方式不同,MultiRocket 使用一组较小且固定的卷积核,并增加了额外变换,其效率更高,同时获得了更高的准确率.

以上模型都是一维视角下的分类模型,Wang 等[10]提出的一种将时间序列转换为图像的格拉姆角场(Gramian Angular Field,GAF)使得图像视角下的时间序列分类模型开始出现,GAF 结合卷积类神经网络也是目前图像视角下时间序列分类的普遍方案. 2020 年,庞新宇等[11]融合 GAF 和卷积神经网络(convolution neural network,CNN)对行星齿轮箱进行故障诊断,克服了浅层特征对故障信号表征能力不足的问题. 2021 年 8 月姜家国等[12]基于 GAF 和密集连接卷积网络(DenseNet)实现了针对滚动轴承的故障诊断方法,在凯斯西储大学轴承数据集上的表现优异.

可以看出,在时间分类问题的解决上,主流方案依旧是一维视角下的模型,图像视角下的研究相对较少. 此外,通过在时间序列分类领域官方的 UCR 数据集上进行实验,发现 GAF 结合 CNN 的方式与先进的一维模型相比稍有逊色. 为了提高图像视角下时间序列分类模型的竞争力,将对 GAF-CNN 方法进行以下两点改进. (1)对 GAF 方法优化,提出不平衡因子法,降低了时间序列转换为图像的时间. 本文删去了 GAF 中的不必要运算,以此降低时间序列转换为图像的未简化时间复杂度,并以引入两个超参数的方式来丰富转换后的图像特征,以此来保证分类准确率. (2)提高了图像视角下的时间序列分类的准确率. 通过对图像识别模块进行替换,由原本的 CNN 类模型替换为视觉自注意力模型(vision transformer,ViT),以注意力的形式对不同子块分配权重,使得模型关注更为关键的信息,进而提高时序图像的分类精度. 此外,文中在视觉自注意力模型中设计了一个轻量级的卷积令牌组件,来进一步提高模型的分类性能.

# 1　相关技术

## 1.1　格拉姆角场法

格拉姆角场法是目前将一维时间序列转换为二维图像使用最为广泛的方法. 使用格拉姆角场法将时间序列转换为图像的步骤如下:

Step 1:给定时间序列 $\boldsymbol{X}=\{x_1,x_2,\cdots,x_n\}$,将序列进行最大最小值标准化,使得序列中的所有数据都落在[0,1]区间内.

$$\tilde{x}_i=\frac{x_i-\min(\boldsymbol{X})}{\max(\boldsymbol{X})-\min(\boldsymbol{X})}. \tag{1}$$

Step 2:将标准化后的时间序列映射到极坐标系,角度记为 $\phi$,半径记作 $r$,$t$ 为时间戳,$N$ 为正则化极坐标的跨度常数因子.

$$\begin{cases}\varphi_i=\arccos(\tilde{x}_i), & -1\leqslant\tilde{x}_i\leqslant 1,\tilde{x}_i\in\tilde{x}\\ r=\dfrac{t_i}{N}, & t_i\in N\end{cases} \tag{2}$$

Step 3:计算极坐标系下两两元素差值的正弦值,并将其作为 GAF 图像的像素值.

$$\boldsymbol{G}_{ij}=\sin(\varphi_i-\varphi_j) \tag{3}$$

GAF 方法伪代码的实现如下:

**算法 1:格拉姆角场,时间复杂度 $O(n^2)+3O(n)$**

输入:时间序列 $X=\{x_1,x_2,\cdots,x_n\}$
输出:图像 $Y(i,j)=\{y_{00},y_{01},\cdots,y_{nn}\}$
1 $X_\min\leftarrow\mathrm{Min}(X)$
2 $X_\max\leftarrow\mathrm{Max}(X)$
3 for $i=1,2,\cdots,n$ do
4 $\quad X_\cos_i\leftarrow(x_i-X_\min)/(X_\max-X_\min)$
5 $\quad X_\sin_i\leftarrow1-X_\cos_i^2$
6 end for
7 for $i=1,2,\cdots,n$ do
8 $\quad$ for $j=1,2,\cdots,n$ do
9 $\quad\quad y_{ij}\leftarrow X_\sin_i\cdot X_\cos_j-X_\cos_i\cdot X_\sin_j$
10 $\quad$ end for
11 end for
12 return $Y$

其中,行 1 与行 2 分别是计算时间序列的最小值与最大值,时间复杂度均是 $O(n)$. 行 3 至行 6 经过一次循环,时间复杂度为 $O(n)$. 行 7 至行 11 是双层循环,时间复杂度为 $O(n^2)$. 所以利用 GAF 将时间序列转换为图像的时间复杂度为 $O(n^2)+3O(n)$.

## 1.2 视觉全自注意力网络

视觉全自注意力网络[13](ViT)是 Transformer 架构用于图像领域的尝试,模型可以分为以下 3 层.

### 1.2.1 嵌入层

嵌入层为编码层的注意力分配做准备,需要将图像分割为固定大小的子块. 具体操作是将图像各个通道的长和宽分别除以 $P$,图像将被分割为 $N$ 个子图块. 如图 1 所示,$H\times W\times C$ 的图像将会被分割为 $H\times W/P^2$ 个分辨率为 $P\times P\times C$ 的子图块.

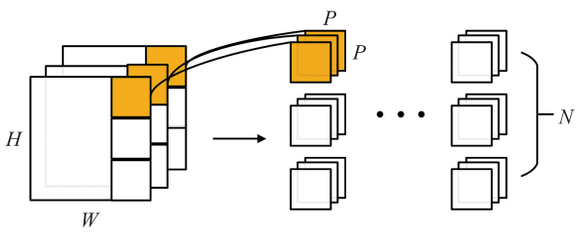



**图 1 分割示意图**

**Fig. 1 Segmentation diagram**

其次,为了区分各个子块的位置,对 1-$N$ 个位置分别编码. 具体的做法是创建与每个子块维度相同的可训练参数矩阵,将参数矩阵与子块特征进行相加,在网络迭代过程中,自主学习位置信息.

