# 选题意义：

在高质量发展的新时代，我国钢铁产业逐步走向产业结构升级和产品质量提高的新方向。作为沟通金属产品与钢铁冶炼的重要中间环节，带钢的品质直接关系产业下游机械产品的品质。而带钢又是相对表面积最大的一种钢材产品，且多用于外围构件，因此表面质量对带钢质量影响较大。例如，裂纹、斑块缺陷会显著破坏带钢的耐腐蚀性与强韧性，导致产品瑕疵多、报废率高、生产效率低。因此在带钢生产线上，通过表面缺陷检测筛选出不合格钢带回炉重造，能有效提升其出厂品质和产品精度，进而促进高效、绿色的钢铁产业发展。

目前主流的带钢表面缺陷识别算法[38-41]利用深度学习算法，通过端到端的有监督学习在特定数据集[1-12]上有较好的缺陷识别性能。然而，由于此类有监督学习固有的强数据依赖的特性，现有的网络结构往往针对特定数据集手工设计并优化。尽管此类设计可以显著提升对应数据集上的检测性能，但也使得模型结构无法在跨数据集时保持泛化能力。为了在不同数据集上得到优秀的检测能力，就需要利用专家知识针对性做网络结构调整，来适配数据集。这使得在线学习等方式无法有效缓解传感器参数漂移问题，增加模型维护成本。

这主要来源于研究者在训练模型时，对带钢表面缺陷识别数据集[1-10]使用了错误的假设：不同数据集的样本具有相似性。这一错误假设来源于不同数据集之间的差距较小的图像分类任务，比如CIFAR-10数据集[44]和ILSVRC数据集[43]。它们的采集场景都极为相似，采集设备均为具有相同标准的民用相机，样本相似度高。因此针对某个数据集设计出的模型结构能在不同数据集上保持可观的泛化能力。对此类任务，在新的数据集上部署模型时，往往只需要对预训练模型拟合几个训练轮数，就可以获得较好性能。但工业化的带钢表面缺陷数据集[1-10]中，其样本采集时的光源分布、传感器类型、预处理方式、程度等重要环境设置差距较大。且不同数据集中缺陷种类多样、纹理类型、尺度等特征也明显不同。因此数据集上的统计标签分布差异、纹理提取需求差异显著削弱了同一个模型结构在不同数据集间的泛化能力。

2017年以来，网络结构搜索（NAS）[17]作为一种自动寻找最优网络结构的算法被广泛研究。由于进化算法容易跳出局部最优、无梯度限制等特点，其被广泛应用于NAS的优化算法环节。进化深度学习利用进化算法（EA）来进行网络结构搜索，通过网络编码将网络结构优化问题用种群迭代来求解。此类方法所需超参数少，且对数据集变化不敏感，因此可以有效解决此领域深度学习算法遇到的瓶颈，是十分具有科研和应用价值的一项课题。

# 研究现状概述：

## 从传统算法到深度学习算法

早期的钢铁质量检测主要由质检员完成，存在时效慢、人力成本高等问题。因此，使用更高精度、更快速度、更规范化的计算机视觉缺陷识别方法对钢铁产业效率提升和成本优化具有重要作用。

随着传统计算机视觉和统计学习方法的发展，人工构建图像特征提取器（边缘检测[51]、小波变换[47]、滤波等）与后端统计学习预测器（Adaboost[48]、SVM[49]、决策树[50]等）结合的方案成为第一批基于视觉的带钢表面缺陷检测方案。然而，传统算法的性能高度依赖图像处理中的人工调参，需要丰富的图像处理领域专家知识。同时，传统算法手工设计特征的思路使其对于复杂形状缺陷、环境光变化、传感器参数漂变等情况的鲁棒性较差，检测性能低。

2014年后，伴随着VGG[32]、GoogleNet[31]、Resnet[14]、DenseNet[30]等高效深度学习网络的发展，基于深度学习的带钢表面缺陷识别[38-40]逐渐被广泛应用。此类模型具有特征提取能力强、准确率更高、构建难度更小、参数依赖更弱等特征。

### 带钢表面缺陷检测数据

对于有监督的深度卷积网络，其训练效果很大程度上依赖于数据集。由于数据集质量直接关系到最终算法部署时的性能，选取数量大、标注准的数据集极为关键。

在带钢表面缺陷检测领域，东北大学的宋克臣于2013年首先提出的NEU Surface Defect Database数据集[1]使用最为广泛。数据集简称NEU-CLS，共包含六种缺陷（轧入氧化皮RS、斑块Pa、裂纹Cr、麻面PS、夹杂物In和划痕Sc），每种类型均包含 300 张图像。此数据集中样本存在类内缺陷在外观上存在较大差异、类间缺陷具有相似性、不同样本间光照变化、不同样本材质变化等特点。 [2018年广东工业智造大数据创新大赛](https://tianchi.aliyun.com/markets/tianchi/industry?spm=5176.12281976.0.0.520766f05woH66)中，阿里巴巴提出铝型材表面瑕疵识别数据集，针对裂纹、起皮、划伤、腐蚀和气泡等瑕疵构建了8000张2560×1920像素的数据集。训练集（3000张）中，无缺陷图片1351张，单瑕疵图片2281张，多瑕疵图片229张。不同类别样本的数量之间存在样本不均衡问题，不同类别的物体尺度也分布不均衡。在2019年Kaggle举办的Steel Defect Detection比赛中，Severstal公司提出了Severstal Steel Dataset数据集[6]。此数据集来源于俄罗斯钢铁工业数据湖，包含5506个用于目标检测任务的样本和四种类型的缺陷。He等人[2]于2020年提出NEU-DET数据集，将 NEU-CLS中的数据进一步细化标注，提供用于目标检测任务的Bounding-Box标签。同年，Song等人[5]提出了用于图像分割的SD-saliency-900数据集，对夹杂物、补丁和划痕三类缺陷提供像素级标记。Bao等人[4]提出Surface Defects-4i表面缺陷分割数据集，包含铝、钢、铁轨和磁砖等属于常见的金属表面缺陷及其对应像素级标签。同时增加了非金属类（皮革和瓷砖）作为扩展数据，来进一步证明模型泛化能力。Niu等人[53]提出了用于钢轨表面缺陷检测的RSDDS-113数据集，其中的缺陷由轨道表面检测领域的一些专业人员进行了标注。2021年，Feng等人[12]参考NEU-CLS提出了改进的X-SDD数据集。用于缺陷的分类任务的X-SDD数据集包含七种典型的热轧带钢缺陷，共 1360 个样本。Zhang等人[52]提出了Rail-5k，包含最常见的13 种铁路缺陷类型的5000 多张高质量图像，其中1100 张图像有注释 。2022年，Severstal公司在AIA Manufacturing Project - Defect inspection比赛中提出了改进的Severstal Steel Dataset数据集[7]。

由于带钢表面缺陷检测领域的数据集较少，背景近似的其他场景数据集也广泛被用于带钢表面缺陷检测模型的训练和评估。最早的DAGM 2007[11]是多种工业背景下各类纹理的表面缺陷图像数据集，基于工业场景的光学纹理模型和缺陷模型生成人工生成。组成6个数据集组成，每个数据集包含 1000 张纹理无缺陷的图像，以及 150 张有一个标记的缺陷。Kolektor Surface-Defect Dataset（KolektorSDD）数据集[8]是根据由Kolektor Group doo采集并标注的电子换向器（金属）表面缺陷图像构建的，其中包含399张大小为 500×1250像素的图像，标注精度仅适用于图像分类任务。2021年，该实验室进一步为此数据集增加了Bounding-Box标签，使其可以被用于目标检测任务。同年，该实验室在KolektorSDD基础上提出实际工业场景下采集的 KolektorSDD2数据集[9]，其中包含 3000 多张包含多种类型缺陷（划痕、小斑点、表面瑕疵等）的图像，均与带钢表面缺陷检测领域相似。KolektorSDD2采用像素级别标签，可用于缺陷的语义分割。Mishra等人[10]提出的BTAD（ beanTech Anomaly Detection）数据集采集于真实工业生产场景的3种工业产品。数据集共2830 张图像，展示了零件主体和其表面的缺陷。

