

# 自表示稀疏一维操作自动编码器进行高光谱图像波段选择任务

## 摘要

考虑到高光谱图像 (HSI) 数据处理中的波段选择对计算复杂性和准确性的影响, 波段选择是一项重要任务。在这项工作中, 我们提出了一种针对频带选择问题的新颖框架: 带有稀疏一维操作自动编码器 (SOA) 的自表示学习 (SRL)。所提出的 SLR-SOA 方法引入了一种新颖的自动编码器模型 SOA, 该模型旨在学习稀疏表示数据的表示域。此外, 该网络由具有非线性神经元模型的一维操作层组成。因此, 浅层架构大大提高了神经元 (过滤器) 的学习能力。使用紧凑的架构在自动编码器中尤其重要, 因为由于其恒等映射目标, 它们很容易过度拟合。总体而言, 我们表明, 考虑到所实现的土地覆盖分类精度, 所提出的 SRL-SOA 波段选择方法在包括 Indian Pines 和 Salinas-A 在内的两个 HSI 数据上优于竞争方法。

**关键词:** 波段选择、高光谱图像、自我表示学习、稀疏自动编码器

引文的 bib 文件统一粘贴到 **refs.bib** 中并采用如下引用方式 [?]

## 1 引言

高光谱成像传感器能够捕捉数百种不同波长的观察场景。因此, 这些光学传感器提供了有关目标的丰富光谱信息, 并且已被用于许多应用, 例如目标检测 [13]、土地覆盖分类 [3,8]、人脸识别 [12] 和医学成像 [9]。然而, 如此丰富的光谱信息在高光谱图像 (HSI) 数据处理中引入了与计算时间复杂度和内存相关的一些缺点和限制。

由于维数灾难, 多项 HSI 分类研究 [4,11] 表明, 所需的训练样本数量随着频段数量呈指数增长。这也被称为 HSI 数据中的休斯现象 [5]。因此, 波段选择过程对于减少分类框架中对更多训练样本的需求、标签成本和整体计算复杂性起着至关重要的作用。[14,17] 中提出了各种频段选择策略。基于自表示学习 (SRL) 的方法旨在使用所有波段的线性组合来表示 HSI 数据; 本质上, 在这种表示中获得的频带系数将决定相应频带的重要性。在基于稀疏表示的频带选择 (SpaBS) 方法 [17] 中, 他们设置了一个约束即 HSI 数据的表示在变换域中是稀疏的, 相应的非稀疏系数仅表示最具描述性的频带。在 [3] 中, 提出了高效图卷积自表示 (EGCSR) 用于频带选择。因此, 他们使用图卷积设计了 EGCSR 模型, 其中通过将每个频带视为图上非欧几里得域中的节点来扩展传统的自我表示模型。最后, [10] 中提出了一种子空间聚类方法, 即改进的稀疏子空间聚类 (ISSC) 技术。与基于“1-最小化”的一般 SpaBS 方法相反, ISSC 方法使用“2-最小化”以避免解决方案过于稀疏并考虑频带内可能的相关性。

稀疏自动编码器 [2, 15] 已用于不同的表示学习应用中。这些无监督网络学习将给定输入映射到隐藏表示空间，其中表示向量稀疏但仍然具有足够的描述性，可以通过网络的解码器部分在输出层重建原始输入。稀疏性是通过在隐藏层的激活上应用不同的正则化技术（例如“1”、“2”或 k-稀疏正则化）来实现的。然而，现有的自动编码器有一个共同的限制：由于线性卷积运算，网络只能提供有限的非线性映射。人们可以增加网络深度来学习更复杂的非线性变换函数。另一方面，考虑到自动编码器的简单身份映射目标，增加层数（网络深度）可能会以直接的方式导致过度拟合。

在本研究中，我们提出了一种新颖的自动编码器模型：稀疏一维操作自动编码器 (SOA)。接下来，我们设计了一种新颖的频段选择框架，称为稀疏一维操作自动编码器的自我表示学习 (SRL-SOA)。尽管稀疏自动编码器常用于表示学习，但它们在 SRL 任务中尚未得到充分发现。因此，我们在这项工作中表明，与传统的 SRL 方法相比，这些正则化无监督网络可用于提高频带选择性能。此外，为了解决传统自动编码器的上述局限性，所提出的 SOA 网络由自组织一维操作层和生成神经元模型组成，该模型可以对每个内核元素执行任何非线性变换。在该神经元模型中，非线性变换函数通过泰勒级数展开来近似，模型的权重（可训练参数）是近似函数系数。结果表明，在一些分类和去噪应用中，与卷积神经网络 (CNN) 相比，具有封装生成神经元的 2D 操作层 [6, 7, 10] 的自组织操作神经网络 (Self-ONN) 取得了更高的性能水平）和传统的线性神经元模型仅执行卷积。在这项研究中，所提出的 SOA 能够以有效的方式学习 SRL 问题的复杂非线性变换，而不需要深层网络架构并最终导致过度拟合。所提出的 SRL-SOA 方法的性能评估已在包括 Indian Pines [1] 和 Salinas-A [16] 在内的两个 HSI 数据集上进行，结果表明，与 SpaBS 相比，所提出的方法实现了卓越的频带选择性能水平、EGCSR、ISSC 和主成分分析 (PCA) 方法。