### 1.2.2 编码层

编码层是整个模型的核心部分,由 $L$ 个 Transformer Encoder Block 堆叠组成. Transformer Encoder Block 结构如图 2 所示.

其中,Transformer Encoder Block 的核心是 Multi Head Attention,旨在对子块序列分配注意力,在捕捉全局信息的同时加大关键信息的权重.

设输入到 Multi Head Attention 的子块序列为$[P_1,P_2,\cdots,P_i]$,对每一个子块进行 3 次线性变换,将特征维度变换至 $d_k$,分别记作查询向量 $\boldsymbol{Q}$(to match others),键值向量 $\boldsymbol{K}$(to be matched),属性值向量 $\boldsymbol{V}$(to be extracted),则 $P_i$ 的自注意力得分可由以下公式获得:

$$\begin{aligned}&Attention(P_i)=\frac{\mathrm{e}^{q_i}}{\sum_{j=1}^{N}\mathrm{e}^{q_i}}\times[V_1,V_2,\cdots,V_N],\\&q_i=\frac{Q_i\times K^T}{\sqrt{d_k}}.\end{aligned} \tag{4}$$

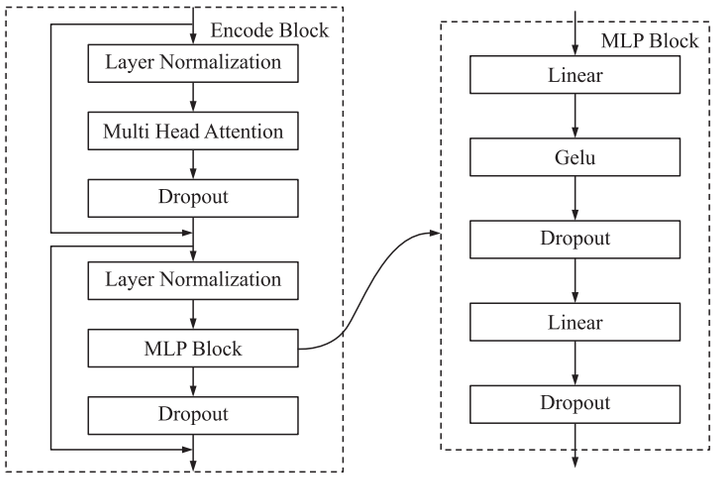



**图 2　Transformer encoder block 结构图**

**Fig. 2　Transformer encoder block structure diagram**

在得到 $P_i$ 的自注意力得分后,$P_i$ 的 $h$ 头注意力就可以由以下公式计算得出:

$$\begin{aligned} &MultiHead(P_i)=[head_1,head_2,\cdots,head_n]\times W_h,\\ &head_i=Attention(P_i). \end{aligned} \tag{5}$$

1.2.3　输出层

模型在经过多个编码层后,各个子块的多头注意力已经被分配完毕. 输出层(MLP Head)由 Linear 层,Tanh 激活函数,Linear 层组成,用来对所有子块的信息进行聚合,得到该图像属于某个类别的得分,最终把得分最高的类别作为模型的预测类别.

# 2　IFS-LCT-ViT 方法

## 2.1　不平衡因子法

不平衡因子法(imbalance factor subtraction,IFS)是对格拉姆角场法的简化,格拉姆角场需要计算元素的反余弦值,在计算完反余弦值后还需计算反余弦值两两差值之间的正弦值,而本文移除了生成过程中的三角函数操作,取而代之的是引入两个不平衡因子.

Step 1:给定时间序列 $\boldsymbol{X}=\{x_1,x_2,\cdots,x_n\}$,直接计算两两元素之间的差值,得到差值矩阵 $\boldsymbol{G}$.

$$\boldsymbol{G_{ij}}=x_i-x_j \tag{6}$$

Step 2:引入不平衡因子 $\alpha$ 和 $\beta$,使得对角线对称的矩阵变得不平衡,以此获取更加丰富的特征.

$$\boldsymbol{G_{ij}}=(G_{ij}-\alpha)/\beta \tag{7}$$

Step 3:将矩阵对角线元素的常数项填充,进一步丰富图像特征.

$$\boldsymbol{G_{ij}}=\begin{cases} G_{ij}, & i\neq j\\ G_{ij}+\dfrac{\alpha}{\beta}+x_i, & i=j \end{cases} \tag{8}$$

IFS 方法伪代码的实现如下:

**算法 2:不平衡因子,时间复杂度 $\boldsymbol{O(n^2)}$**

输入:时间序列 $X=\{x_1,x_2,\cdots,x_n\}$,因子 $\alpha,\beta$

输出:图像 $Y(i,j)=\{y_{00},y_{01},\cdots,y_{nn}\}$

```
for i=1,2,…,n do
    for j=1,2,…,n do
        y_ij ← x_i − x_j
        y_ij ← (y_ij − α)/β
        if(i=j)
            y_ij ← y_ij + x_j + α/β
```

```
        end for
    end for
    return Y
```

可以看出,方法只涉及一个双层循环,所以时间复杂度仅为 $O(n^2)$,与 GAF 方法的 $O(n^2)+3O(n)$ 相比存在优势. 此外,GAF 方法所转换的图像,关于对角线互为相反数且对角线上的元素值均为 0,在神经网络中属于重复特征,IFS 通过引入不平衡因子和对对角线元素填充的方式丰富了这一部分特征.