通过以上数据集的综述，发现钢铁表面缺陷检测存在如下问题：

### 样本数量不足

相比于ImageNet数据集中1400多万张样本数据, 表面缺陷检测中面临的关键的问题是样本数据不足,。目前开源的实际场景钢表面缺陷检测数据集只有几百张或几十张有效的缺陷图片。这类问题被称为在机器学习中被称为小样本学习（few-shot）。

### 数据集间差异大

钢铁表面缺陷检测并没有一个统一的，大规模的数据集，不同的缺陷检测数据集，在样本数量，正负样本比例，复杂度等方面都有很大的不同。此类任务数据集在采集对象（热钢表面[1]、金属物体表面[8]、钢卷、理论拟合模型[11]）、传感器（X光[12]、RGB图像传感器[4]、灰度图像传感器[8]）、光照类型和强度（自然光、结构光、点光源）、标注类型（图像分类[1]、目标检测[2]、图像分割[9]）、纹理类型等诸多参数上存在差异。而用于模型部署的实际场景数据集也存在类似问题，和论文中使用的NEU-CLS等数据集差异较大。

|  |  |
| --- | --- |
| 参数 | 类别 |
| 采集对象 | 热钢表面[1]、金属物体表面[8]、钢卷、理论拟合模型[11] |
| 传感器（采集设备） | X光[12]、RGB图像传感器[4]、灰度图像传感器[8] |
| 光照类型和强度 | 自然光、结构光、点光源 |
| 标注类型 | 图像分类[1]、目标检测[2]、图像分割[9] |
| 纹理类型 |  |
| 分辨率 |  |

如果不能设计使模型架构在类型差别巨大的数据集之间具有泛化能力的系统，带钢表面缺陷检测模型往往需要在每个实际部署的生产环境单独进行人工设计，不仅依赖专家知识，提升了人力成本，还导致模型样本采集的实际场景变化非常敏感。例如，对相机摆放倾斜角度变大、与钢带距离变小都会导致采集图像中缺陷尺寸变大，从而降低已有算法识别性能。而如今目标检测、图像分割模型往往针对小尺寸物体做网络结构上的优化（由于模型性能更多取决于难样本的识别能力，而小物体是典型的难样本），使其关键特征图更多表征感受野更小、更偏向网络底层的物体。有些模型还会增加跳跃连接、限制网络层数来提升对小目标的识别能力。这导致在数据集样本中缺陷尺寸变大后，仅用已有模型重新拟合或是更换数据集重新训练不足以达到最好的检测效果。

一般来说，深度神经网络（DNN）的性能取决于架构和对应的权重两个部分。以往的研究[1-16]仅关注了如何通过学习获取最优权重，从而获得对应数据集的最小损失。它们忽略了网络结构也需要对数据分布变化做出对应调整，而这是已有的深度学习方法是无法实现的。（分布外目标检测、域适应需要人为更换模型结构）

分辨率对网络结构（比如网络深度）的影响。用于说明NAS而不是域适应的效果。

实验表明，对于特定的任务类型、数据集特性、网络架构模式，图像分辨率对于神经网络深度的需求存在相关性。较高的图像分辨率需更深的神经网络来获得更好的性能，而较低的图像分辨率需要使用较浅的神经网络防止参数量过大而过拟合。这是因为高分辨率图像包含更多的细节和信息，需要更多的层次和复杂性来捕捉这些信息。而低分辨率图像由于信息较少，相对简单，可能不需要过多的网络深度。

为了进行实验，我们固定了网络的其他参数（如网络宽度、感受野等）并使用特定的网络架构（如金字塔网络、多尺度网络等）。

经典的VGGNet、GoogLeNet和ResNet等网络架构的研究中，通过在不同深度的网络中进行实验，发现较深的网络在高分辨率图像上能够获得更好的性能。这表明高分辨率图像可能需要更深的网络来捕获更丰富的特征信息。

此外，随着图像分辨率的增加，图像中的细节和信息量也增加，从而对网络的感知能力和表达能力提出了更高的要求。因此，较高的图像分辨率可能需要更深的网络来处理更复杂的特征和模式，以实现更好的性能。

这些论文以及其他相关研究表明，对于高分辨率图像，较深的神经网络通常能够获得更好的性能。然而，具体的结论仍需根据任务类型、数据集特性和网络架构等因素来确定，因此在实际应用中，需要进行具体的实验和分析。

防止每一次部署都需要重新确定网络结构，就需要NAS搜索。

### 依赖专家知识

人工设计 CNN 需要相当多的网络架构设计和CV领域的专业知识。对于不同数据集、不同领域、不同需求，CNN的网络结构都需要做适配性的调整。比如，缺陷检测领域中常用灰度、CT、X光等数据，其图像背景单一、色彩和结构不丰富、目标物体的结构相对固定、语义特征不丰富。因此所用网络不宜太深，参数过多，否则过拟合。

所提出的算法在设计 CNN 架构时是完全自动的。特别是，既不需要开始前的预处理，也不需要 CNN 的后处理。

## 现有方法

部分工作从拓展已有数据集入手，如Yang等人[13]使用旋转、镜像和高斯卷积处理等基本图像处理人工拓展数据集。同时为了量化增强效果，使用MSR增强图像的灰度标准差和信息熵两个指标评估。然而，此类方法仅针对没有数据优化的早期深度卷积网络有提升效果。目前的主流深度学习网络都有集成数据增强模块，比如ResNet[14]中的随机缩放裁剪、水平翻转，YOLOX[15]中的Mosaic、Mixup和DeepLabV3[16]的旋转、平移、Cutout。重复的无效扩增数据集不但没有性能提升，反而使得网络存在训练时间长、过拟合风险加大的问题，实用性较低。

## 网络结构搜索

网络架构搜索（NAS）适用于针对特定数据集自动设计最好的深度神经网络（DNN）的架构，可有效解决上述问题。其无须领域内专家知识和网络设计知识来针对网络结构中的大量超参数进行设计，仅需设定搜索范围、数据集、评价指标和搜索算法即可在多个数据集上自动寻找最适应的网络结构。2016年Zoph首次提出了基于强化学习（RL）的图像分类NAS算法[17]。尽管此算法由于搜索空间过大、优化算法和评价指标不合理等问题，需要数千张GPU搜索数天，但这一全新的思路仍开启了NAS算法研究的新时代。后续的工作尝试了基于梯度算法的NAS[18]和基于进化算法的NAS[19]。

在其中最关键网络寻优算法中，进化算法（EA）使用非常广泛。EA是一类基于种群的优化求解器，通过模拟自然界中物种的进化（交叉、变异、选择），以快速解决具有各类的优化问题。其中使用最广的是遗传算法 (GA) [20]、遗传编程算法 (GP) [22] 和粒子群优化算法 (PSO) [21]。由于 EA方法对局部最小值不敏感且不需要梯度信息的特性，它已被广泛应用于解决复杂的非凸优化问题[23]。这些难求解析解的任务通常不存在数学表达式形式的目标函数[24]（如NAS问题），或者搜索空间的计算复杂度过大的NP难问题[25]（如TSP问题）。

在早期的NAS算法[26]中，神经网络的权重和结构都在同一个搜索过程中完成。然而，这样的搜索空间太大，仅适用于类似于BP神经网络等浅层网络和表格化少量数据。对于计算机视觉任务中的深度卷积神经网络（DCNN），这样完整搜索的GPU耗时将完全无法接受。Darwish 等人[28]的工作还关注基于进化算法的NAS和超参数组合优化上，但超参数搜索会显著增加训练耗时。Stanley 等人[29]综述了进化神经网络的发展，揭示了权重优化的重要性，但忽略了网络架构的决定性作用。近年主流的方法[27]关注于仅搜索网络结构，网络的具体权重参数仍由梯度下降（SGD）的训练过程完成。

在基于进化算法的深度神经网络架构搜索领域，2017年Real的工作[27]可谓开山之作。其架构可分为搜索空间、搜索策略和性能估计策略三个大部分。大致流程如下。首先，算法在预先定义的初始空间内初始化一个种群。群体中的每个个体都代表NAS的一个解决方案，即一种可能的DNN 架构。每个预选架构在加入种群之前都需要被编码为一个个体。然后，算法将根据对应评价指标给生成个体的适应度打分。依据评价结果，整个种群在搜索空间内依照搜索策略开始进化过程。种群在每次迭代中通过选择和进化算子进行更新，直到满足停止准则。