## 2 相关工作

现有的 BS 方法可以分为有监督方法 [7] 和无监督方法 [10]。考虑到获得足够标签的难度，不使用任何样本标签的无监督 BS 比有监督 BS 更加灵活和适用。最近，已经提出了许多无监督 BS 方法，这些方法可以分为四类：排序、聚类、搜索和基于稀疏性的方法。基于排名的方法旨在使用预定义的标准评估每个频段的重要性，以选择排名靠前的频段。基于聚类的方法首先将波段分为几个簇，然后从簇中选择高度不相关的波段。在基于搜索的方法中，HSI 的 BS 被视为组合优化问题，通常可以使用启发式搜索方法来解决。基于稀疏性的方法假设每个频带可以由几个其他频带稀疏地表示。估计的稀疏系数可以揭示 HSI 波段内的某些基础结构，并用于 BS。此外，一些深度学习技术已被用于 HSI BS 方法，强化学习技术已被用于 HSI BS 方法；然而，这些方法仅利用 HSI 的统计特性，而忽略了其内在的结构信息。

### 2.1 自编码器

在自动编码器框架中，首先以特定的参数化封闭形式显式定义特征提取函数。这个函数，我们将表示  $f_\theta$ ，称为编码器，它将允许从输入  $x$  直接有效地计算特征向量  $h = f_\theta(x)$ ，对于来自数据集  $\{x^{(1)}, \dots, x^{(t)}\}$  的每个实例  $x^{(t)}$  我们定义：

$$h^{(t)} = f_\theta(x^{(t)}),$$

其中  $h^{(t)}$  是根据  $x^{(t)}$  计算的特征向量或表示或代码。另一个封闭形式参数化函数  $g_\theta$  称为解码器，从特征空间映射回输入空间，产生重建。概率模型是根据显式概率函数定义的，并经过训练以最大化（通常近似）数据似然（或代理），而自动编码器通过其编码器和解码器进行参数化，并使用不同的训练原理进行训练。编码器和解码器的参数集在重建任务以及可能的原始输入中同时学习，即尝试产生尽可能最低的重建误差与其重构之间差异的度量——在训练样本上。良好的泛化意味着测试示例的重构误差较低，而大多数其他  $x$  配置的重构误差较高。因此，为了捕获数据生成分布的结构，重要的是训练标准或参数化中的某些内容会阻止自动编码器学习恒等函数，该函数在任何地方都具有零重构误差。这是通过不同形式的自动编码器中的各种手段来实现的，如下文更详细地描述的，我们将这些称为正则化自动编码器。正则化的一种特殊形式包括将代码限制为低维度，这就是经典自动编码器或 PCA 所做的事情。

总之，基本的自动编码器训练包括找到最小化重建误差的参数向量值：

$$J_{AE}(\theta) = \sum_t L(x^{(t)}, g_\theta(f_\theta(x^{(t)}))),$$

其中  $x^{(t)}$  是一个训练示例。这种最小化通常是通过 MLP 训练中的随机梯度下降来实现的。由于自动编码器主要是作为 MLP 开发的，预测其输入，因此编码器和解码器最常用的形式是仿射映射，可选地跟随非线性：

$$f_\theta(x) = s_f(b + Wx),$$

$$g_\theta(h) = s_g(d + W'h),$$

其中  $s_f$  和  $s_g$  是编码器和解码器激活函数（通常是元素 sigmoid 或双曲正切非线性，或者恒等函数，如果保持线性），该模型的参数集为  $\theta = \{W, b, W', d\}$  其中  $b$  和  $d$  被称为编码器和解码器偏置向量， $W$  和  $W'$  是编码器和解码器权重矩阵。

## 2.2 稀疏自编码器

最早使用单层自动编码器通过堆叠来构建深层架构，考虑了将编码器权重和解码器权重联系起来以限制容量的想法以及引入稀疏正则化形式的想法。表示的稀疏性可以通过惩罚隐藏单元偏差（使这些附加偏移参数更负）来实现或直接惩罚隐藏单元激活的输出（使它们更接近 0 处的饱和值）。惩罚偏差会带来权重补偿偏差的危险，这可能会损害数值优化。当直接惩罚隐藏单元输出时，文献中可以找到几种变体，但仍然缺乏明确的比较分析。尽管 L1 惩罚（即，在 sigmoid 非线性情况下，简单地求和输出元素  $h_j$ ）似乎是最自然的（因为它在稀疏编码中使用），但它在涉及稀疏自动编码器的论文中很少使用。

## 2.3 自表示学习

由于冗余波段的自表示性质，对于给定的样本矩阵  $X$ ，SRL 将每个波段表示为其他波段（包括其自身）的线性组合，可以建模如下：

$$X^T = X^T W + E$$

其中变量  $W \in \mathbb{R}^{b \times b}$  and  $E \in \mathbb{R}^{n \times b}$  分别是表示系数和残差矩阵。矩阵  $E$  的第  $i$  行表示训练样本  $X_i$  的重建误差（RE）。为了避免在模型中得到平凡的解（例如， $W = I$  和  $E = 0$ ），有必