## 2.2 轻量卷积令牌

从 1.2.1 节中可以看出,ViT 模型以简单硬分割的方式将时序转换图分为不同的子块,这种硬分割的方式使得原本关联的局部信息处于不同的子块之中,使得模型难以对这些原本关联的局部特征进行建模.

一种解决思路是将硬分割改为卷积的滑动操作[14],不同的子块中将具有重叠部分,使得局部信息可以存放在同一个子块中,但这样会使得子块的数量激增,导致模型的运算量大大增加.

本文对原本时间序列进行一维卷积,并将卷积结果作为 ViT 的分类令牌,利用分类令牌完成序列局部特征的提取,来弥补图像的局部信息损失. 这种方法不仅可以提高模型的局部表现力,计算量上与利用卷积滑动产生子块的方法相比也大幅减少.

卷积令牌为模型引入了额外参数量,为减少这一部分对模型的影响,尽可能使一维卷积令牌模块轻量化,本文利用一维深度可分离卷积层来代替普通一维卷积层. 一维深度可分离卷积层结构如图 3 所示.

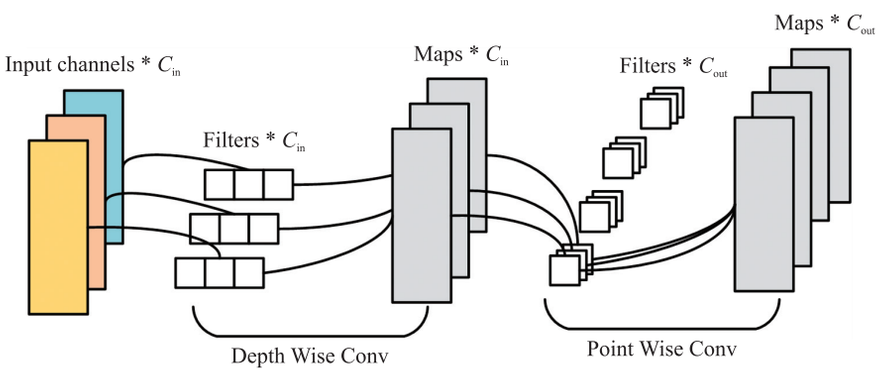



**图 3 一维深度可卷积结构图**

**Fig. 3 Depth convolutionable structure diagram**

从图 3 中可以看出,一维深度可分离卷积层分为 Depth Wise 卷积阶段和 Point Wise 卷积阶段. 在 Depth Wise 阶段,使用分组卷积技术,组数设置为输入的通道数 $C$,即每个通道为一组进行卷积,设卷积尺寸为 $k$,则 Depth Wise 的参数量为 $k\times C$. 在 Point Wise 阶段,卷积核的尺寸置为 1,设输出通道设置为 $C_1$,Point Wise 的参数量为 $C\times C_1$. 若使用普通卷积层,参数量为 $k\times C\times C_1$,所以减少的参数量为 $C(k\times C_1-k-C_1)$.

在一维深度可分离卷积层的基础上,轻量卷积令牌模块(lightweight convolutional token,LCT)的整体结构如图 4 所示. 首先是 3 层 DPConv Block 串联,随着卷积层数的叠加,模型的感受也随之增大. 每个 DPConv Block 又由一维深度可分离卷积层、归一化层、激活层组成. 其次,将卷积层的输出输入到自适应平均池化层,不同长度的时间序列在经过平均池化层后会池化到指定长度. 最后将通道维度与特征维度进行重组,得到与 ViT 子块大小相同的令牌.

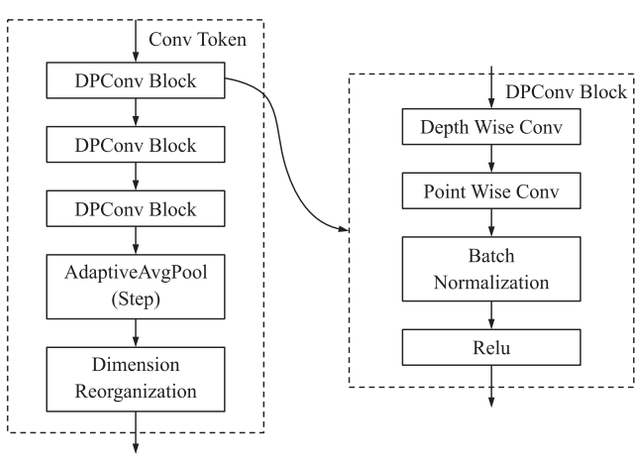



**图 4 卷积令牌模型结构图**

**Fig. 4 Structure diagram of the convolutional token model**

其中,自适应平均池化层的步长设置为 $P\times P/C$,其中 $P$ 为 ViT 中 Patch 大小,$C$ 是最后

一个 DPConv Block 的输出通道维数,最后由通道重组模块将自适应平均池化层输出 $C\times\text{Step}$ 重组为 $P\times P$.

### 2.3　模型整体架构

以 1.2 节中 ViT 的模型结构为基础,则不平衡因子模块应置于 ViT 的位置编码步骤前,即将时间序列转换为图像后再进行分割与编码操作. 而轻量卷积令牌模块所生成的子块应与图像子块一同输入到编码层中,则该模块应与图像生成和位置编码两步骤同时进行. 最终,IFS-LCT-ViT 模型的整体结构如图 5 所示.

具体步骤如下:

Step 1:利用 IFS 方法将时间序列维度进行重塑,将其转换为图像.

Step 2:对图像进行分割并对每个位置进行编码.

Step 3:利用轻量卷积令牌模块将时间序列转换为与图像子块大小相同的令牌,并嵌在所有子块之前.

Step 4:将令牌与子块输入到 Transformer Encode Blocks 中.