通过对VGG[32]、GoogleNet[31]、Resnet[14]、DenseNet[30]创新点的总结，可以发现对于深度卷积神经网络（DCNN）而言，其性能主要取决于CNN架构的三个方面：每一层的超参数[34]（例如卷积层的核大小、池化类型和全连接的神经元数量）、架构的深度[33]和层与层之间的连接[35]（如密集连接和跳跃连接）。

通常来说，某一层的超参数搜索空间分为Micro Search Space和Macro Search Space两类[37]。其中Micro Search Space [36]只搜索几种已证明高效的固定拓扑关系的Block构成的搜索空间，如GoogleNet[31]的Inception Block、DenseNet[30]的Dense Block和Resnet[14]的Res-Block。他们都具有良好的性能，并且能降低参数编码长度。这些模块已用于诸多网络结构的手工设计，被广泛的研究证明是有效的。Macro Search Space [17]搜索所有可能解，其设计出的结构更具特异性、性能较好。但过大的搜索空间导致计算量指数级上升，通常需要服务器集群来进行训练。

为了约束搜索空间来减少计算量，固定深度、高质量的初始化和固定部分结构是常用的方法。其中固定深度是一个强约束，大大减小了编码空间的大小，但也可能导致算法无法达到最优解[36]。高质量的初始化[37]通常使用已知性能较好的网络作为起点，来进一步搜索更优个体。固定部分结构[45]指搜索过程中默认一些模块的组合，比如几个卷积层之后将一个最大池化层[31]或1x1卷积来汇聚信息、加入层均值化和通道均值化。

综上所述，尽管在带钢表面缺陷研究中已有大量深度卷积神经网络模型在特定数据集上取得了较高的性能，但带钢表面不同数据集间的巨大差异使得手工设计网络对其他数据集缺少泛化能力。基于这一情况，发展已较为成熟的NAS在解决此类跨数据集的网络结构优化问题上具有较高潜力。其中，基于EA的进化深度学习不需要梯度信息、能够跳出局部最优、计算量小，是非常理想的解决方案。

# 主要研究内容：

**数据预处理：**

针对已有数据集[1-12]来进行以下操作：

数据清洗，筛选出数据中无标签对应的样本并剔除出数据集，防止后续模型训练、推理过程中出现数据加载错误。

数据增强，尽管已有大多数网络都具有自定义的数据扩增操作，但进行NAS的大规模训练时数据增强的操作将被反复进行，消耗CPU资源并大量占用硬盘读取串口。可使用的方法包括ResNet[14]中的随机缩放裁剪、水平翻转，YOLOX[15]中的Mosaic、Mixup和DeepLabV3[16]的旋转、平移、Cutout。

数据划分，使所有数据集满足k折交叉验证所需的数据要求。减少复制等所需时间，同时保证模型在不同数据集上的训练具有控制变量的可比性。

数据验证，部分数据集[4]中的样本存在漏标现象。通过已有性能较好的模型对数据集执行推理，再设置合适阈值就可以筛选出置信度较低的可疑样本。最后再利用CAM特征图可视化[46]等方式，对照存疑图像及其对应标签，就可以排除掉标签错误、遗漏等样本。

数据转化，将用于分割的像素级标签转化为用于目标检测的Bounding Box标记，将用于目标检测的Bounding Box标签转化为图像分类标签。这样就可以显著增加缺陷图像分类任务、缺陷目标检测任务的数据量，有助于模型训练和对比。

**搭建进化深度学习的模型:**

 通过进化算法和目标检测两个方向进行文献综述，找出目前主流模型和SOTA模型进行复现，搭建基于进化算法的图像分类、目标检测算法框架。在统一的实验环境下对比测试已有模型，并通过消融实验结合对应文章找到其模型的关键点和提升性能的Trick。

**进行模型改进尝试：**

对已经形成的基线模型，通过热力图可视化[46]等方式挖掘难样本，找出缺陷检测模型改进的方向。针对进化算法环节，实现GA、PSO、ACO等不同算法。在控制变量的条件下（如固定GPU时、固定计算量等）进行对比实验，并进一步探索各算法对性能影响的原因。

## 采用的研究思路：

在开始设计前，首先学习理论知识储备与补充，阅读机器学习、统计学习方法、带钢生产以及Python、Pytorch等相关书籍。同时做好笔记，标记本次设计可能需要的知识内容。

做好相关知识储备后，阅读大量相关领域的文献，包括NAS、目标检测、图像分类以及优化算法理论等等。在充分阅读文献后，通过文献综述总结学者专家研究思路，从而开拓自己的想法，找到模型改进的创新点。

对于数据预处理，参考已有模型的算法实现，使用C++和OpenCV完成。

对于搭建好的模型尝试多方面改进，如进化算法的设计、进化算法与NAS的融合策略等。

对得到的模型各模块进行消融实验并论证提出方法有效性，总结完成论文。

# 带钢表面缺陷检测与数据预处理

## 带钢表面缺陷检测

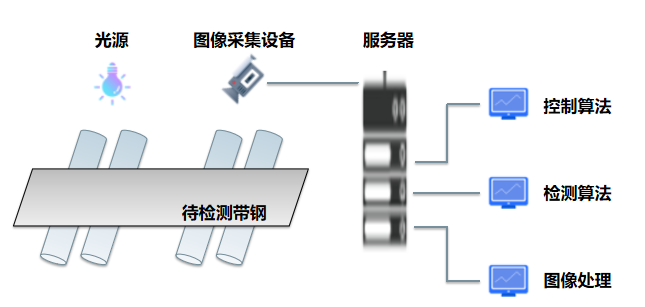
### 背景（工艺介绍）

带钢是钢铁工业中的重要产品，广泛应用于汽车、家电、建筑和其他领域。为了确保高品质的产品，带钢在生产过程中需要进行严格的表面缺陷检测。传统的人工检测方式效率低下且易受人员疲劳影响，因此，基于计算机视觉的带钢表面缺陷自动检测技术逐渐受到关注。

在一般的工业场景中[1]，缺陷类型可分为几何缺陷和表面强度缺陷或密度缺陷两类。其中几何缺陷包括：凹坑、划痕、裂纹、毛刺、凸起。表面强度缺陷或密度缺陷如：氧化、生锈和污渍。几何缺陷改变表面反射，表面强度缺陷或密度缺陷改变表面反射和吸收。

夹杂物缺陷 (Inclusion)，红铁皮缺陷 (Red iron sheet)，划痕缺陷 (Scratches)，裂纹缺陷 (Crazing)，斑块状缺陷 (Patches)，凹坑表面缺陷 (Pitted surface)，鳞片状氧化物表面缺陷 (Rolled in scale)。夹杂物缺陷缺陷会导致钢铁的强度和韧性下降，从而影响其使用寿命和安全性；Red iron sheet缺陷会导致钢铁表面出现凹坑，影响其外观和表面质量；Scratches缺陷则可能导致钢铁表面出现裂纹或凹槽，影响其机械性能和耐腐蚀性能。其他缺陷类型也会对钢铁产生不同程度的影响。因此，在生产过程中需要尽可能地避免这些缺陷的产生，并采取相应的措施来修复已经存在的缺陷。

### 检测系统



通过回顾已有文献[1,13]发现，带钢表面缺陷检测系统的硬件和软件通常包含光学传感器、光照系统、图像处理和分析系统、控制系统、数据存储和管理系统、人机交互界面、带钢传送系统等几个关键组成部分。

#### 图像采集设备

图像采集设备用于采集和处理带钢表面的图像数据或视频数据，是检测系统中的核心组件。图像采集设备通过感光元件类型、相机摆放位置、镜头焦距和到带钢表面的距离等因素，基本决定了采集图像的粒度、信息量、纹理类型、特征粒度等重要因素。传感器参数差异是不同数据集间采集图像差异大的主要原因，主要体现在以下几个方面。

#### 感光元件

感光元件是图像采集设备中价格最高、最重要的元件。在采集不同数据集时，最大的差别普遍来源于感光元件的分辨率、信噪比控制。比如X-SDD中分辨率为128×128，而NEU-CLS中分辨率为200×200。其差别还表现在如下的更多属性上。

感光元件类型：主流的工业检测传感器有CCD（Charge-Coupled Device）传感器、CMOS（Complementary Metal-Oxide-Semiconductor）传感器、红外线（Infrared）传感器。CMOS有更低的功耗、更快的读取速度和更便于集成的特点。红外线探测物体表面的热能辐射，生成对应的数字信号，并通过相关算法来检测物体表面的缺陷或异常情况。红外线传感器在检测金属、塑料、陶瓷等物体的表面缺陷时具有很好的效果，被广泛应用于工业生产中。