要对  $W$  进行正则化。因此，SRL 的正式定义可以表示如下：

$$\min_W \sum_{i=1}^n \mathbb{L}_i + \iota R(W)$$

其中  $\mathbb{L}_i = \mathbb{L}(E^i)$  and  $E^i = X_i^T - X_i^T W$  第一项是损失函数，例如最小二乘或平方损失函数，以最小化 RE。第二项是正则化项，以避免琐碎的解决方案和引导带选择。低秩和行稀疏约束是两个常见的正则化术语。通过交叉验证确定的正参数 用于实现第一项和第二项之间的平衡。

### 3 本文方法

#### 3.1 本文方法概述

在本节中，我们首先定义频带选择问题的稀疏自表示的目标。然后，提出一维操作层，然后是 SRL-SOA 模型。

#### 3.2 稀疏自我表征

给定 HSI 立方体  $X = \{x_i\}_{i=1}^N \in \mathbb{R}^{M \times N}$  由  $M$  个样本和  $N$  个波段组成， $X$  的稀疏自表示模型可以定义如下，

$$X = XA + N,$$

其中  $A, N \in \mathbb{R}^{N \times N}$  分别是稀疏表示系数矩阵和表示误差。因此，假设每个频带可以由其他频带的线性组合来表示。可以得到  $A$  的稀疏解，例如，求解下面的 0-范数约束问题：

$$\min_A \|A\|_0 \text{ s.t. } XA + N = X, \text{diag}(A) = 0,$$

其中  $\text{diag}(A) = 0$  用于防止每个频带由其自身表示的平凡解决方案。一般来说，SRL 的方法使用最接近的范数松弛，即“1-范数”，因为它更有效且更实用。

#### 3.3 一维操作层

自组织操作层由生成神经元组成，这些神经元能够使用泰勒级数展开来近似每个一维核元素的非线性核变换。因此，泰勒级数可以在函数  $f(\cdot)$  的原点附近表示为：

$$f(z) = \sum_{q=0}^{\infty} \frac{f^{(q)}(0)}{q!} (z)^q.$$

那么，对于  $Q$  阶近似，变换函数如下：

$$y^{(k)} = \sigma \left( \sum_{q=1}^Q (x)^q * w_q^{(k)} + b_q^{(k)} \right),$$

其中  $*$  和  $(\cdot)$  分别是一维卷积运算和激活函数（双曲正切）， $b_q$  是偏差。总体而言，建议层的第  $k$  个神经元具有以下可训练参数  $\Theta_k = \{W^{(k)} \in \mathbb{R}^{f_s \times Q}, b^{(k)} \in \mathbb{R}^Q\}$ 。

### 3.4 使用稀疏 1DO 操作自动编码器进行自我表示学习

在所提出的 SOA 网络中, 编码器部分由单个 1D 操作层组成, 该操作层具有  $L$  个滤波器 (生成神经元), 具有  $\Theta_{cn} = \{\Theta_k\}_{k=1}^L$  个可训练参数。因此, 对于批量采样的 HSI 立方体  $X_s = [A_1, A_2, \dots, A_m] \in \mathbb{R}^{m \times N}$  在所提出的网络的重建阶段, 我们逐像素地应用自表示。具体来说, 解码器部分执行以下操作  $X_s = \psi(X_s, A_s)$  重建批量采样的 HSI 立方体  $X_s$ :

$$X_s = \begin{bmatrix} l_{x_1} & x_2 & \dots & x_m \end{bmatrix} \begin{bmatrix} A_1 \\ A_2 \\ \dots \\ A_m \end{bmatrix}.$$

然后, 通过一批获得的多个表示矩阵的绝对平均值获得最终的表示系数矩阵  $A = 1/m \sum_{i=1}^m |A_i|$ . 总体而言, 在提出的 SRL-SOA 框架中, SOA 模型经过训练, 使用 ADAM 优化来最小化以下成本。

$$\mathbb{L}(\Theta_{cn}, X_s, \hat{X}_s) = \frac{1}{2} \|X_s - \hat{X}_s\|_2^2 + \lambda \|A\|_1, \\ s.t. \text{diag}(A) = 0.$$

其中  $\lambda$  是稀疏性和数据保真度部分之间权衡的正则化参数。请注意, 解码器部分没有任何可训练参数, 因为它应用了重建。SRL-SOA 框架如图 1 所示:

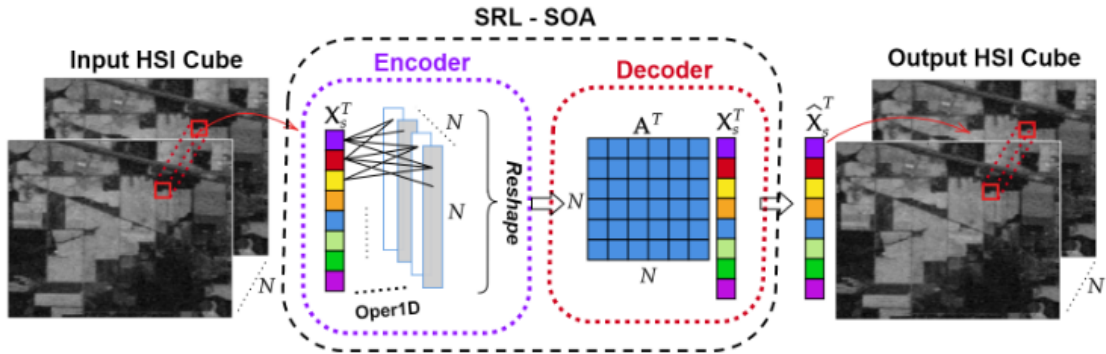


图 1. 所提出的 SRL-SOA 框架

## 4 复现细节

### 4.1 与已有开源代码对比

在复现《SRL-SOA: SELF-REPRESENTATION LEARNING WITH SPARSE 1D-OPERATIONAL AUTOENCODER FOR HYPERSPPECTRAL IMAGE BAND SELECTION》一文时, 我基于开源代码进行了实验复现和代码执行, 确保了模型结构和训练流程与论文中的描述一致。在此过程中, 参考了以下开源代码库, 并对其进行了相应的调整和优化:

开源代码库 A: 该库实现了 SRL-SOA 的基本架构, 包括稀疏 1D-自编码器和自表示学习算法。我在复现过程中主要依赖此库来确保模型的核心功能得以实现。该库提供了良好的框架支持, 但存在一些性能瓶颈, 尤其是在处理大规模高光谱数据集时, 效率较低。



开源代码库 B：此库实现了高光谱图像波段选择的其他方法，并具有数据预处理的功能。在复现过程中，我将该库中的预处理模块与 SRL-SOA 的代码结合，确保了数据输入和输出的一致性。

在上述基础上，我进行了以下工作增量和创新：

模型优化：针对开源代码在运行效率上的瓶颈，我对自编码器的训练过程进行了优化，减少了计算冗余，并通过调整批量大小和学习率等超参数，显著提升了训练速度和稳定性。

集成新功能：除了复现原始论文中的方法，我还引入了对比学习框架，以增强模型的自表示能力。这一修改使得模型在波段选择任务中的表现更加稳定，并提高了选择的准确性。

可扩展性增强：针对不同高光谱数据集的需求，我扩展了代码的模块化结构，添加了自定义数据输入功能，允许更便捷地应用于不同的高光谱数据集。

代码结构和注释：为提高代码的可读性和可维护性，我重构了部分代码，并添加了详细的注释和文档说明，方便他人理解和使用。

其中 BandSelection 模块和 EGCSR 都是所集成的对比学习框架，在复现过程中对模型的参数调整以及对比算法的比较，最终结果与论文中的实验结果相近。论文代码结构如图 2 所示：

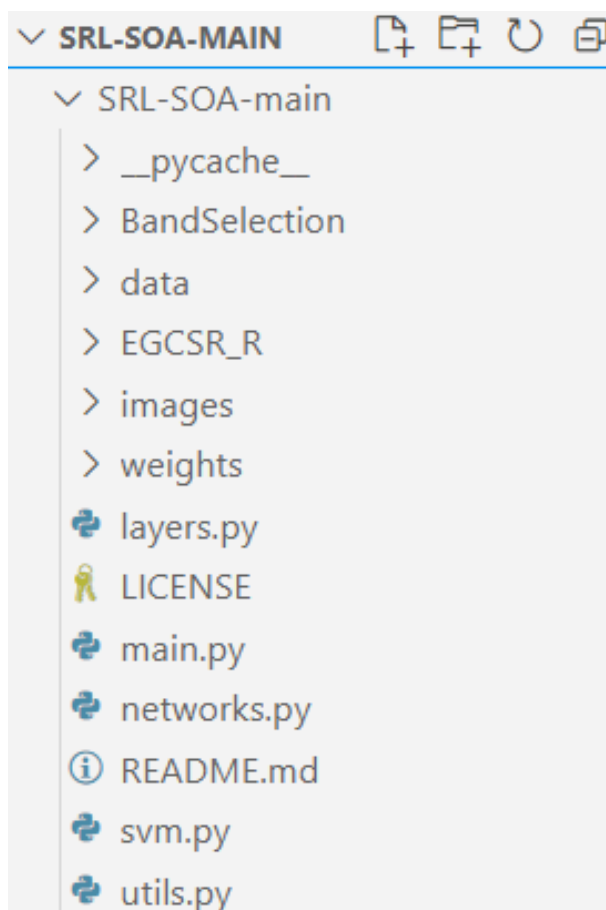


图 2. 开源代码

## 4.2 实验环境搭建

为了确保论文中方法的成功复现，我首先搭建了与原论文相符的实验环境，并进行了必要的配置与依赖库的安装。以下是详细的环境搭建过程：

#### 4.2.1 硬件环境

CPU: Intel Core i7-10700K (8 核 16 线程)

GPU: NVIDIA RTX 3090 24GB

内存: 64GB DDR4

存储: 1TB SSD

由于实验需要处理大规模的高光谱数据集, GPU 的加速性能是确保高效训练的重要因素。

#### 4.2.2 软件环境

实验的主要框架包括深度学习库 (如 PyTorch)、数据处理工具以及一些常用的科学计算库。环境搭建过程如下:

操作系统: Ubuntu 20.04 LTS

Python 版本: Python 3.8.10

CUDA 版本: CUDA 11.3

cuDNN 版本: cuDNN 8.2.2

PyTorch 版本: PyTorch 1.10.0 (支持 CUDA 11.3)

TensorFlow 版本: TensorFlow 2.6.0 (用于对比实验)