Step 5:将编码层连接到输出层,得到时间序列的预测类别.

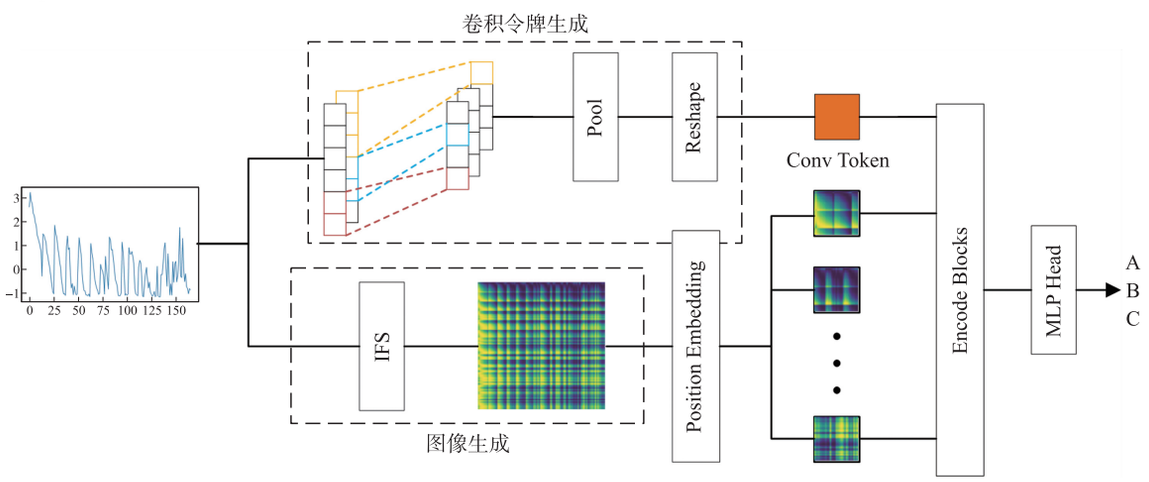



**图 5　IFS-LCT-ViT 模型结构图**

**Fig. 5　IFS-LCT-ViT model structure diagram**

## 3　实验与结果

### 3.1　数据集及评价指标

UCR 数据集是时间序列分类的经典数据集,其收集了来自包括医疗、运动、光谱等不同领域的 128 个时间序列集合. UCR 官方给出了基于 ED 和 DTW 距离度量的 1-NN 分类准确率,作为该数据集的基准准确率. 本文选取了其中的 11 个数据集进行实验,各数据集具体信息如表 1 所示.

**表 1　数据集详细信息表**

**Table 1　The dataset details table**

| 数据集名称 | 训练集 | 测试集 | 类别数 | 长度 | 欧式距离 | 动态规整距离 |
|---|---|---|---|---|---|---|
| ChlorineConcentration | 467 | 3 840 | 3 | 166 | 0.65 | 0.65 |
| DistalPhalanxOutlineAgeGroup | 400 | 139 | 3 | 80 | 0.63 | 0.77 |
| DistalPhalanxOutlineCorrect | 600 | 276 | 2 | 80 | 0.72 | 0.72 |
| DistalPhalanxTW | 400 | 139 | 6 | 80 | 0.63 | 0.59 |
| MedicalImages | 381 | 760 | 10 | 99 | 0.68 | 0.74 |
| MiddlePhalanxOutlineAgeGroup | 400 | 154 | 3 | 80 | 0.52 | 0.50 |
| MiddlePhalanxOutlineCorrect | 600 | 291 | 2 | 80 | 0.77 | 0.70 |
| MiddlePhalanxTW | 399 | 154 | 6 | 80 | 0.51 | 0.51 |
| PhalangesOutlinesCorrect | 1 800 | 858 | 2 | 80 | 0.76 | 0.73 |
| ProximalPhalanxTW | 400 | 205 | 6 | 80 | 0.71 | 0.76 |
| ItalyPowerDemand | 67 | 1 029 | 2 | 24 | 0.96 | 0.95 |

为了验证各个模型的有效性,本文沿用 UCR 数据集官方的准确率作为评价指标,准确率($Acc$)的计算公式如公式(9):

$$Acc=\frac{TP+TN}{TP+TN+FP+FN},\tag{9}$$

$TP$ 代表正样本预测为正,$TN$ 代表正样本预测为负,$FN$ 代表负样本预测为负,$FP$ 代表负样本预测为正.

### 3.2 实验环境与参数配置

本文的实验使用图形处理器(GPU, graphics processing unit)并结合统一计算架构(CUDA, Compute Unified Device Architecture)来加速运算. 实验具体的硬件环境和软件环境如表 2 所示.

表 2 实验运行环境

Table 2 The experimental operating environment

| 实验环境 | 条目 | 版本 |
|---|---|---|
| 硬件环境 | System | Windows 11 |
| | GPU | GTX 3060 |
| 软件环境 | Python | 3.7.10 |
| | Pytorch | 1.7.0 |
| | CUDA | 11.0 |

模型参数配置:首先是不平衡因子法中的两个不平衡因子,文中设置 $\alpha$ 和 $\beta$ 均为 0.5. 其次是轻量卷积令牌模块参数,3 个卷积层的卷积核大小设置为[3,5,7],通道数设置为[128,256,128],自适应平均池化层的步长设置为 8. 之后是 ViT 的参数设置,子块大小设置为 32 * 32,自注意力特征维度设置为 512,MLP Head 特征维度设置为 1 024,模型深度设置为 6. 最后是模型优化器的学习率设置为 1e-5.

训练参数配置:训练最大迭代轮次设置为 1 000,批处理大小为 25,每个 epoch 后都进行一次测试,测试集准确率连续 50 个轮次不再提高停止训练. 优化器使用 Adam 优化器,学习率使用 1e-5,损失函数使用交叉熵损失函数.