分辨率：分辨率是指图像传感器可以捕捉到的最小细节。它通常以像素为单位表示，即指在水平和垂直方向上包含多少个离散的纪录单元。分辨率越高，图像细节越清晰，但相应的数据量和处理要求也会增加。 带钢的表面通常会存在一些微小的凹凸不平和颗粒等细微的缺陷，这些缺陷需要使用较高分辨率的相机进行拍摄。

灵敏度：灵敏度是指图像传感器对光信号的响应能力，即在特定光照条件下，产生的电信号与光强度之间的关系。灵敏度越高，传感器对光信号的响应能力越大，能够捕捉到更微弱的光信号，从而提高图像质量。

帧率：帧率是指图像传感器每秒钟可以采集和传输的图像帧数。帧率越高，图像更新速度越快，能够捕捉到更快速度的物体运动和变化，从而提高检测精度和效率。

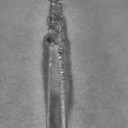
噪声：噪声是指图像传感器产生的随机扰动信号，可能会影响图像质量并进一步干扰细粒度缺陷的检测结果。噪声分为暗噪声和热噪声两种类型，可以通过降噪技术或优化传感器设计来降低。

摆放位置：摆放位置决定了相机拍摄被测物的角度和距离，过远或过近都可能导致图像失真或无法清晰识别。同时，由于检测算法通常依赖于光源、相机和被测物品之间的位置关系，因此机器摆放位置与缺陷检测数据类型高度相关。选择相机距离时需要考虑被检测物品的大小、形状、缺陷类型和检测要求等因素。选择位置也应考虑照明均匀性问题。不同位置的光照强度和角度不同，可能导致光照不均匀，从而影响缺陷检测的准确性。在选择机器摆放位置时应尽量避免遮挡和光照反射等问题。

#### 镜头

光学系统的镜头在整套系统中属于间接影响因素，在镜头光轴无偏、焦平面正确、相场均匀的情况下不会对图像采集质量造成影响。但镜头的选型不同，系统的其他部分也会发生相应的变化。

检测范围：镜头焦距较小的镜头通常具有较大的视场角，可以覆盖较大的检测范围，适用于检测较大面积的钢表面缺陷。而焦距较大的镜头则视场角较小，适用于检测较小面积的钢表面缺陷。

角分辨率：镜头焦距较小的镜头通常具有较低的放大倍率，对表面缺陷的细节分辨率可能较低。而焦距较大的镜头通常具有较高的放大倍率，可以提供更高的细节分辨率，对小尺寸的缺陷检测更为敏感。

焦深：焦距较小的镜头通常具有较大的焦深，即在一定距离范围内，物体可以保持清晰焦距。这对于钢表面缺陷检测来说可能不利，因为焦深较大时，表面缺陷与背景的对比可能较低，导致缺陷检测的灵敏度降低。而焦距较大的镜头焦深较小，可以提供较高的对比度，对缺陷检测更为有利。

图x：NEU-CLS和X-SDD中slag inclusion类别不同尺度的缺陷

检测速度：焦距较小的镜头通常适用于大范围的表面缺陷检测，可以提供较快的检测速度。而焦距较大的镜头通常适用于小范围的高分辨率缺陷检测，可能需要更长的检测时间。

操作距离：焦距较小的镜头通常可以在相对较远的距离内进行缺陷检测，适合远距离或不便靠近的检测场景。而焦距较大的镜头通常需要较近的操作距离，适合近距离的缺陷检测。

#### 光照系统

用于提供足够的光源，以确保带钢表面的缺陷在图像中能够清晰可见。光照系统可以包括不同类型的灯光，如白光、红外光等，以便对不同类型的缺陷进行检测。

光照系统在缺陷检测起到了重要的作用，通过对光照条件进行控制和调节，对缺陷检测的性能产生影响。光源类型和强度影响图像质量中的信噪比、光强度场均衡性、对比度、颜色等视觉特性，对图像数据有重要影响作用。光照的选择需要根据材料特性和目标缺陷特征选择。

光照对缺陷检测的影响：光照条件对缺陷检测的性能有着直接的影响。光照不均匀、光照强度不一致、光照方向不同等因素都可能导致图像中的缺陷被掩盖或者误检，从而降低了缺陷检测的准确性和可靠性。

图x：相同数据集下采集亮度变化

金属是常见的反光率较高的材料，因此需要精心设置光源。

金属带钢的表面反射性很强，因此需要选择适当的光源和角度，以减少反射和阴影的干扰，从而提高缺陷检测的准确性。

在生产线上进行实时检测，因此光照系统还需要具备快速、稳定和可靠的特性，以满足实时检测的要求。

带钢在生产线中是通过带钢传送系统不断滚动的，其运动速度v和镜头焦距f决定了在特定分辨率下安全快门速度（最低曝光时间）的大小。一般来说，要求快门速度不低于1/f\*1/v\*粒度参数\*分辨率参数秒，以此来保证图像清晰。这就要求光照强度符合特定光圈和ISO下的亮度指标。

不同的缺陷类型和大小需要采用不同的安全快门速度进行检测。一般来说，较小的缺陷或者需要进行高精度的定位需要使用更高的安全快门速度，而较大的缺陷则可以使用较低的安全快门速度。

光照系统类型

光照系统的种类：光照系统在缺陷检测领域有多种不同的类型，包括均匀光源、点光源、环形光源、光纤光源等。不同类型的光照系统适用于不同的缺陷检测场景和需求，可以根据具体应用的需求来选择合适的光照系统。

光照系统的参数：光照系统的参数包括光照强度、光源距离、光照方向、光照颜色等。这些参数的选择和调节对于缺陷检测的性能有着重要的影响。例如，在光照强度较低的情况下，可能导致图像中的缺陷细节被掩盖，而在光照强度较高的情况下，可能会出现过曝现象。因此，合理设置和调节光照系统的参数对于缺陷检测的准确性和可靠性非常关键。光源设置一般需要生产现场临时调试，但通常来说，光源距离与光表面符合平方反比定律。即强度与距离的平方成反比。光源距离物体的距离加倍时，在相同曝光条件下，物体表面接收到的光线强度将减少至原来的四分之一。

下面是一些常见的光源设置。

环形光源：环形光源是一种常见的光照系统，它能够提供均匀的光照，并且可以减少阴影和反射，从而提高图像质量和缺陷检测的准确性。

图x：照明均衡情况下的缺陷，照明不均衡情况下的缺陷

平行光源：平行光源是另一种常见的光照系统，它可以提供明亮的光照，并且可以减少畸变和阴影，从而提高缺陷检测的准确性。

闪光灯：闪光灯是一种常用于摄影的光照系统，它可以提供强烈的、短暂的光照，从而产生高质量的图像，但由于其强烈的闪光会导致严重的反射和阴影，因此在表面缺陷检测中应用较少。

红外光源：红外光源是一种特殊的光照系统，它可以透过一些材料和涂层，从而在检测缺陷时具有一定的优势。但由于其工作原理较为复杂，因此应用较少。

激光光源：激光光源是一种强烈而准确的光照系统，它可以产生高质量的图像，并且能够精确定位缺陷位置，但由于其价格昂贵且使用较为复杂，因此在实际应用中应用较少。

光照系统的控制方式：光照系统可以通过手动方式或自动方式进行控制。手动方式需要操作人员根据实际需求来手动调节光照系统的参数，而自动方式则可以通过传感器、反馈控制等方式来实现对光照系统的自动调节，从而保持图像的一致性和稳定性。比如闪光光源下的TTL自动测光曝光模式，能实时、全自动衡量所需光量，是常用的闪光光源控制方案。

RGB光源和单色光源在光照系统中的需求存在一些区别，包括光源颜色控制、光源稳定性、光源调节精度、成本和复杂性等方面。

RGB光源在颜色控制的同时需要保持三个通道的光强度稳定，避免颜色偏移或者光强度波动对缺陷检测结果产生干扰。而单色光源只需保持单一颜色的光强度稳定，相对较为简单。

#### 图像处理和分析系统

用于对从光学传感器或摄像头获取的图像或视频数据进行处理和分析。图像处理设备主要是指计算机及其配套设备，如GPU、内存等。其中算法可以包括图像增强、边缘检测、形状识别、纹理分析等，以便自动地检测带钢表面的缺陷。