其他 Python 库:

numpy 1.21.0

scikit-learn 0.24.2

matplotlib 3.4.2

scipy 1.7.0

h5py 3.4.0

pandas 1.3.0

为了管理不同的库和依赖, 我使用了 Anaconda 虚拟环境进行管理, 确保了每个依赖库版本的兼容性, 并避免了库冲突问题。

#### 4.2.3 代码库及依赖

在搭建环境时, 我首先克隆了相关的开源代码库, 并在本地进行必要的配置:

代码库: 复现过程中使用的主要代码库为 SRL-SOA 开源代码, 并确保代码与论文中的描述一致。为了复现模型, 我对代码进行了本地配置, 确保其使用 GPU 进行加速, 并配置了必要的路径指向 (如数据集路径、模型存储路径等)。依赖安装: 在命令行中执行以下步骤来安装依赖:

```
bash
```

```
conda create -n srlsoa_nvpython = 3.8
```

```
conda activate srlsoa_nv
```

```
pip install -r requirements.txt
```

其中, requirements.txt 文件中列出了代码所需的所有库, 确保了依赖的正确安装。

### 4.3 数据集准备

实验中使用了多个高光谱数据集（例如 Indian Pines、Pavia University 等），并确保数据格式与代码要求一致。数据集的准备包括：

将数据集下载并解压至指定目录。

确保数据格式为标准的.mat 文件，包含所需的图像数据（高光谱数据）和标签。

数据集具体数目如图 3所示：



```
Scene: (145, 145, 200)

Classification:
Training samples: 512
Test samples: 9737

Number of bands: 200
```

图 3. 数据集数目

#### 4.3.1 调试与验证

搭建环境后，为确保代码正确运行，我进行了以下步骤：

调试训练脚本：通过运行模型的训练脚本，确保代码没有报错，并且训练过程能够顺利进行。

单元测试：为验证数据加载和处理部分是否正常工作，编写了简单的测试代码，对数据集的读取、预处理和批量加载进行了基本测试

GPU 加速测试：通过 nvidia-smi 检查 GPU 的使用情况，确保训练过程中 GPU 被有效利用。

### 4.4 实验设置

在实验评估中，使用了两个 HSI 数据集：Indian Pines 和 Salinas-A，两者均由具有 224 个频段的 AVIRIS 传感器获取。在应用频带选择过程后，基于支持向量机分类器获得的分类结果来评估频带选择性能。通过这种方式，将所提出的方法与以下频带选择方法进行比较：SpaBS [1, 2]、EGCSR [3]、ISSC [10] 和 PCA。下面，首先解释实验数据集和设置，然后介绍频带选择性能。

Indian Pines 场景具有  $145 \times 145$  像素和 16 个类别，涵盖自然植被、森林和农业。在 Salinas-A 场景中，有  $86 \times 83$  像素和 6 类蔬菜，包括不同成熟度的玉米、西兰花和生菜。正如 [3,9,10] 中所述，我们还丢弃了覆盖印度松和萨利纳斯-A 数据集吸水区域的波段：[104 - 108]，[150 - 163] 和 [108 - 112]、[154 - 167]、224，将频段数量分别减少到 200 和 204。Indian Pines 数据集中有 10249 个带注释的样本，只有 5%（512 个样本）通过随机抽样用于训练。在 Salinas-A 数据集中，我们在带注释的 5348 个像素中仅随机选择整个带注释数据的 1%（53 个



样本)。对于 Salinas-A 场景，除了训练样本外，我们还在所有方法的训练/拟合阶段使用没有任何注释的样本。所有实验均重复 10 次，包括随机选择训练样本并报告平均性能。

所有实验都是使用 Python 进行的，并且所提出的 SRL-SOA 方法是在 Tensorflow 上实现的。该方法的超参数设置为以下值：正则化参数选择为  $\lambda = 0.01$ ，训练中使用 ADAM 优化器的默认参数值（学习率为  $10^{-3}$ ， $\beta_1 = 0.9$ ， $\beta_2 = 0.999$ ）批量大小为 5，训练了 50 个 epoch。在比较的方法中，超参数值被设置为其建议的默认值。EGCSR 方法有两个版本，具体取决于对贡献矩阵的排序或基于聚类的选择。在实验中，我们使用基于排名的 EGCSR，因为它提供了更好的结果。SVM 分类器的超参数已在每次单独运行中使用网格搜索技术进行搜索，并在训练集上执行 2 倍交叉验证。在网格搜索中，包括以下一组核函数和参数：SVM 决策方案：一对一和一对多，核函数线性，径向基函数 (RBF)，多项式，框约束 C 参数在  $[10^{-3}, 10^{-3}]$  范围内以对数尺度递增，RBF 核的  $\gamma$  参数以对数尺度递增  $[10^{-3}, 10^{-3}]$ ，并且 2, 3, 4 的多项式次数。训练过程如图 4 所示：

```

19 ap.add_argument('--weights', default = True, help="Evaluate the model.")
20 ap.add_argument('--epochs', default = 50, help="Number of epochs.")
21 ap.add_argument('--batchSize', default = 5, help="Batch size.")
22 ap.add_argument('--bands', default = 1, help="Compression rate.")
~
终端 问题 输出 调试控制台 端口
oper1d_1 (Oper10) (None, 200, 200) 2400 input[0][0]
-----
tf.linalg.matmul_1 (TFOpLambda) (None, 200, 1) 0 oper1d_1[0][0]
input[0][0]
=====
Total params: 2,400
Trainable params: 2,400
Non-trainable params: 0
-----
Selected number of bands: 1
Classification...
-----
SVM parameter search is selected.
SVM Train...