### 3.3 对比实验

#### 3.3.1 不平衡因子法有效性验证

(1)时间效率验证,IFS 与 GAF 方法的转换时间如表 3 所示,表中黑体字表示最优的性能.

表 3 运行时间对比

Table 3 Runtime comparison

ms

| 名称 | 训练集 | | | 测试集 | | |
|---|---|---|---|---|---|---|
| | GAF | IFS | 改进 | GAF | IFS | 改进 |
| CC | 49 | **41** | 16.33% | 457 | **434** | 5.03% |
| DPOAG | 13 | **9** | 30.77% | 6 | **4** | 33.33% |
| DPOC | 25 | **11** | 56.00% | 10 | **8** | 20.00% |
| DPTW | 11 | **9** | 18.18% | 4 | **3** | 25.00% |
| MI | 14 | **10** | 28.57% | 40 | **27** | 32.50% |
| MPOAG | 12 | **7** | 41.67% | 5 | **3** | 40.00% |
| MPOC | 15 | **10** | 33.33% | 9 | **5** | 44.44% |
| MPTW | 11 | **6** | 45.45% | 5 | **3** | 40.00% |
| POC | 47 | **35** | 25.53% | 24 | **17** | 29.17% |
| PPTW | 12 | **7** | 41.67% | 8 | **5** | 37.50% |
| IPD | 0.5 | **0.2** | 60.00% | 6 | **4** | 33.33% |
| 平均改进 | | | 31.64% | | | 30.94% |

从表中数据可得,IFS 方法与 GAF 方法相比,在 11 个数据集的训练集上平均提高了 31.64%,在测试集上平均提高了 30.94%,表明 IFS 在时间效率上具有优势. 虽然 IFS 与 GAF 一样,计算过程大体上都分为 3 个步骤,但不平衡因子法的二三步骤在代码实现时,可以嵌入到第一个步骤的双层循环中,从而降低时间复杂度,运行时间也自然较短.

(2)准确率验证,利用 CNN[12] 和 ViT 作为图像识别模型,各个数据集上的准确率如表 4 所示,表中黑体字表示最优的性能.

从表中数据可得,在 11 个数据集上,以 CNN 作为图像识别模型,IFS 方法获得了 6 次最高准确率,高于 GAF 的 5 次,IFS 方法有轻微的优势. 而在以 ViT 作为图像识别模型上,IFS 获得了 7 次最高准确率,优于 GAF 的 4 次,优势更为明显. 由此可得在分类准确率上,IFS 也要优于 GAF 方法,原因在于 IFS 方法所转换的图像在相同分辨率的情况下,为网络增加了额外特征.

表 4　GAF 与 IFS 准确率对比

Table 4　GAF versus IFS accuracy

| 数据集名称 | CNN | | ViT | |
|---|---|---|---|---|
| | GAF | IFS | GAF | IFS |
| CC | **0.794** | 0.781 | 0.872 | **0.898** |
| DPOAG | 0.794 | **0.806** | 0.77 | **0.771** |
| DPOC | 0.793 | **0.822** | 0.797 | **0.804** |
| DPTW | **0.724** | 0.712 | **0.727** | 0.716 |
| MI | 0.755 | **0.763** | 0.763 | **0.772** |
| MPOAG | **0.617** | 0.604 | 0.649 | **0.652** |
| MPOC | **0.860** | 0.859 | **0.866** | 0.843 |
| MPTW | **0.592** | 0.584 | **0.636** | 0.624 |
| POC | 0.839 | **0.848** | 0.828 | **0.838** |
| PPTW | 0.808 | **0.829** | **0.839** | 0.834 |
| IPD | 0.947 | **0.968** | 0.97 | **0.973** |
| 最大值次数 | 5 | **6** | 4 | **7** |

此外,从表中数据可以得出,在 CNN 与 ViT 的对比中,ViT 模型获得了 8 次最高准确率,高于 CNN 的 3 次,表明时序转换图的识别上,ViT 模型要优于 CNN 模型. 原因在于卷积神经网络使用卷积核的堆叠来获取全图感知,但实际上随着神经网络层数的加深,梯度传递产生困难,容易陷入局部偏好,而局部偏好适配自然图像,在时序转换图上作用有限.

### 3.3.2　轻量卷积令牌有效性验证

在 IFS 和 ViT 的基础上,为了验证轻量卷积令牌的有效性,在 11 个数据集上将 IFS-ViT 与 IFS-LCT-ViT 进行对比,实验数据如表 5 所示,表中黑体字表示最优的性能. 从表中数据可得,使用轻量级的卷积令牌与不使用相比,模型在 9 个数据集上获得了更高准确率,原因在于 ViT 的硬分割切断了图像的内部联系,卷积令牌的存在提取了原始序列中的局部特征,对模型所提取的特征进行了补充. 实验数据表明,本文设计的轻量卷积令牌在大多数数据集上对模型做出了改进,证明了该模块的有效性.

表 5　令牌有效性对比

Table 5　Token vilidity comparison

| 数据集名称 | IFS-ViT | IFS-LCT-ViT | 改进 |
|---|---|---|---|
| CC | **0.898** | 0.859 | −0.039 |
| DPOAG | 0.771 | **0.802** | 0.031 |
| DPOC | 0.804 | **0.809** | 0.005 |
| DPTW | 0.716 | **0.727** | 0.011 |
| MI | 0.772 | **0.805** | 0.033 |
| MPOAG | 0.652 | **0.682** | 0.03 |
| MPOC | 0.843 | **0.859** | 0.016 |
| MPTW | 0.624 | **0.630** | 0.006 |
| POC | 0.838 | **0.853** | 0.015 |
| PPTW | 0.834 | **0.840** | 0.006 |
| IPD | **0.973** | 0.971 | −0.002 |
| 最大值次数 | 2 | **9** | |

### 3.3.3　模型整体有效性验证

为了验证模型整体的有效性,选取了目前时间序列分类领域的先进模型作为对比模型. 选取的模型有 GRU-FCN(2018)、TST(2020)、GAF-CNN(2021)、XCM(2021)、OSCNN(2022)、MultiRocket(2022). 在 11 个数据集上,本模型与其他 6 个模型准确率如表 6 所示,表中黑体字表示最优的性能.

