

# 基于改进ResNet的电池极片毛刺分类研究

李朋飞

上海理工大学机械工程学院，上海

收稿日期：2025年1月24日；录用日期：2025年2月17日；发布日期：2025年2月25日

## 摘要

针对电池极片毛刺漏检、过杀和检测效率低问题，提出一种基于改进ResNet的电池极片毛刺分类方法。在数据预处理阶段，采用图像生成模型来增加数据的多样性和复杂性，帮助模型更好地泛化；选择ResNet18作为骨干网络，通过融合改进MSE注意力机制提高每个毛刺类别的召回率和精确率，并利用GSConv卷积实现模型轻量化加快模型推理速度。实验结果表明：改进后的网络Ours-ResNet18相比较改进前网络F1-score得分达到96.78%，提高了5.92%，单张图片CPU检测时间达到15.26 ms，能够满足实际的检测需求。

## 关键词

电池极片毛刺，ResNet，注意力机制，GSConv卷积

# Research on Burr Classification of Battery Pole Pieces Based on Improved ResNet

Pengfei Li

School of Mechanical Engineering, University of Shanghai for Science and Technology, Shanghai

Received: Jan. 24<sup>th</sup>, 2025; accepted: Feb. 17<sup>th</sup>, 2025; published: Feb. 25<sup>th</sup>, 2025

## Abstract

Aiming at the problems of missed detection, over-killing and low detection efficiency of burrs on battery pole pieces, a classification method for burrs on battery pole pieces based on the improved ResNet is proposed. In the data preprocessing stage, an image generation model is adopted to increase the diversity and complexity of the data, helping the model to generalize better. ResNet18 is selected as the backbone network. By integrating and improving the MSE attention mechanism, the recall rate and precision rate of each burr category are improved. And GSConv convolution is utilized to make the model lightweight and accelerate the model's inference speed. The experimental results show that compared with the network before improvement, the F1-score of the improved network Ours-ResNet18 reaches 96.78%, an increase of 5.92%. The CPU detection time for a single

image is 15.26 ms, which can meet the actual detection requirements.

## Keywords

Burrs on Battery Pole Pieces, ResNet, Attention Mechanism, GSConv Convolution

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 引言

电池极片毛刺是指电池极片表面突出的细小金属或其他杂质，可能会导致电池在工作过程中出现短路、漏电等安全问题，甚至影响电池的性能和寿命[1]。在极片表面存在毛刺会导致不均匀的电流分布，降低了电池的传输效率，进而缩短了电池的寿命[2]。通过检测和处理毛刺，可以提高电池的性能并延长其使用寿命。在生产线上对电池进行毛刺检测，可以及早发现问题并及时进行修正，避免将有缺陷的电池装配到最终产品中，从而节省成本和时间。

目前对极片缺陷检测的研究，包括传统图像处理算法和人工智能算法。传统的毛刺分类检测方法主要包括极片毛刺检测普遍采用目视显微镜[3]，存在景深不够、倍率不足，实际测量数据误差较大。当前极片缺陷检测研究有进展，传统目视显微镜检测有局限，不过学者们积极利用先进算法改进。像王[4]等用图像处理与卷积神经网络法经标定轮廓区域延展图像输入模型确定缺陷类型；Mao [5]等以深度学习手段，通过预处理图像增强缺陷特征来构建分类算法；Xu [6]等基于多特征融合和 PSO-SVM 实现检测与识别；黄[7]等融合特征并借助支持向量机完成缺陷分类。

本文以电池极片毛刺为研究对象，采集缺陷图像并制作数据集，对 ResNet18 模型进行网络改进，并与其他传统神经网络模型 VGG16 [8]、AlexNet [9]、MobileNet\_V3\_small [10]、ResNet 等进行对比分析，验证本文模型对毛刺识别的性能提升。通过对网络改进，实现快速准确地识别不同毛刺类别，为改进工业缺陷检测提供解决方案，提升电池制造安全保证。