#### 控制系统

用于控制光学传感器、光照系统以及图像处理和分析算法的运行。控制系统可以包括硬件控制器和软件界面，以便操作员能够对检测过程进行设置和调整。

#### 数据存储和管理系统

用于保存带钢表面缺陷检测过程中生成的图像、视频和检测结果数据。这些数据可以用于后续的数据分析、报告生成和质量管理。进一步地，系统还可以利用在线机器学习、迁移学习等技巧，对上线后的模型进行升级。

#### 人机交互界面

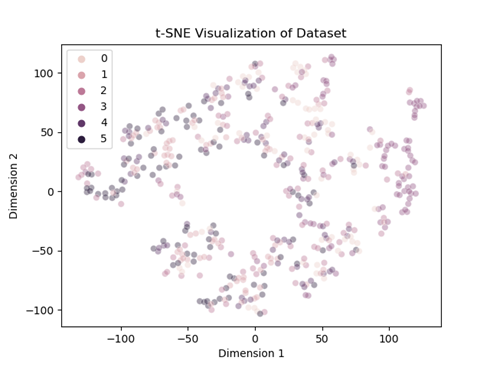
用于操作员与系统进行交互，包括设置检测参数、启动和停止检测过程、查看检测结果、在线互动标注等。

#### 带钢传送系统

用于将待检测的带钢送入检测区域，通常包括传送带、传动装置、定位装置等。传送系统需要确保带钢在检测过程中能够保持稳定的运动状态，以便获取清晰的图像或视频数据。

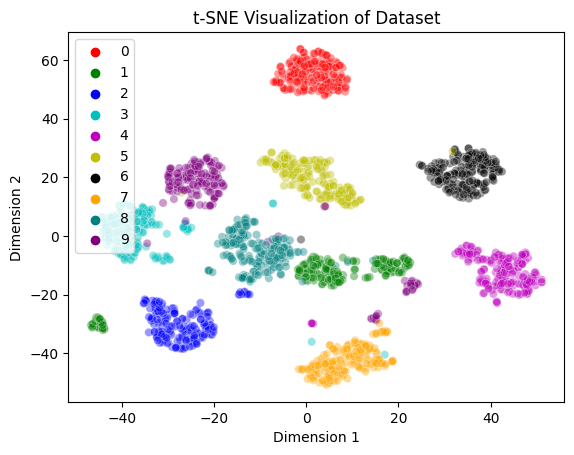
## 带钢缺陷检测数据集

### NEU-CLS



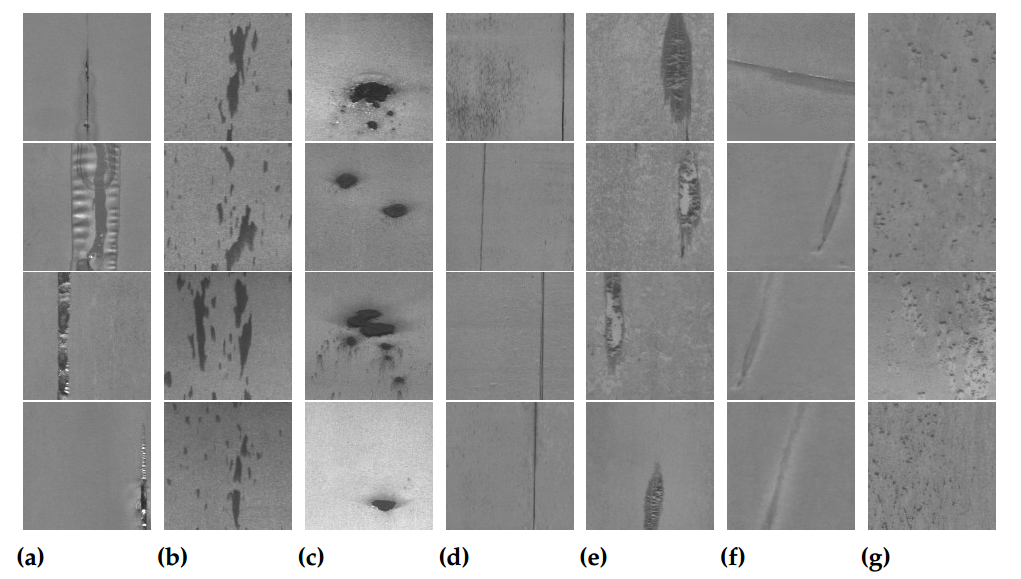
t-SNE（t-Distributed Stochastic Neighbor Embedding）是一种非线性降维算法，可以将高维数据映射到低维空间中进行可视化。该算法通过学习高维数据的局部结构和全局结构，将相似的数据映射到低维空间中的近邻点，并将不相似的数据映射到远离点。这样，t-SNE可以在保留数据结构的同时，将高维数据可视化为二维或三维图形，帮助我们更好地理解和分析数据。

在本文中，我们使用Scikit-learn库中的t-SNE算法对一个包含6个类别、每个类别100个样本的数据集进行可视化。我们通过Seaborn库对可视化结果进行美化，并编写了一个评分函数来衡量不同类别之间的混合程度。这些技术和工具可以帮助我们更好地理解和分析数据。



从中我们可以观察到，其特征分布不均匀，很难被聚类算法分类。

### X-SDD



夹杂物缺陷如图 3a 所示，通常发生在板坯连铸过程中。它们是由于板坯中夹渣引起的大量夹杂物的存在而形成的，这些夹杂物在随后的热轧过程中被扩展和暴露。夹杂物缺陷的特征是一种可见的黑色非金属物质，与周围的金属有明显的色差。结渣缺陷严重的钢带通常需要切断，而结渣缺陷较小的钢带有时可以通过人工抛光去除。

缺陷类型包括以下七种：夹杂物缺陷 (Inclusion)，红铁皮缺陷 (Red iron sheet)，划痕缺陷 (Scratches)，裂纹缺陷 (Crazing)，斑块状缺陷 (Patches)，凹坑表面缺陷 (Pitted surface)，鳞片状氧化物表面缺陷 (Rolled in scale)。

划痕：一般出现在钢带的下表面，全长和全宽随机分布，形成原因可能是热轧区突起或死辊、被动辊与钢带表面摩擦。外观特点是钢带表面缺陷呈直线状和沟槽状。

板系统的氧化皮：形成原因可能是高温高速轧制过程中，辊道被动转动、辊道死辊、辊道弯曲变形、轧辊磨损表面、轧件表面破损，导致氧化铁颗粒在随后的轧制变形过程中轧入轧件。外观特征为缺损位置基本固定，类似擦伤和挫伤。

后整理卷印：一般发生在宽度小于1200mm的边缘，并沿长度方向连续分布，形成原因可能是工作辊与支撑辊之间存在打滑，导致工作辊表面出现点状、短条状损伤。外观特征为点状和短条状凹坑，密布于同一部位。

温度系统的氧化层：形成比较复杂，可能由多种原因造成，包括轧制安排不合理、钢带含碳量高、积水使用不当、粗轧温度控制过高、钢带下表面温度高于上表面温度以及带钢通过精轧辊前，机架经过强烈氧化等。外观特征以松散或散砂为特征。

夹杂物缺陷 (Inclusion)，会导致钢铁的强度和韧性下降，从而影响其使用寿命和安全性

红铁皮缺陷 (Red iron sheet)，会导致钢铁表面出现凹坑，影响其外观和表面质量

划痕缺陷 (Scratches)，可能导致钢铁表面出现裂纹或凹槽，影响其机械性能和耐腐蚀性能。

裂纹缺陷 (Crazing)，呈现出细小的裂纹和网状图案，通常是由于钢板表面张力不均匀而引起的。

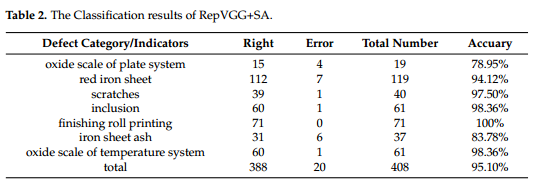
斑块状缺陷 (Patches)，呈现出较大的斑块状图案，通常是由于钢板表面氧化或其他污染物质所致。