表 6　各模型准确率对比

Table 6　Accuracy comparison of each model

| 名称 | 模型 | | | | | | |
|---|---|---|---|---|---|---|---|
| | GRU-FCN | TST | GAF-CNN | XCM | OSCNN | MultiRocket | Ours |
| CC | 0.790 | 0.740 | 0.794 | 0.850 | 0.834 | 0.746 | **0.859** |
| DPOAG | 0.782 | 0.778 | 0.794 | 0.778 | 0.786 | 0.774 | **0.802** |
| DPOC | 0.790 | 0.787 | 0.793 | 0.795 | **0.812** | 0.811 | 0.809 |
| DPTW | **0.747** | 0.718 | 0.724 | 0.724 | **0.747** | 0.724 | 0.727 |
| MI | 0.801 | 0.707 | 0.755 | 0.705 | 0.804 | **0.807** | 0.805 |
| MPOAG | 0.658 | 0.660 | 0.617 | 0.652 | 0.666 | 0.639 | **0.682** |
| MPOC | 0.832 | 0.841 | 0.860 | 0.838 | 0.847 | **0.863** | 0.860 |
| MPTW | 0.612 | 0.610 | 0.592 | 0.624 | 0.611 | 0.589 | **0.630** |
| POC | 0.824 | 0.827 | 0.839 | 0.831 | 0.848 | 0.852 | **0.853** |
| PPTW | 0.833 | 0.824 | 0.808 | 0.832 | 0.833 | 0.818 | **0.840** |
| IPD | 0.956 | **0.973** | 0.947 | 0.967 | 0.955 | 0.967 | 0.971 |
| 最大值次数 | 1 | 1 | 0 | 0 | 2 | 2 | **6** |

从表中数据可知,在所有模型对比上,IFS-LCT-ViT 在 7 个模型中,11 个数据集上,以 6 次最高准确率明显高于其他模型的 1、1、0、0、2 和 2 次,在 DPOAG,DPOAG,MPOAG,MPOC,MPTW,POC 数据集上准确率达到了 85.9%、80.2%、68.2%、63.0%、85.3%和 84.0%,这充分证明了 IFS-LCT-ViT 在时间序列分类问题上的优越性,有效提高了以图像分类模型为基础的时间序列分类方法的竞争力.

为了进一步验证 IFS-LCT-ViT 的性能,本小节引入了自定义的平均性能评价指标. 将文中各模型的平均排名得分作为度量标准,平均排名得分的计算公式如公式(10)所示.

$$Score(i)=N-\frac{\sum_{n=1}^{N}Rank_n(i)}{N}, \tag{10}$$

其中 $N$ 为数据集个数,$Rank_n(i)$ 代表模型 $i$ 在第 $n$ 个数据集上的排名,当模型的平均排名越高时,分数也会相应变多,各个模型的平均排名得分如图 6 所示.

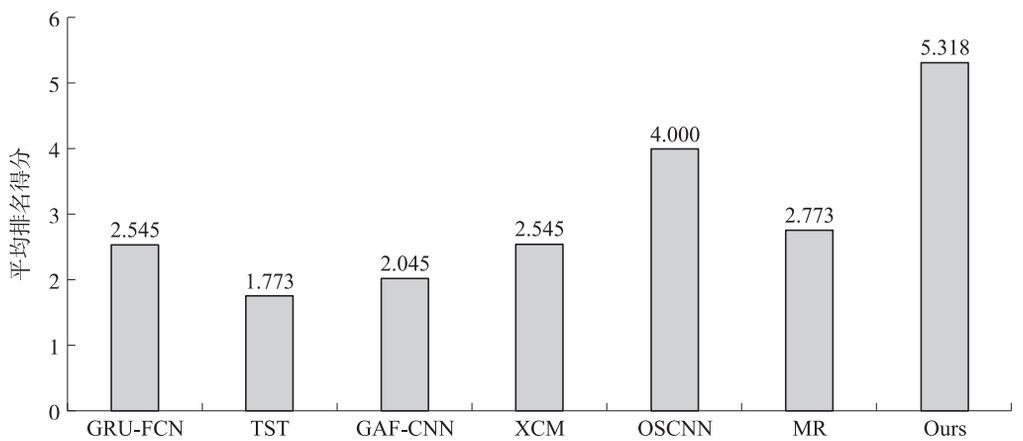



**图 6　各模型平均排名得分**

**Fig. 6　Average ranking scores for each model**

从图中可得,IFS-LCT-ViT 得分为 5.318,高于其他模型的 2.545、1.773、2.045、2.545、4.0 和 2.773,表明模型的平均性能也足够优秀.

在图像视角下,神经网络所需处理的数据量会更大,在推理效率上会弱于一维视角模型. 但为了分类性能的提升,一定的效率损失是可以接受的,各模型在整个测试集上所需的推理时间如图 7 所示.

从图中可以看出,在整个测试集的推理上,IFS-LCT-ViT 模型的效率要优于二维视角下的 GAF-CNN 模型和一维视角下的 GRU-FCN 模型,弱于其他几个一维视角模型. 但时间上的差距在 1 s 以内,在实际推理中,为了分类性能的提升,此等差距处于可接受范围之内.