## 2. 数据采集与预处理

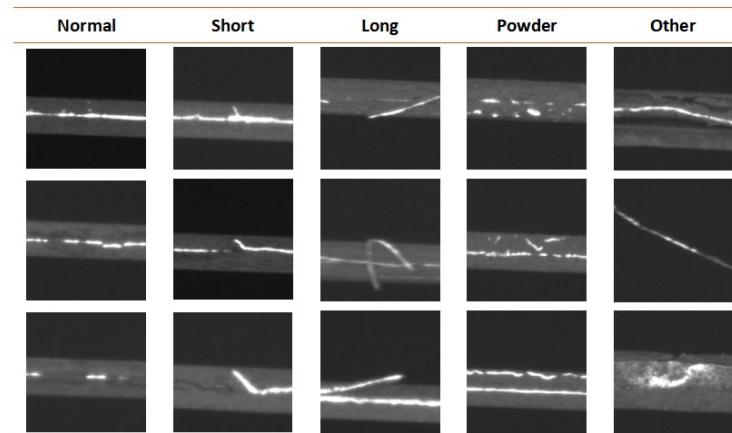


Figure 1. Sample images of data for various categories

图 1. 各类别数据样图

本文通过工业相机采集全部极片毛刺图像，采集并裁剪后图像的原始分辨率为 $224 \times 224$ 。收集数据集依据电池极片边缘特征精准划分为五类，如图1所示。“Normal”类电池极片边缘极为平滑，无明显毛刺，边缘整齐，无凸起凹陷，彰显卓越加工品质。“Short”类的电池极片边缘偶现短小不连续毛刺，且其垂直延伸未超灰色隔膜界限，属轻微瑕疵且毛刺未超过涂层边缘 $7\text{ }\mu\text{m}$ 。“Long”类的极片边缘毛刺显著伸长，越过隔膜边界 $7\text{ }\mu\text{m}$ ，常因加工流程不稳定，如刀具磨损、切削力不均所致。“Powder”类极片边缘毛刺复杂，伴有大量颗粒或粉末残留，杂质源于加工碎屑、润滑剂残留等，有损外观且威胁电池性能。“Other”类涵盖机器挤压所致特殊情况，尤其隔膜断裂意外。

**Table 1.** Dataset distribution**表 1.** 数据集分布

| Class      | Normal | Short | Long | Powder | Other |
|------------|--------|-------|------|--------|-------|
| Training   | 987    | 1185  | 1883 | 415    | 432   |
| Validation | 247    | 297   | 471  | 104    | 109   |
| Total      | 1234   | 1482  | 2354 | 519    | 541   |

训练网络模型时需要足够的数据集样本进行训练，避免网络出现过拟合问题[11]。为了增强数据样本以减少过拟合，提高模型的泛化能力，标准化张量，随机水平翻转图像数据增强等方式，这些转换有助于模型学习对图像中的水平方向不敏感的特征[12]，然而这个图像预处理方式并不能改变图像本质，无法获得更加丰富的缺陷特征。

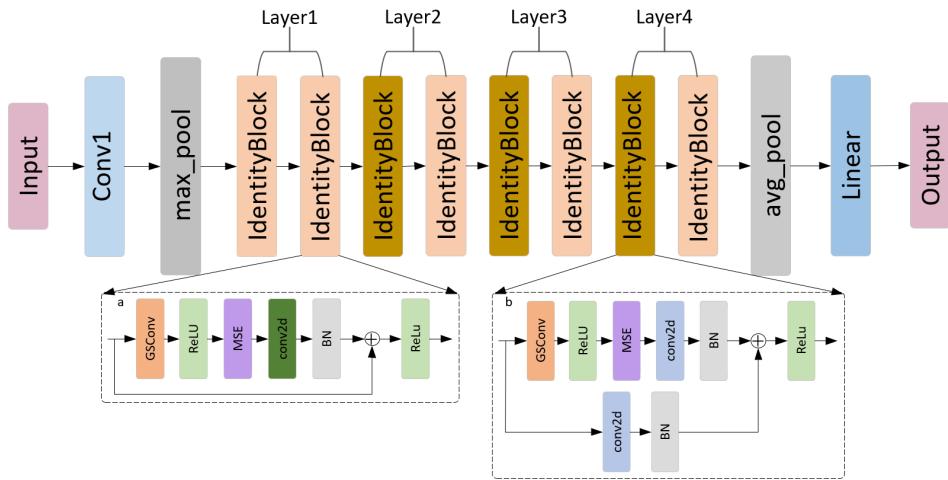
本文将筛选出的5130张图像按照其特征进行人工标注，另外在这些源图像中挑选出部分图像作为图像生成使用的训练集，再利用图像生成1000张图像，最终重新打乱每一个类别的源图像与生成图像按照8:2比例划分为训练集与验证集如表1所示。