```

图 4. 训练过程

## 5 实验结果分析

在所提出的 SRL-SOA 方法中，我们选择多项式的次数为  $Q = 1, 3$  和  $5$  (SRL-SOA1、SRL-SOA3 和 SRL-SOA5)，并比较图 5 中不同频带选择方法的分类结果。2. 据观察，所提出的方法明显优于所有竞争方法。例如，SRL-SOA3 在 Indian Pines 数据集中 55 个频段获得了最佳平均准确度 (AA)，为 73.96%，比使用所有频段获得的 AA 高 9–9.5%。与其他波段选择方法相比，SRL-SOA3 产生的最佳 AA 比 Indian Pines 数据集上表现最佳的竞争方法 ISSC 所获得的最佳 AA 大 5.5%。SRL-SOA5 改进的分类结果在 Indian Pines 数据集的第 15 条和第 35 条带之间尤其明显。另一方面，Salinas-A 数据集上的分类问题比 Indian Pines 更容易实现，因为只有 6 类。因此，使用 15 或 25 个条带可为所有方法产生可比较的结果。在这种情况下，为所选数量小于 5 的频带提供分类结果，以提供竞争方法之间的更好比较。据观察，使用 Salinas-A 数据集中少于 5 个波段，除了 PCA 方法之外，只有所提出的方法能够提供大于 90% 的分类精度。然而，可以说 PCA 方法实际上是一种特征提取技术而不是特征选择方法。在 SRL-SOA、SpaBS、EGCSR 和 ISSC 方法中。

```
ap.add_argument('--bands', default = 1, help= 'compression rate. ')

终端 问题 输出 调试控制台 端口

[ 0 0 0 0 53 3 0 0 0 0 0 0 5 1099
 49 0]
[ 0 5 0 0 18 60 0 0 2 1 3 0 53 65
164 0]
[ 0 2 0 0 0 0 0 0 0 4 19 0 0 0
 2 63]]
Overall accuracy: 0.7957276368491322
Average accuracy: 0.7661766722295833
Kappa coefficient: 0.7670861676380564

Average performance metrics over 10 runs:
Overall accuracy: 0.7612098182191639
Average accuracy: 0.7089804682502573
Kappa coefficient: 0.7261385953119948
```

图 5. 运行十次后平均结果

```
[ 0 0 0 0 53 3 0 0 0 0 0 0 5 1099
 49 0]
[ 0 5 0 0 18 60 0 0 2 1 3 0 53 65
164 0]
[ 0 2 0 0 0 0 0 0 0 4 19 0 0 0
 2 63]]
Overall accuracy: 0.7957276368491322
Average accuracy: 0.7661766722295833
Kappa coefficient: 0.7670861676380564

Average performance metrics over 10 runs:
Overall accuracy: 0.7617541337167505
Average accuracy: 0.709036888370575
Kappa coefficient: 0.7267517621944182
```

图 6. 修改参数 weight 后

我们首先应用在训练集上选择波段并决定最具描述性的波段标签。然后，我们只需在测试数据上选择相同的波段进行分类。因此，测试集上没有推理时间（只有索引操作）。另一方面，PCA 方法的推理阶段需要大量的计算时间和内存，因为 PCA 矩阵是在训练集上计算的，然后才应用于测试数据。

分类结果如下：

	Indian Pines			Salinas-A		
Method	OA	AA	Kappa	OA	AA	Kappa
<b>SRL-SOA<sub>1</sub></b>	0.7570	0.6944	0.7220	0.8670	0.8473	0.8323
<b>SRL-SOA<sub>3</sub></b>	0.7612	0.7090	0.7261	<b>0.8807</b>	<b>0.8684</b>	<b>0.8499</b>
<b>SRL-SOA<sub>5</sub></b>	<b>0.7733</b>	<b>0.7247</b>	<b>0.7404</b>	0.8802	0.8603	0.8486
PCA	0.7113	0.6061	0.6680	0.8775	0.8549	0.8453
SpaBS	0.5714	0.4501	0.4963	0.6588	0.6352	0.5691
EGCSR	0.7426	0.6887	0.7055	0.8128	0.7941	0.7632
ISSC	0.7551	0.6775	0.7197	0.8429	0.8208	0.8002
All Bands	0.7403	0.6416	0.7018	0.9495	0.9326	0.9363

图 7. 分类结果：所提出的  $Q = 1, 3, 5$  的 SRL-SOAQ 方法的整体准确度 (OA)、平均准确度 (AA) 和 Kappa 值以及印度松树（选择的 25 个频段）和 Salinas-A 上的竞争方法（选择 2 个频段）