凹坑表面缺陷 (Pitted surface)，呈现出许多小凹坑和凸起，通常是由于钢板表面存在氧化物、夹杂物或其他污染物质所致。

鳞片状氧化物表面缺陷 (Rolled in scale)。呈现出较大的鳞片状图案，通常是由于钢板表面存在铁锈或其他氧化物所致。

|  |  |  |
| --- | --- | --- |
|  |  |  |
| 夹杂物缺陷 (Inclusion) |  |  |
| 红铁皮缺陷 (Red iron sheet) |  |  |
| 划痕缺陷 (Scratches) |  |  |
| 裂纹缺陷 (Crazing) |  |  |
| 斑块状缺陷 (Patches) |  |  |
| 凹坑表面缺陷 (Pitted surface) |  |  |
| 鳞片状氧化物表面缺陷 (Rolled in scale) |  |  |

不同数据的权重不同从饼图中可以看出，由于实际生产中各种缺陷的范围不同，因此不同类型的缺陷数量存在差异。例如，红铁皮等缺陷可能广泛分布在单个钢卷上，因此可以采集大量样品；而板材系统的氧化铁皮等缺陷更容易去除。



the other is that the morphology of plate channel scale and slag inclusion is relatively close, which is prone to misclassification, which leads to several plate channel scale being classified as slag inclusion in the testset.

### X-SDD与NEU-CLS对比

X-SDD上的缺陷与NEU-CLS上的缺陷在形态上有所不同。

## 图像预处理与图像数据增强

图像预处理通过对图像做优化来提高模型的鲁棒性：数据增强可以通过引入不同的变化和变形，增加训练数据集的多样性，从而提高缺陷检测模型的鲁棒性，使其能够更好地适应真实工业生产中的不同场景和条件，减少过拟合的风险。

增加样本数量：在实际的缺陷检测应用中，获取大规模的缺陷样本数据可能会面临困难，而数据增强可以通过对少量样本进行多次变换生成更多的样本，从而增加训练数据集的规模，提高模型的训练效果。

改善模型的泛化能力：数据增强可以通过模拟真实世界中不同的光照、噪声、遮挡等情况，使模型更好地学习不同的缺陷特征，并能够在实际应用中更好地应对各种情况，从而提高模型的泛化能力，增强其在未知数据上的检测能力。

增加模型对缺陷多样性的识别能力：数据增强可以引入不同类型、不同形状、不同尺寸、不同位置等不同的缺陷样本，从而增加模型对不同缺陷多样性的识别能力，使其更好地适应实际生产中可能出现的各种缺陷情况。

提高缺陷检测的精度和稳定性：通过合理的数据增强策略，可以使缺陷检测模型更好地学习缺陷的特征，并且在检测过程中更加准确和稳定，从而提高缺陷检测的精度和可靠性。

图像数据增强和图像预处理都是图像处理中常用的技术，但它们的目的和方法有所不同。

图像预处理是指在对图像进行分析和处理之前，对原始图像进行一系列预处理操作，以提高图像质量、减少图像噪声、增强图像对比度、移除图像背景等。常见的图像预处理方法包括图像滤波、灰度化、二值化、去噪、图像平滑处理等。图像预处理的目的是为了更好地进行后续图像处理和分析任务，例如物体检测、图像分类等。

而图像数据增强是指通过一系列变换操作，增加训练数据集的样本数量，以提高模型的泛化能力和鲁棒性。常见的图像数据增强方法包括旋转、平移、缩放、裁剪、翻转、添加噪声、颜色变换等。通过图像数据增强，可以使模型对图像中不同的变换操作具有一定的鲁棒性，提高模型的准确率和泛化能力。

因此，图像预处理和图像数据增强虽然都是图像处理的技术，但是它们的目的和方法是不同的。图像预处理旨在优化图像的质量和特征，以更好地进行后续的图像分析和处理任务，而图像数据增强旨在增加训练数据的数量和多样性，以提高模型的泛化能力和鲁棒性。

### 预处理

#### 通用场景的图像预处理方法

直方图均衡化

直方图均衡化是一种用于增强图像对比度的非线性处理技术。它通过重新分配像素灰度值的概率分布，将图像灰度级范围拉伸到更大的范围，从而增强图像的局部对比度。在具体实现中，首先计算图像的灰度直方图，然后根据灰度值累计分布函数，重新分配像素灰度值，最终得到均衡化后的图像。

降噪

降噪是图像处理中常见的一项任务，其目的是去除图像中的噪声，从而提高图像的质量和准确性。降噪技术包括基于滤波器的方法和基于统计学的方法。其中，基于滤波器的方法通过卷积一个滤波器来平滑图像，并去除图像中的高频噪声。基于统计学的方法则是利用图像的统计学特征，如均值和方差等，通过数学模型来估计和去除噪声。

像素值归一化：

像素值归一化是将图像中的像素值通过一定的变换方式，将其映射到特定的区间内，以便后续处理和分析。最常见的像素值归一化方法是最小-最大归一化，该方法通过线性变换将像素值映射到[0,1]区间内。具体而言，最小-最大归一化将图像中的每个像素值减去最小像素值，再除以最大像素值与最小像素值的差值，从而实现归一化。其他常见的像素值归一化方法包括z-score归一化和均值方差归一化等。

图像平滑处理：

图像平滑处理是通过应用一定的滤波器来平滑图像并减少噪声。平滑处理可以采用线性滤波器，如均值滤波器、高斯滤波器等，也可以采用非线性滤波器，如中值滤波器、双边滤波器等。其中，高斯滤波器是一种基于高斯函数的线性滤波器，它可以平滑图像并保留图像的边缘信息。中值滤波器是一种基于排序统计的非线性滤波器，它可以有效地去除图像中的椒盐噪声和斑点噪声。双边滤波器是一种结合了空间域信息和像素值相似性的滤波器，它可以平滑图像并保留图像的边缘和细节信息。

#### 工业场景下特有的图像预处理方法

光照校正：

在工业场景中，由于灯光强度、角度等因素的影响，图像可能会产生光照不均匀的问题。光照校正技术能够对图像进行光照补偿，提高图像的对比度和清晰度。常用的光照校正方法包括基于灰度世界假设的白平衡算法、基于灰度估计的自适应光照校正算法等。

噪声去除：

在工业场景中，图像可能会受到各种噪声的干扰，如椒盐噪声、高斯噪声、条纹噪声等。噪声去除技术能够有效地去除这些噪声，提高图像的质量和清晰度。常用的噪声去除方法包括基于滤波的方法、基于小波变换的方法等。在工业场景中，传统的图像去噪算法包括基于滤波器的方法、基于小波变换的方法和基于稀疏编码的方法等。

其中，基于滤波器的方法常用于去除高斯噪声，常见的滤波器包括均值滤波器、中值滤波器、高斯滤波器等。这些滤波器通过在图像中对每个像素周围的像素进行平均、中值或加权平均等操作来降低噪声的影响。

基于小波变换的方法则通过将图像转换为小波域，利用小波变换的多分辨率特性来减小噪声的影响。通过对图像进行小波分解和重构，利用小波分解后的高频子带和低频子带之间的关系去除噪声。该算法具有简单、高效、易实现等优点，在带钢表面缺陷检测场景中应用广泛。通常使用小波阈值处理方法，将小波系数中低于某一阈值的系数置为0，从而去除噪声。

基于稀疏编码的方法则通过将图像表示为一组基函数的线性组合，并利用稀疏性来减小噪声的影响。通常使用稀疏表示方法，将图像表示为一组基函数的线性组合，并将系数表示为一个向量，通过对系数进行稀疏化处理来去除噪声。

基于邻域信息的去噪算法通过对图像中邻域像素信息的统计分析，确定噪声与信号的边界，进而通过对噪声区域进行滤波来去噪。常见的基于邻域信息的去噪算法包括基于中值滤波、基于均值滤波等。

这些传统的图像去噪算法在工业场景中应用广泛，具有简单、高效等特点。但是，这些算法往往需要手动调节参数，对不同类型的噪声效果有所差异，并且难以适应不同场景的需求。因此，我们在训练、验证过程中采用了参数、步骤完全相同的降噪算法，来保证模型评估的公平性。