### 3. 模型构建

#### 3.1. 选择主干网络

在针对极片毛刺的分类检测任务中，我们旨在构建一个模型，该模型不仅需维持高精确率与召回率以确保检测准确性，同时需进行轻量化设计，以减少计算资源消耗，加快检测速度，从而充分满足工业生产线上的实时检测需求。传统的深度神经网络存在梯度消失的问题，结合了残差网络的优势，改善梯度消失的问题，并构建更深层的网络结构、提取更复杂和抽象的特征[13]。针对上述问题，何凯明[14]等提出ResNet，核心结构是残差模块(Residual block)，能将信息从浅层传递到深层，使网络在增加层数的同时保证性能。本文选择ResNet18作为主干网络，其主要特点包括：引入了残差块，通过跳跃连接来解决深度神经网络训练时的梯度消失问题，帮助网络更容易优化和训练更深的网络，采用基本的卷积、批量归一化、池化等结构。同时，在不同层数的网络中增加了残差块来提高网络性能，在网络的最后采用全局平均池化层来减少模型参数数量和计算量，以及防止过拟合，最后部分是一个全连接层，用于将提取的特征映射到输出类别的分数。

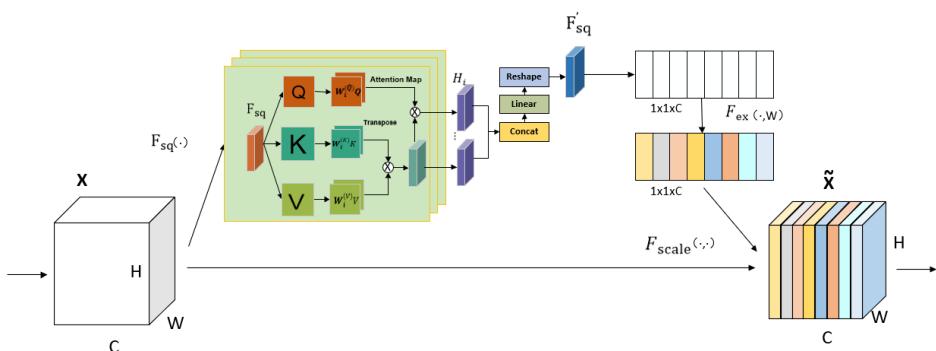
本文结合ResNet网络的优势，在主干网络上做了一些改进具体内容如下：用轻量积代替标准卷积减少模型的参数量与计算量，在残差块中使用融合改进的MSE注意力机制对特征的不同位置提供不同的注意，并进行全局建模，改进后Ours-ResNet18型框架如图2所示，其中Layer1-Layer4分别包括2个残差模块，每一个残差块中都将采用GSConv卷积和MSE注意力机制技术，改进的残差模块结构如图2(a)，另外，Layer2-Layer4的第一个残差模块需要将输出和输入的数据维度转换成相同的，采用shortcut方式传递，结构如图2(b)。



**Figure 2.** The structure diagram of the improved Ours-ResNet18 network  
**图 2.** 改进 Ours-ResNet18 网络结构图

### 3.2. 改进 MSE 注意力机制

为了动态地学习输入特征图的通道之间的关系，从而使模型在学习特征表示时能够更加关注重要的特征，本文在网络的残差块中引入 SE 注意力机制[15]：其基本思想是通过两个步骤来提取通道间的关系：压缩和激励。然而在压缩阶段，将特征图的空间维度压缩到  $1 \times 1$  意味着所有空间位置的信息都被整合成了一个单一的值，这会导致空间信息的显著损失。因此在特征图压缩后特征图激励之前引入多头注意力机制：设计了注意力特征编码器对经过特征提取网络的特征图进行运算处理，每个通道的全局特征可以通过多个注意力头来独立处理，捕捉特征图的交互信息[16]，提取目标完整信息进而把控图像全局的关键信息，获得更为精细的特征信息并提升网络性能。改进后的 MSE 注意力机制(Multi-SE-Attention)结构如下图 3 所示：



**Figure 3.** Result diagram of the improved MSE attention mechanism  
**图 3.** 改进 MSE 注意力机制结果图

其主要工作原理步骤如下

(1) Squeeze 将特征图  $X$  跨空间维度  $H \times W$  的特映射进行聚合，生成一个通道描述符，将全局空间信息运用全局平局池化的方法压缩到通道描述符中，如式所示。

$$z_c = F_{sq}(u_c) = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W u_c(i, j) \quad (1)$$