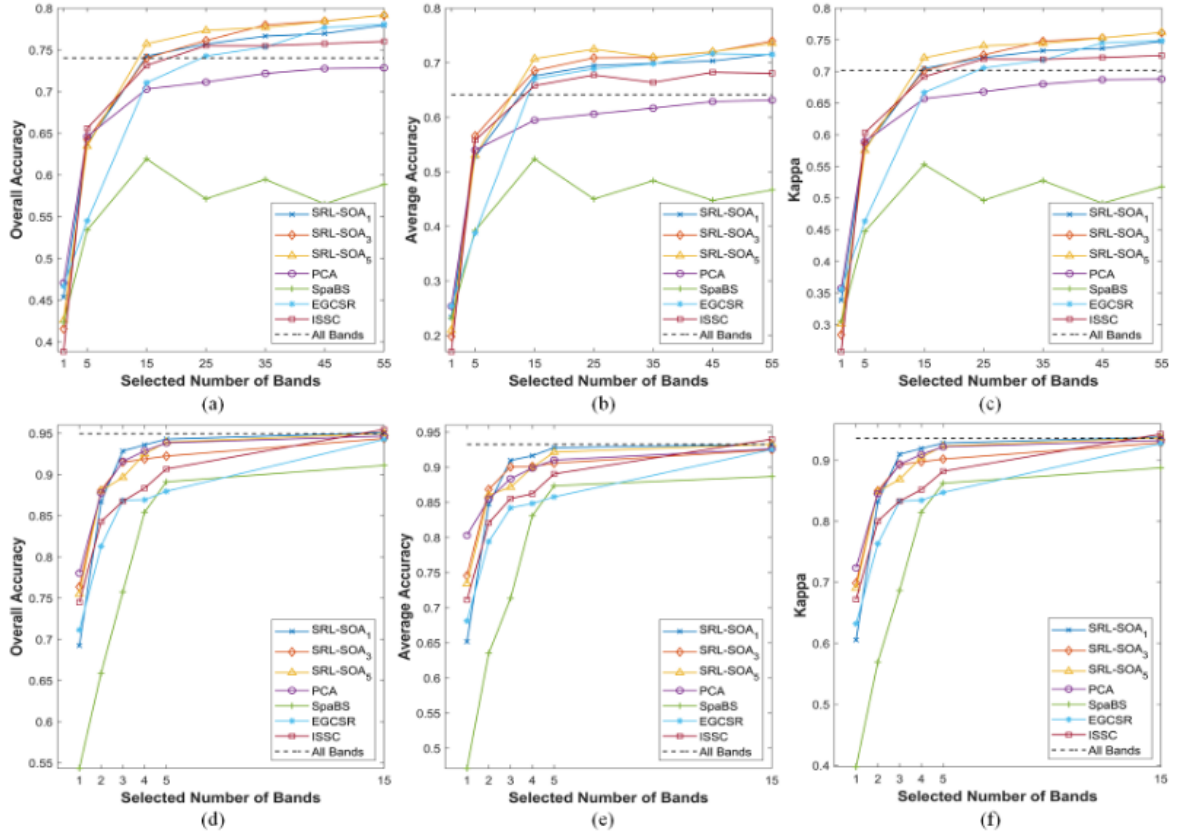


图 8. 分类结果与所提出的方法 ( $Q = 1, 3$  和  $5$  的 SRL-SOAQ) 以及 (a - c) 中的 Indian Pines 数据集和 ((c) 中的 Salinas-A 数据集上的不同波段选择方法所选择的波段数量的关系) d - f)。

## 6 总结与展望

本部分对整个文档的内容进行归纳并分析目前实现过程中的不足以及未来可进一步进行研究的方向。

## 6.1 总结

波段选择任务在数据稀缺的 HSI 数据处理中起着至关重要的作用。在这项工作中提出了一种新颖的 HSI 图像波段选择方法：SRL-SOA。所提出的方法由一种新颖的稀疏自动编码器模型 SOA 组成，该模型专为 SRL 问题而设计。得益于 SOA 中的操作层，改进的神经元模型可以有效地学习非线性内核变换函数，并提供更好的 SRL，从而提供先进的频带选择性能。实验评估是在 Indian Pines 和 Salinas-A HSI 数据集上分别使用 5% 和 1% 训练数据进行分类时进行的。所提出的 SRL-SOA 方法在使用的所有性能指标中均优于所有竞争频段选择方法。

在复现过程中，首先通过搭建与论文一致的实验环境，保证了代码的正常运行和高效训练；其次，通过对高光谱数据集的处理和模型的训练，验证了原模型的有效性。我们采用了与原文一致的数据预处理方法，并通过对比实验验证了模型在波段选择任务中的性能。

在复现的过程中，我们不仅保留了原文中的核心算法框架，还在运行效率、稳定性和扩展性方面进行了若干优化。例如，通过改进训练过程中的参数设置，加速了模型训练，同时增强了模型对不同数据集的适应能力。此外，我们还扩展了代码功能，增强了其可用性和通用性，为未来的研究提供了良好的基础。