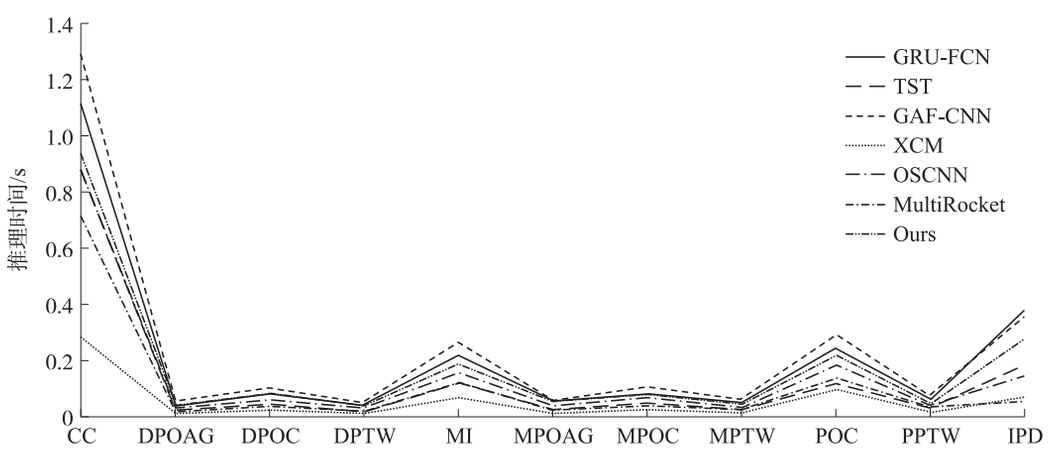



**图 7　各模型推理时间**

**Fig. 7　Inference time for each model**

### 3.4　参数实验

本部分将对模型优化器中的学习率以及不平衡因子生成法中的两个不平衡因子进行实验.

#### 3.4.1　学习率实验

学习率是模型训练一个重要的超参数,取学习率为 1e-4,1e-5 和 1e-6 对模型进行实验,实验数据如表 7 所示.

从表 7 数据可知,当学习率取 1e-4,1e-5 和 1e-6 时,分别可以在 11 个数据集上获得 2,8,2 次最高准确率. 由此可得,对于该模型,1e-5 是更具普适性的准确率. 以 MI 数据集为例,对训练过程进行可视化,实验结果如图 8 所示.

表 7　学习率对比

Table 7　Token comparison

| 数据集名称 | 1e-4 | 1e-5 | 1e-6 |
|---|---|---|---|
| CC | 0.849 | **0.859** | 0.817 |
| DPOAG | 0.799 | **0.802** | 0.784 |
| DPOC | 0.808 | **0.809** | 0.804 |
| DPTW | **0.727** | **0.727** | 0.698 |
| MI | 0.775 | **0.805** | 0.795 |
| MPOAG | 0.662 | **0.682** | 0.662 |
| MPOC | 0.839 | **0.859** | 0.849 |
| MPTW | 0.623 | 0.630 | **0.636** |
| POC | 0.851 | **0.853** | 0.851 |
| PPTW | 0.834 | 0.840 | **0.849** |
| IPD | **0.972** | 0.971 | 0.968 |
| 最大值次数 | 2 | **8** | 2 |

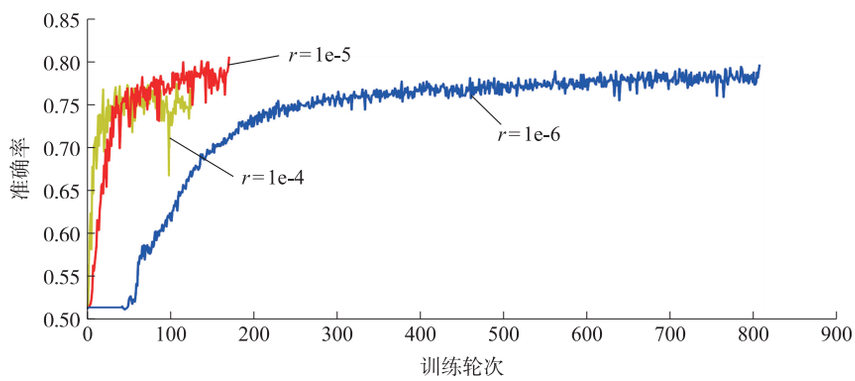



图 8　不同学习率实验

Fig. 8　Experiments with different learning rates

从图中可以看出,当学习率为 1e-4 时,收敛速度最快,但训练过程非常不稳定,使得模型在最优值附近摇摆,难以收敛到最佳值. 当学习率为 1e-6 时,训练过程相对而言较为稳定,但模型的训练轮次达到了 800 以上,大大增加了训练时长,同时太小的步长容易使得模型陷入局部最优. 当学习率为 1e-5 时, 训练轮次较 1e-4 没有明显增加,同时也获得了最优的准确率,表明 1e-5 是较为合适的学习率.

#### 3.4.2　平衡因子实验

本小节对 $\alpha$ 和 $\beta$ 两个不平衡因子进行实验,当 $\beta$ 取 0.5 时,不同 $\alpha$ 取值的实验结果如图 9 所示;$\alpha$ 取 0.5 时,不同 $\beta$ 取值的实验结果如图 10 所示.