式中： $z_c$  表示输入特征图的通道描述符， $i, j$  分别代表各通道和空间维度的索引。

(2) 多头注意力机制是一种源于自然语言处理领域的技术，其核心思想是在处理序列数据时，通过计算并分配输入序列中每个位置的注意力权重，使得模型能够更好地捕获关键信息，这种机制在图像处理领域同样适用[17]。

首先将来自特征提取网络的特征图输入到注意力特征编码器，将特征图随机映射为  $Q \in R^{d_Q}$  和  $K \in R^{d_K}$  和  $V \in R^{d_V}$  三个参数矩阵，之后根据预设的注意力头个数，对  $Q, K, V$  进行拆解变换，得到  $W_i^{(Q)}Q, W_i^{(K)}K$  和  $W_i^{(V)}V$  力计算处理，其中自注意力模型为：

$$\text{Attention} = (Q, K, V) = \text{softmax} \left( \frac{QK^T}{\sqrt{d_K}} \right) V \quad (2)$$

我们在每个注意力头中对，和的不同部分分别进行自注意力计算处理，其中注意力头可表示为：

$$H_i = \text{Attention}(W_i^{(Q)}Q, W_i^{(K)}K, W_i^{(V)}V) \quad (3)$$

将每个注意力头得到的输出矩阵拼接起来，经过全连接层提取特征关联信息，再经过 Reshape 操作将特征图还原为原来的维度，输出特征图，继续传递到网络的深层次结构中。特征图经过多头注意力机制处理后，捕获到了图像全局的关联性特征，抑制了无关噪声，增强了的特征提取能力和网络表征能力。

(3) Excitation 将步骤(2)得到的通过一个全连接层，将得到结果输入到 ReLU 激活函数中，再通过一个全连接层，最后通过 Sigmoid 激活函数，如式(4)、(5)所示。

$$\text{Sigmoid}(z) = \frac{1}{1 + e^{-z}} \quad (4)$$

$$S = F_{ex}(z, W) = \sigma(g(z, W)) = \sigma(W_2 \delta(W_1 z)) \quad (5)$$

式中： $\delta$  代表 ReLU 激活函数， $\sigma$  代表 Sigmoid 激活函数、为全连接层， $W_1$  的维度为  $C \times r \times C$ ， $W_2$  的维度为  $C \times r \times C$ 。

(4) Reweight 操作将每个通道的权重和特征图按照通道相乘得到加权后的  $\tilde{t}_c$ ，最终得到按通道加权的输出  $\tilde{T}$ ，如式(6)、(7)所示：