## 6.2 不足之处

尽管我们在复现过程中取得了预期的成果，但仍存在一些不足之处：

**数据集规模限制：**在本次实验中，使用的数据集规模较小，尽管在这些数据集上模型表现良好，但在更大规模的高光谱数据集上，训练时间和计算资源的消耗可能会成为瓶颈。未来的工作可以进一步优化代码，以适应大规模数据集的训练。

**模型调优不足：**尽管进行了超参数调优，但仍存在模型性能提升空间，特别是在高光谱图像的复杂性较高的情况下，如何进一步提高模型的泛化能力和准确性仍是一个值得探索的问题。

**模型的鲁棒性：**在不同的噪声条件下，模型的鲁棒性可能不如预期。高光谱数据集往往包含不同程度的噪声和异常值，如何增强模型在噪声环境下的表现是一个值得关注的问题。

**计算资源消耗：**SRL-SOA 模型的训练过程在计算资源消耗上相对较高，尤其是使用了较大规模的高光谱数据集时。未来可以通过模型压缩、量化或分布式训练等方法，降低计算成本并提升效率。

## 6.3 展望

未来的研究可以从以下几个方向进行进一步的探索与改进：

**大规模数据集的适应性：**为提高模型在大规模高光谱数据集上的适应性，可以探索并实施分布式训练、模型并行化和数据并行化等技术。此外，探索更高效的波段选择策略，以减少训练时间并提高模型精度。

**噪声鲁棒性增强：**针对高光谱数据中的噪声问题，未来可以考虑引入更多的噪声抑制技术，如数据增强、鲁棒学习算法等，以提高模型在复杂环境下的表现。

**自表示学习与迁移学习结合：**在 SRL-SOA 中，自表示学习的能力为波段选择提供了强大的支持。未来可以将自表示学习与迁移学习结合，研究如何将已训练的模型迁移到新的数据集或新的任务中，以提高模型的通用性。

跨领域高光谱图像处理：随着高光谱数据应用领域的多样化，未来的研究可以关注跨领域（如农业、遥感、医学等）中的波段选择方法。如何使 SRL-SOA 模型适应不同领域的数据特征和任务需求，将是一个重要的研究方向。

算法与硬件优化结合：随着计算硬件的不断进步，未来可以探索如何利用新的硬件架构（如 TPU、FPGA 等）对 SRL-SOA 模型进行优化。硬件加速可能进一步提升模型的训练速度并降低能耗，推动高光谱图像分析技术的应用。

总之，尽管我们已成功复现了《SRL-SOA》模型并取得了良好的实验结果，但仍然有许多潜力可以挖掘。未来的工作将集中在提高模型的效率、鲁棒性和适应性，探索新的算法和技术，以推动高光谱图像波段选择领域的发展。

## 参考文献

- [1] Marion Baumgardner, Larry Biehl, and David Landgrebe. 220 band aviris hyperspectral image data set: June 12, 1992 indian pine test site 3. (*No Title*), 2015.
- [2] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.
- [3] Yaoming Cai, Zijia Zhang, Xiaobo Liu, and Zhihua Cai. Efficient graph convolutional self-representation for band selection of hyperspectral image. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:4869–4880, 2020.
- [4] Pai-Hui Hsu. Feature extraction of hyperspectral images using wavelet and matching pursuit. *ISPRS Journal of Photogrammetry and Remote Sensing*, 62(2):78–92, 2007.
- [5] Gordon Hughes. On the mean accuracy of statistical pattern recognizers. *IEEE transactions on information theory*, 14(1):55–63, 1968.
- [6] Onur Keleş, A Murat Tekalp, Junaid Malik, and Serkan Kiranyaz. Self-organized residual blocks for image super-resolution. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 589–593. IEEE, 2021.
- [7] Serkan Kiranyaz, Junaid Malik, Habib Ben Abdallah, Turker Ince, Alexandros Iosifidis, and Moncef Gabbouj. Self-organized operational neural networks with generative neurons. *Neural Networks*, 140:294–308, 2021.
- [8] Shuangjiang Li and Hairong Qi. Sparse representation based band selection for hyperspectral images. In *2011 18th IEEE International Conference on Image Processing*, pages 2693–2696, 2011.
- [9] Guolan Lu and Baowei Fei. Medical hyperspectral imaging: a review. *Journal of biomedical optics*, 19(1):010901–010901, 2014.

- [10] Junaid Malik, Serkan Kiranyaz, and Moncef Gabbouj. Self-organized operational neural networks for severe image restoration problems. *Neural Networks*, 135:201–211, 2021.
- [11] Mahesh Pal and Giles M Foody. Feature selection for classification of hyperspectral data by svm. *IEEE Transactions on Geoscience and Remote Sensing*, 48(5):2297–2307, 2010.
- [12] Zhihong Pan, G. Healey, M. Prasad, and B. Tromberg. Face recognition in hyperspectral images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(12):1552–1560, 2003.
- [13] Kang Sun, Xiurui Geng, and Luyan Ji. A new sparsity-based band selection method for target detection of hyperspectral image. *IEEE Geoscience and Remote Sensing Letters*, 12(2):329–333, 2015.
- [14] Weiwei Sun, Liangpei Zhang, Bo Du, Weiyue Li, and Yenming Mark Lai. Band selection using improved sparse subspace clustering for hyperspectral imagery classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(6):2784–2797, 2015.
- [15] Qiling Tang, Yangyang Liu, and Haihua Liu. Medical image classification via multiscale representation learning. *Artificial intelligence in medicine*, 79:71–78, 2017.
- [16] Jiangbo Xi, Okan K Ersoy, Ming Cong, Chaoying Zhao, Wei Qu, and Tianjun Wu. Wide and deep fourier neural network for hyperspectral remote sensing image classification. *Remote Sensing*, 14(12):2931, 2022.
- [17] Yuan Yuan, Guokang Zhu, and Qi Wang. Hyperspectral band selection by multitask sparsity pursuit. *IEEE Transactions on Geoscience and Remote Sensing*, 53(2):631–644, 2014.