在模型实际部署到工业场景中时，需要根据实际场景重新调整参数。

自适应图像去噪算法是一种在不同噪声场景下自适应调节参数的图像去噪方法。它能够在保持图像细节的同时，有效地去除图像中的噪声。其基本思想是通过对图像进行分析，自适应地调节去噪参数，以达到最佳的去噪效果。常见的自适应图像去噪算法包括基于图像局部方差的方法、基于非局部相似性的方法、基于深度学习的方法等。其中，基于图像局部方差的方法通过对图像的局部方差进行估计，并根据不同的噪声场景自适应地调整去噪参数，以达到最佳的去噪效果。基于非局部相似性的方法则是利用图像中相似块之间的相似性来自适应地调节去噪参数。基于深度学习的方法则是通过对大量图像进行训练，学习到图像中的特征信息，并自适应地调节去噪参数。自适应图像去噪算法能够有效地应对不同噪声场景下的图像去噪问题，并且不需要手动调节参数，具有更好的鲁棒性和适应性。因此，在工业场景中得到了广泛应用。

图像增强：

在工业场景中，有时需要对图像进行增强，以提高图像的对比度和清晰度。图像增强技术包括直方图均衡化、局部自适应直方图均衡化、对数变换、伽马变换等。其中，局部自适应直方图均衡化是一种针对图像局部区域的直方图均衡化算法，能够在提高图像对比度的同时避免出现过度增强和噪声放大的问题。

形态学处理：

在工业场景中，形态学处理技术经常用于图像的形态学分割、轮廓提取、形态学重构等方面。形态学处理基于形态学操作，如膨胀、腐蚀、开运算、闭运算等，能够有效地处理图像中的连通区域、边缘和噪声等问题，提高图像的清晰度和准确度。

这些工业场景下的特定图像预处理方法能够有效地提高图像的质量和可用性，对于工业自动化、质量控制和缺陷检测等方面有着广泛的应用。

标签清洗：

在带钢表面缺陷领域，数据集使用较少，缺乏公开验证和纠错。这些数据集可能存在许多错误、不一致或缺失的标签信息，可能会导致机器学习算法的训练效果不佳或预测精度下降。因此对于未经过验证的数据集，标签清洗至关重要。

图像数据集的标签清洗是指对标签进行筛选、纠正、修正等操作，以确保标签的准确性和一致性，从而提高模型的训练效果和推理精度。

标签清洗通常包括以下几个步骤：

数据预处理：对原始数据进行预处理，如格式转换、大小调整、色彩空间转换等操作，使其适合于模型训练和标签清洗。

标签筛选：筛选出符合要求的标签，比如去除错误标签、重复标签、无意义标签等。

标签纠错：对存在错误的标签进行修正，比如将标签中的错误标签替换为正确标签、将标签中的多余信息删除等。许多数据集中会包含一定比例的未打标数据，这一步需要对图像和标签取并集来保证推理和验证时模型得到正确的反馈。

标签补全：对存在缺失的标签进行补全，比如利用其他标注信息补全缺失标注、利用数据增强技术扩充数据集等。这一步需要编写脚本检查xml文档中的信息是否完全、格式是否正确。防止在后续训练过程中出现乱码等问题。

标签一致性检查：确保数据集中的标签具有一致性，比如对同一物体进行标注时，不同标注者应当标注相同的标签。

标签清洗对于图像数据集的训练和应用非常重要，可以提高模型的准确性和鲁棒性。但是标签清洗也是一项繁琐、耗时的工作，需要耗费大量的人力和时间。因此，一些自动化的标签清洗工具也在不断研究和发展中，以提高标签清洗的效率和准确性。

### 图像数据增强

#### 通用场景的图像数据增强

数据增强是深度学习中非常重要的一环，可以通过增加训练集的数量和多样性，从而提高模型的泛化能力和鲁棒性。在图像处理中，数据增强的方法也被称为图像增强。以下是常见的图像增强技术：

随机裁剪：

随机裁剪是指在训练图像中随机选取一部分作为输入图像，可以增加训练集的多样性，提高模型的泛化能力。同时，裁剪的位置和大小也可以随机生成，增加数据的随机性。

随机旋转：

随机旋转是指对训练图像进行随机旋转，从而增加数据集的多样性。可以将图像按照随机角度旋转，并根据需要进行插值处理，保证图像的质量。

随机缩放：

随机缩放是指对训练图像进行随机缩放，从而增加数据集的多样性。可以将图像按照随机比例缩放，并根据需要进行插值处理，保证图像的质量。

随机翻转：

随机翻转是指对训练图像进行随机水平或垂直翻转，从而增加数据集的多样性。可以根据需要随机进行水平或垂直翻转，或同时进行水平和垂直翻转。

随机颜色变换：

随机颜色变换是指对训练图像进行随机颜色调整，如改变图像的亮度、对比度、饱和度等，从而增加数据集的多样性。可以通过变换色调、亮度、对比度和饱和度等参数，实现图像的颜色变换。

添加噪声：

在实际应用中，图像可能会受到噪声的影响，因此通过增加噪声的方法可以增加数据集的多样性，从而提高模型的鲁棒性。可以添加高斯噪声、椒盐噪声、条纹噪声等不同类型的噪声。

#### 工业场景的图像数据增强

在工业场景下，由于生产环境的特殊性，可以采用一些特有的数据增强策略来提升缺陷检测模型的性能。以下是一些可能在工业场景中常见的特有数据增强策略：

图像旋转和翻转：由于工业生产中物体的朝向和位置可能存在变化，可以通过对图像进行旋转和翻转变换来模拟实际生产中的朝向和位置变化。例如，对图像进行随机角度的旋转、水平或垂直翻转等变换，从而增加模型对不同朝向和位置的缺陷的识别能力。

光照变换：在工业生产现场，光照条件可能会受到环境、设备和材料等因素的影响，因此可以通过光照变换来模拟实际生产中的光照变化情况。例如，对图像进行亮度、对比度、曝光度等光照变换，从而提高模型对不同光照条件下的缺陷的检测能力。

噪声添加：在实际生产中，图像可能受到噪声的干扰，例如传感器噪声、电磁干扰等，可以通过添加噪声来模拟实际生产中的噪声情况。例如，向图像中添加高斯噪声、椒盐噪声等，从而提高模型对噪声环境下的缺陷的检测鲁棒性。

遮挡变换：在实际生产中，由于物体之间可能存在遮挡情况，可以通过遮挡变换来模拟实际生产中的遮挡情况。例如，向图像中添加随机形状、位置和尺寸的遮挡物体，从而增加模型对遮挡缺陷的检测能力。

数据平衡处理：在实际生产中，不同类型的缺陷可能存在样本不平衡的情况，可以通过数据平衡处理来解决这一问题。例如，通过欠采样、过采样等方法来平衡不同类别的缺陷样本数量，从而提高模型对不同类型缺陷的识别能力。

这些特有的数据增强策略可以根据具体的工业生产场景和缺陷检测任务的需求进行选择和调整，从而提升模型在工业场景下的缺陷检测性能。

#### 误区

虽然数据增强方法可以有效提高模型的性能和鲁棒性，但是在某些特定情况下，一些增强方法可能会产生负面效果，甚至会导致模型性能下降。以下是一些在特定情况下不适用的增强方法：

随机裁剪

当图像中包含重要的物体或信息时，随机裁剪可能会导致物体或信息的丢失或变形，从而影响模型的性能。在这种情况下，应该采用其他的数据增强方法，如缩放或平移等。

随机旋转

当图像中包含文本或水平线时，随机旋转可能会导致文本或线变形，从而影响模型的性能。在这种情况下，应该限制旋转的角度，或采用其他的数据增强方法。

随机缩放

当图像中包含物体或特征时，过度缩放可能会导致物体或特征变形或丢失，从而影响模型的性能。在这种情况下，应该限制缩放的比例，或采用其他的数据增强方法。

随机颜色变换

当图像中包含特定颜色的物体或信息时，随机颜色变换可能会导致物体或信息失真或变形，从而影响模型的性能。在这种情况下，应该限制颜色变换的幅度，或采用其他的数据增强方法。

添加噪声

当图像中包含细节或纹理时，添加噪声可能会导致细节或纹理变得不清晰或失真，从而影响模型的性能。在这种情况下，应该限制噪声的强度，或采用其他的数据增强方法。

工业场景下特有的方法

在工业场景下，由于生产环境的特殊性，可以采用一些特有的数据增强策略来提升缺陷检测模型的性能。以下是一些可能在工业场景中常见的特有数据增强策略：

图像旋转和翻转：由于工业生产中物体的朝向和位置可能存在变化，可以通过对图像进行旋转和翻转变换来模拟实际生产中的朝向和位置变化。例如，对图像进行随机角度的旋转、水平或垂直翻转等变换，从而增加模型对不同朝向和位置的缺陷的识别能力。