$$\tilde{t}_c = F_{scale}(u_c, s_c) = s_c u_c \quad (6)$$

$$\tilde{T} | \tilde{t}_1, \tilde{t}_2, \tilde{t}_3 | \quad (7)$$

式中： $F_{scale}(u_c, s_c)$  表示对输入特征图  $u_c$  中的每个通道进行逐元素乘法。

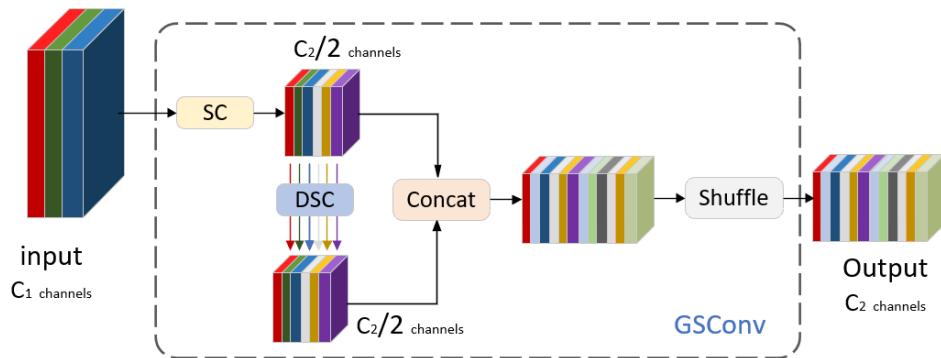
### 3.3. 轻量化设计

由于在主干网络中融合注意力机制网络的参数量及计算量大大增加，与实际的工业场景模型轻量化的要求不相符，因此需要对网络在保证准确率的情况下对其进行轻量化设计。

传统的深度可分离卷积 DSC 由 DW (Depthwise) 卷积和 PW (Pointwise) 卷积组成，旨在减小计算复杂度和模型参数量，从而提高计算效率和模型性能[18]。从参数计算量的角度分析：标准卷积的参数计算量为  $N \times M \times D_F^2 \times D_K^2$ ，而深度可分离卷积的计算量为  $M \times D_F^2 \times D_K^2 + (1 \times 1 \times M \times N \times D_F^2)$ ，其中  $D_F$  为特征图尺寸， $D_K$  为卷积核尺寸，输入通道数记为  $M$ ，输出通道数记为  $N$ ，由式(8)计算可得，深度可分离卷积的计算量与传统卷积计算量的比值为  $1/N + 1/D_K^2$ 。

$$\frac{\text{深度可分离卷积}}{\text{普通卷积}} = \frac{M \times D_F^2 \times (D_K^2 \times N)}{N \times M \times D_F^2 \times D_K^2} = \frac{D_K^2 + N}{N \times D_K^2} = \frac{1}{N} + \frac{1}{D_K^2} \quad (8)$$

但 DSC 有个很明显的缺点：在计算过程中，输入图像的通道信息被隔离，导致 DSC 的特征提取和融合能力比 SC (Standard Convolution) 低得多。为进一步加以改进，并让 DSC 的输出结果尽可能接近 SC，Li [19] 等人设计了一种将 SC、GSConv 和 shuffle 混合使用的 GSConv 卷积。GSConv 模块的结构图如图 4 所示：



**Figure 4.** GSConv convolutional structure  
**图 4.** GSConv 卷积结构

它结合密集计算的准确性和深度计算的轻量化，在确保精度的基础上，不仅解决了传统的深度可分离卷积带来的问题，能减少参数量，还能提高检测速度。DSC、SC、GSConv 的时间复杂度如公式(9)~(11)所示，其中，W 为输出特征图的宽度；H 为输出特征图的高度； $K_1, K_2$  为卷积核的大小； $C_1$  为每个卷积核的通道数，也是输入特征图的通道数， $C_2$  为输出特征图的通道数。

$$\text{Time}_{\text{SC}} \sim O(W \cdot H \cdot K_1 \cdot K_2 \cdot C_1 \cdot C_2) \quad (9)$$

$$\text{Time}_{\text{DSC}} \sim O(W \cdot H \cdot K_1 \cdot K_2 \cdot 1 \cdot C_2) \quad (10)$$

$$\text{Time}_{\text{GSConv}} \sim O\left[W \cdot H \cdot K_1 \cdot K_2 \cdot \frac{C_2}{2} (C_1 + 1)\right] \quad (11)$$

## 4. 实验结果与分析

### 4.1. 实验平台

使用 Linux 操作系统，AMD Ryzen Threadripper 3960X 24-Core Processor 处理器，24GB 显存，NVIDIA GeForce RTX 3090 的 GPU。开发平台为 VScode，使用 Cuda 12.0 搭建环境，Python 3.10.12 编程语言，PyTorch 2.0.1 深度学习框架。模型训练 200 个 epoch，设置 batchsize 为 64，通过 Pytorch 深度学习框架进行模型搭建，算法使用 CUDA 库进行加速运算。

### 4.2. 模型评价指标

使用准确率(Accuracy)、精确率(Precision)、召回率(Recall)和 F1-score 作为评价标准，具体定义如式(12)至(15)：

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (12)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (13)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (14)$$

$$\text{F1} = 2 \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (15)$$

TP 表示模型将正类正确分类为正类的样本数量, FP 表示将负类错误分类为正类的样本数量, TN 表示将负类正确分类为负类的样本数量, FN 表示将正类错误分类为负类的样本数量。通常准确率、精确率、召回率和 F1-score 的值越大, 表示模型的分类性能越好。

### 4.3. 结果与分析

#### 4.3.1. 不同分类模型对比

实验将所提模型 Ours-ResNet18 与 AlexNet、VGG16、MobileNet\_V3\_small、ShuffleNet\_V2\_x1\_0、ResNet18 和 ResNet34 进行对比, 评估所提模型在极片毛刺缺陷数据集上的分类性能和效果, 各模型在验证集上的召回率和精确率曲线如表 2 所示。