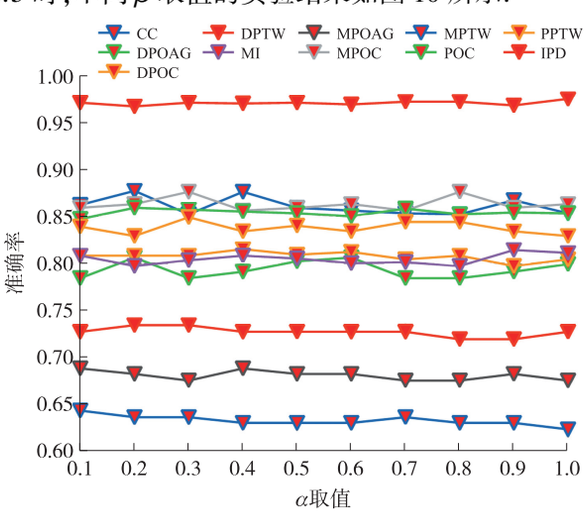



图 9　不同 $\alpha$ 实验

Fig. 9　Experiments with different $\alpha$

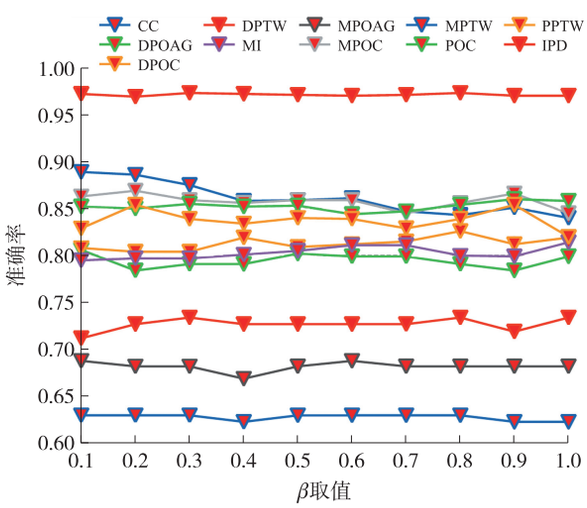



图 10　不同 $\beta$ 实验

Fig. 10　Experiments with different $\beta$

从图中数据可知,当衰减因子和缩放因子取默认值 0.5 时,在不同数据集上并不是最优表现,说明该方法具有一定的发展潜力. 衰减因子和缩放因子决定了图像关于对角线的偏差程度以及图像整体的对比度,不同参数的组合对不同数据集可识性的提升存在差异. 因此,若想充分发挥此种方法在准确率方向上的优势,可以通过实验去寻找更优解.

## 4 结论

本文提出了 IFS-LCT-ViT 模型,该模型具有多方面的优势. 首先,IFS 方法在降低时间复杂度的同时,为模型也带来了更高的准确率. 其次,文中选用 ViT 作为图像分类模型,与 CNN 模型相比,该模型更适合时序转换图的识别. 此外,文中设计了轻量卷积令牌作为 ViT 模型硬分割图像的补偿,进一步提高了模型的分类性能. 通过在 UCR 的 11 个公开数据集上进行的实验,与当前先进的几个模型对比,IFS-LCT-ViT 优势明显. 但对于不同数据集不平衡因子的选择尚未找到通用办法,只能通过大量的实验去找到一个最佳的组合,未来将对不平衡因子的自适应选取进行研究,以此在不同的数据集上都能获得最佳的表现.

[参考文献]


[1] PETERSON L E. K-nearest neighbor[J]. Scholarpedia,2009,4(2):1883.

[2] ABANDA A,MORI U,LOZANO J A. A review on distance based time series classification[J]. Data mining and knowledge discovery,2019,33(2):378-412.

[3] MÜLLER M. Dynamic time warping[M]. Information Retrieval for Music and Motion. Berlin,Heidelberg:Springer,2007:69-84.

[4] ELSAYED N,MAIDA A S,BAYOUMI M. Deep gated recurrent and convolutional network hybrid model for univariate time series classification[J]. [2018-12-18]. https://arxiv.org/pdf/1812.07683.pdf.

[5] ZERVEAS G,JAYARAMAN S,PATEL D,et al. A transformer-based framework for multivariate time series representation learning[C]//Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. Online. 2021:2114-2124.

[6] FAUVEL K, LIN T, MASSON V, et al. Xcm: An explainable convolutional neural network for multivariate time series classification[J]. Mathematics,2021,9(23):3137.

[7] TANG W, LONG G, LIU L, et al. Omni-Scale CNNs: a simple and effective kernel size configuration for time series classification[C]//ICLR 2022-10th International Conference on Learning Representations. Online. 2022.

[8] TAN C W, DEMPSTER A, BERGMEIR C, et al. Multirocket: multiple poding operators and transformations for fast and effective time series classification[J]. Data mining and knowledge discovery,2022,36(5):1623-1646.

[9] DEMPSTER A,PETITJEAN F,WEBB G I. ROCKET:exceptionally fast and accurate time series classification using random convolutional kernels[J]. Data mining and knowledge discovery,2020,34(5):1454-1495.

[10] WANG Z,OATES T. Imaging time-series to improve classification and imputation[C]//Proceedings of the 24th International Conference on Artificial Intelligence. Bunenos Aires:Argentina. 2015:3939-3945.

[11] 庞新宇,仝钰,魏子涵. 一种 GAF-CNN 行星齿轮箱故障诊断方法[J]. 北京理工大学学报,2020,40(11):1161-1167.

[12] 姜家国,郭曼利,杨思国. 基于 GAF 和 DenseNet 的滚动轴承故障诊断方法[J]. 工矿自动化,2021,47(08):84-89.

[13] DOSOVITSKIY A,BEYER L,KOLESNIKOV A,et al. An image is worth 16x16 words:Transformers for image recognition at scale[J]. [2020-10-22]. https://arxiv.org/pdf/2010.11929.pdf.

[14] YUAN L, CHEN Y, WANG T, et al. Tokens-to-token vit: Training vision transformers from scratch on imagenet[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. Online. 2021:558-567.


[责任编辑:陆炳新]