**Table 2.** Comparison of classic classification models

**表 2. 经典分类模型对比**

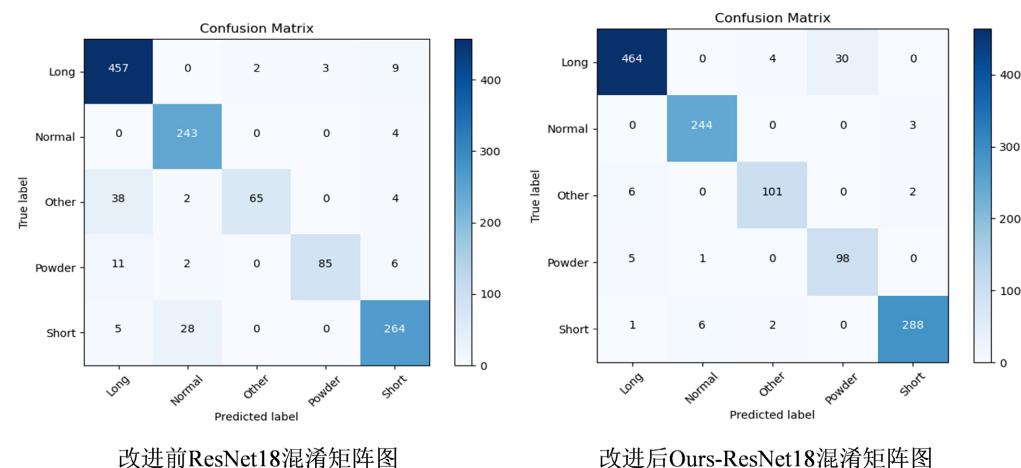
| Model              | Accuracy/% | Precision/% | Recall/% | F1-score/% | Params/MB | Cpu/ms |
|--------------------|------------|-------------|----------|------------|-----------|--------|
| AlexNet            | 88.10      | 88.20       | 88.21    | 88.21      | 217.53    | 32.81  |
| VGG16              | 87.99      | 89.82       | 88.38    | 89.09      | 512.24    | 74.26  |
| Swin_v2_t          | 81.47      | 88.92       | 81.53    | 85.07      | 72.26     | 71.52  |
| MobileNet_V3_small | 87.30      | 92.86       | 87.28    | 89.99      | 5.81      | 12.64  |
| ShuffleNet_V2_x1_0 | 85.18      | 93.33       | 84.80    | 88.86      | 4.80      | 11.30  |
| ResNet18           | 88.07      | 93.73       | 88.15    | 90.86      | 42.64     | 19.57  |
| ResNet34           | 88.84      | 94.66       | 88.76    | 91.62      | 81.20     | 27.80  |
| Ours-ResNet18      | 96.47      | 96.96       | 96.60    | 96.78      | 28.72     | 15.26  |

由上表 2 可以看出, 对比的 AlexNet、VGG16、MobileNet\_V3\_small、ShuffleNet\_V2\_x1\_0、ResNet18 和 ResNet34 网络模型的 F1-score 都达到 85%以上, 其中主干网络 ResNet18 的 F1-score 达到 90.86%, 为对比网络的最高值, 所提模型 Ours-ResNet18 的 F1-score 达到 96.78%, 相比改进前 F1-score 提高 5.92%, 同时在准确率、精确率和召回率上比改进前分别提升了 8.4%、3.23% 和 8.45%, 且都高于其他对比模型; 在单张平均检测速度方面, 相比改进前大约提高 22%, 虽然略慢于 ShuffleNet\_V2\_x1\_0 和 MobileNet\_V3\_small, 但 F1-score 较两个模型分别提高了 6.79% 和 7.92%, 企业在毛刺分类检测过程中, 单张节拍小于 30ms, 本文方法检测精度与速度满足实际生产要求。综合评价, 所提 Ours-ResNet18 网络在此次分类任务上减小了模型大小, 还兼顾了检测精度和检测速度, 综合性能更优。

#### 4.3.2. 模型验证

为进一步验证模型的分类效果, 用混淆矩阵查看测试集上各类别分类结果, 并得到本文方法与原模型 ResNet18 在每类缺陷上的识别精确率和召回率评价指标, 两个模型的混淆矩阵如图 5, 实验结果如表 3 所示。

从表 3 分析可知在模型改进后, 所有类别的精确度(Precision)、召回率(Recall)和 F1 分数均有显著提升。这表明改进后的模型在识别不同类别的物体时, 具有更高的准确度和可靠性。总体而言, 改进后的 Ours-ResNet18 模型在识别不同类别的物体时, 性能得到了显著提升。



**Figure 5.** Comparison of confusion matrices  
**图 5. 混淆矩阵对比**

**Table 3.** Comparison of various indicators before and after model improvement.

**表 3. 模型改进前后各指标对比**

| Class      | 改进前 ResNet18 |          |            | 改进后 Ours-ResNet18 |          |            |
|------------|--------------|----------|------------|-------------------|----------|------------|
|            | Precision/%  | Recall/% | F1-score/% | Precision/%       | Recall/% | F1-score/% |
| Long       | 91.62        | 97.45    | 94.44      | 97.68             | 98.51    | 98.10      |
| Normal     | 92.75        | 98.38    | 95.48      | 96.44             | 98.79    | 97.60      |
| Other      | 95.95        | 65.14    | 77.60      | 95.33             | 93.58    | 94.44      |
| Powder     | 94.74        | 86.54    | 90.45      | 97.09             | 96.15    | 96.62      |
| Short      | 93.58        | 93.27    | 93.42      | 98.28             | 95.96    | 97.10      |
| Mean value | 93.73        | 88.15    | 90.86      | 96.96             | 96.60    | 96.78      |

#### 4.3.3. 消融实验

本文结合了多种技术的模型，进行了不同改进模块和不同多头个数的消融实验，如下表 4 所示：其中结合 MSE 与 GSConv 技术 Ours-ResNet18-8 (多头个数为 8) 表现最佳，F1-score 达到了最高的 96.78%。值得注意的是，随着多头数量的增加(从 4 到 16)，准确率并未持续上升，这表明并不是多头数越多模型表现就越优秀。在 CPU 推理时间上，改进后的 Ours-ResNet18 模型相比较于没有使用 GSConv 技术的 MSE 模型有显著改善，单张 cpu 检测时间达到了 15.26ms。

**Table 4.** Comparison of different improvement modules.

**表 4. 不同改进模块对比**

| Model            | Accuracy/% | Precision/% | Recall/% | F1-score/% | Params/MB | Cpu/ms |
|------------------|------------|-------------|----------|------------|-----------|--------|
| SE               | 92.08      | 95.06       | 92.20    | 93.61      | 42.98     | 17.07  |
| MSE              | 93.95      | 95.59       | 93.22    | 94.39      | 43.03     | 19.20  |
| GSConv           | 92.40      | 95.29       | 92.45    | 93.85      | 26.71     | 14.73  |
| SE-GSConv        | 91.52      | 95.17       | 91.86    | 93.49      | 27.05     | 14.01  |
| Ours-ResNet18-4  | 96.18      | 96.48       | 95.85    | 96.16      | 27.09     | 13.99  |
| Ours-ResNet18-8  | 96.47      | 96.96       | 96.60    | 96.78      | 27.55     | 15.26  |
| Ours-ResNet18-16 | 96.25      | 96.93       | 96.48    | 96.70      | 28.72     | 16.31  |

综上所述：结合了 MSE 与 GSConv 技术的模型 Ours-ResNet18 在准确率、精确率、召回率、F1 分数以及 CPU 推理时间上均表现出色，同时保持了较低的参数数量。这表明深度可分离卷积技术是一种有效的模型优化手段，可以在不显著增加模型复杂度的前提下，显著提升模型的性能。

## 5. 小结

本文旨在解决电池极片毛刺分类难以及检测效率低的问题，通过一系列研究得出以下主要结论：首先，利用工业相机采集电池极片毛刺缺陷图像，并借助图像生成技术构建了更为复杂多样的数据集，为后续精准识别奠定了良好的数据基础。其次，以残差网络作为主干网络，提出基于融合改进 MSE 注意力机制的识别模型 Ours-ResNet18，该机制有效提高了每个毛刺类别的召回率和精确率，同时结合 GSConv 卷积实现了模型轻量化，显著加快了模型推理速度，进而提升了对毛刺缺陷识别的准确率以及检测效率。最后，经验证，Ours-ResNet18 网络在构建的数据集上，无论是识别准确率还是检测速度，相较于改进前均有明显提升，并且与其他经典网络对比，此模型在识别细微缺陷特征方面更具优势。但是，Ours-ResNet18 也存在一定的局限性，比如网络结构还有进一步完善的空间。后续计划采集更多涵盖其他缺陷类别的极片毛刺缺陷数据，同时持续优化模型，致力于提取更具判别力的特征，以实现更加广泛和细致的缺陷识别，不断提升电池极片毛刺检测的效果。

## 参考文献

- [1] 韩任杰. 极片毛刺导致电池短路的检测方法[J]. 电池, 2023, 53(5): 538-540.
- [2] 杨续来, 袁帅帅, 杨文静, 等. 锂离子动力电池能量密度特性研究进展[J]. 机械工程学报, 2023, 59(6): 239-254.
- [3] 崔远驰, 张嵩, 唐晗, 等. 中国仪器产业发展: 现状, 挑战及展望[J]. 光学精密工程, 2024, 32(13): 1965-2003.
- [4] 王庆秋, 李林升, 桂久琪, 等. 基于图像处理与卷积神经网络的锂电池极片缺陷检测与分类[J]. 制造业自动化, 2023, 45(10): 50-54.
- [5] Mao, W., Li, L., Tao, Y. and Zhou, W. (2023) Surface Defect Image Classification of Lithium Battery Pole Piece Based on Deep Learning. *IEICE Transactions on Information and Systems*, **106**, 1546-1555.  
<https://doi.org/10.1587/transinf.2023edp7058>
- [6] Xu, C., Li, L., Li, J. and Wen, C. (2021) Surface Defects Detection and Identification of Lithium Battery Pole Piece Based on Multi-Feature Fusion and PSO-SVM. *IEEE Access*, **9**, 85232-85239.  
<https://doi.org/10.1109/access.2021.3067641>
- [7] 黄梦涛, 王露. 锂电池极片缺陷特征融合与分类[J]. 制造业自动化, 2021, 43(10): 61-63.
- [8] 徐文超, 张思祥, 白芳, 等. 基于改进 VGG 网络的单体热电池 X 光图像无损检测方法研究[J]. 光电子·激光, 2024, 35(2): 180-190.
- [9] 沈凌云, 张洁, 孙甲云, 等. 基于改进 Alexnet 模型的光伏电池片瑕疵分类方法[J]. 自动化与信息工程, 2022, 43(4): 42-47.
- [10] 任惠, 夏静, 卢锦玲, 等. 基于红外图像和改进 MobileNet-V3 的光伏组件故障诊断方法[J]. 太阳能学报, 2023, 44(8): 238-245.
- [11] 李莹, 董耀, 何自芬, 等. 基于 ResNet34-TE 的喷印可变彩色二维码缺陷检测[J]. 激光与光电子学进展, 2024, 61(18): 1-12.
- [12] 吴岸聪, 林城梽, 郑伟诗. 面向跨模态行人重识别的单模态自监督信息挖掘[J]. 中国图象图形学报, 2022, 27(10): 2843-2859.
- [13] 莫定涛, 俱莹, 李文进, 等. 面向卫星网络的流量分类方法研究[J/OL]. 计算机工程, 1-13.  
<https://doi.org/10.19678/j.issn.1000-3428.0069654>, 2025-02-20.
- [14] He, K., Zhang, X., Ren, S. and Sun, J. (2016). Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, 27-30 June 2016, 770-778.  
<https://doi.org/10.1109/cvpr.2016.90>
- [15] 谢宇浩, 董前民, 金尚忠, 等. 基于深度神经网络的危化品拉曼光谱识别[J/OL]. 激光与光电子学进展, 1-17.  
<http://kns.cnki.net/kcms/detail/31.1690.tn.20240828.1425.030.html>, 2025-02-20.

- [16] 许冲, 刘庆华. 融合多头注意力机制的探地雷达反演网络[J/OL]. 现代雷达, 1-15.  
<http://kns.cnki.net/kcms/detail/32.1353.TN.20240725.0938.002.html>, 2025-02-20.
- [17] Ru, Y., An, G., Wei, Z. and Chen, H. (2024) Epilepsy Detection Based on Multi-Head Self-Attention Mechanism. *PLOS ONE*, **19**, e0305166. <https://doi.org/10.1371/journal.pone.0305166>
- [18] Xu, H., Xiao, Y., Sun, K. and Cui, L. (2024) Improved Depthwise Separable Convolution for Transfer Learning in Fault Diagnosis. *IEEE Sensors Journal*, **24**, 33606-33613. <https://doi.org/10.1109/jsen.2024.3432921>
- [19] Li, H., Li, J., Wei, H., et al. (2022) Slim-Neck by GSConv: A Better Design Paradigm of Detector Architectures for Autonomous Vehicles.