光照变换：在工业生产现场，光照条件可能会受到环境、设备和材料等因素的影响，因此可以通过光照变换来模拟实际生产中的光照变化情况。例如，对图像进行亮度、对比度、曝光度等光照变换，从而提高模型对不同光照条件下的缺陷的检测能力。

噪声添加：在实际生产中，图像可能受到噪声的干扰，例如传感器噪声、电磁干扰等，可以通过添加噪声来模拟实际生产中的噪声情况。例如，向图像中添加高斯噪声、椒盐噪声等，从而提高模型对噪声环境下的缺陷的检测鲁棒性。

遮挡变换：在实际生产中，由于物体之间可能存在遮挡情况，可以通过遮挡变换来模拟实际生产中的遮挡情况。例如，向图像中添加随机形状、位置和尺寸的遮挡物体，从而增加模型对遮挡缺陷的检测能力。

数据平衡处理：在实际生产中，不同类型的缺陷可能存在样本不平衡的情况，可以通过数据平衡处理来解决这一问题。例如，通过欠采样、过采样等方法来平衡不同类别的缺陷样本数量，从而提高模型对不同类型缺陷的识别能力。

这些特有的数据增强策略可以根据具体的工业生产场景和缺陷检测任务的需求进行选择和调整，从而提升模型在工业场景下的缺陷检测性能。

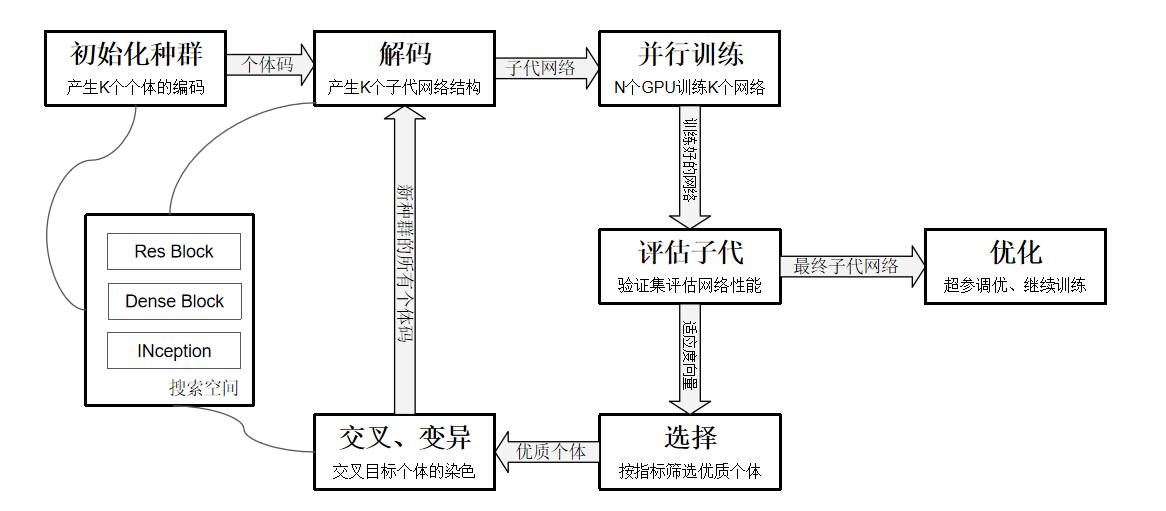
降噪

缺失值、清洗

数据集与label的并集

数据集扩充

# 所提出的方法



## GA

## NAS

输入：种群大小$N$，最大迭代次数$T$，交叉概率$\mu$，变异概率$\nu$。

输出：最佳的卷积神经网络（CNN）。

1. 采用提议的编码策略，使用大小为$N$的种群$P\_0$进行初始化。
2. 评估$P\_0$中个体的适应度。
3. 设$t=0$。
4. 当$t<T$时，进行以下操作：
   * 令$Q\_t=\emptyset$。
   * 当$|Q\_t|<N$时，进行以下操作：
     + 从$P\_t$中使用二元锦标赛选择方法选择两个亲本个体$p\_1$和$p\_2$。
     + 通过交叉概率$\mu$和变异概率$\nu$进行交叉和变异，生成两个后代个体$q\_1$和$q\_2$。
     + 将$q\_1$和$q\_2$添加到集合$Q\_t$中。
   * 对集合$Q\_t$中的个体进行适应度评估。
   * 使用环境选择方法从$P\_t\cup Q\_t$中选择$N$个个体，得到种群$P\_{t+1}$。
   * 令$t=t+1$。
5. 从$P\_T$中选择适应度最好的个体，并将其解码为对应的CNN。

传统神经网络设计流程和神经网络自动搜索（NAS）网络设计流程有一些显著的不同之处。以下是两者之间的对比：

传统神经网络设计流程：

人工设计网络架构：传统神经网络设计流程通常由人工设计网络架构开始。这通常需要人工选择网络的层数、每层的神经元数量、激活函数、损失函数等超参数，并进行多次试验和调整以找到最佳的网络架构。

训练网络：设计好的网络架构被用于训练数据，通过前向传播和反向传播来进行权重更新，以最小化损失函数。这个过程需要多次迭代，直到网络收敛到一个合适的状态。

超参数调优：在训练网络的过程中，还需要调整一些超参数，如学习率、批处理大小等，以优化网络的性能。

验证和测试：在训练网络后，需要使用验证集进行模型验证，以选择最佳的网络架构和超参数。最后，使用测试集来评估最终模型的性能。

NAS网络设计流程：

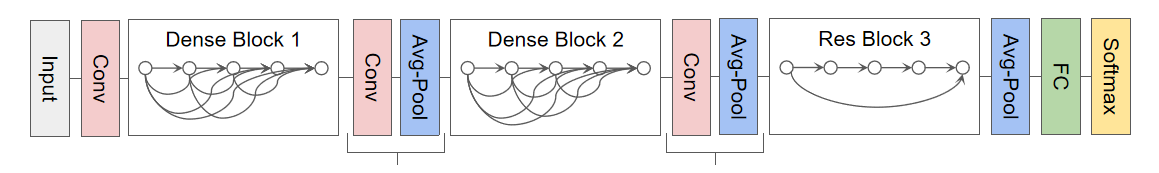
网络搜索空间定义：NAS网络设计流程首先需要定义一个网络搜索空间，包括网络架构的所有可能选择，例如不同层数、不同类型的层（如卷积层、池化层、全连接层等）、每层的参数设置等。

自动搜索：在NAS网络设计流程中，使用自动搜索算法，例如遗传算法、强化学习等，来自动地搜索最佳的网络架构。这通常通过在搜索空间中进行探索和评估不同的网络架构，并根据性能评估结果来生成新的网络架构。这个过程通常需要大量的计算资源和时间。

模型评估和选择：在自动搜索过程中，会根据性能评估指标对生成的网络架构进行评估，并选择性能最佳的网络架构作为最终模型。

模型训练和验证：在选择了最佳网络架构后，需要使用训练数据对其进行训练，并使用验证集进行模型验证和超参数调优，以找到最佳的超参数设置。

测试和部署：在训练和验证后，需要使用测试集对最终模型进行评估，以获得最终的性能指标。然后，可以将经过搜索和优化的最佳网络架构部署到实际应用中。

图x:dense-block和res-block的搜索

### 确定搜索空间

确定搜索空间是神经架构搜索（NAS）的一个重要环节。搜索空间的大小和选择对搜索结果和效率都有很大的影响。搜索空间需要包含可行的、合理的、创新的网络结构。一般而言，确定搜索空间的方法可以从以下方面入手：网络模块的设计：设计一些常用的网络模块，如卷积、池化、残差模块等，再将这些模块进行组合。

操作的种类和方式：选择网络操作的种类，如卷积、全连接等，以及操作的方式，如操作的步长、空洞卷积等。

连接方式：确定连接方式，如链式连接、分支连接等。

网络深度和宽度：决定网络的深度和宽度范围，如最小和最大的层数、最小和最大的通道数。

需要注意的是，在设计搜索空间时应避免过于复杂，过于庞大的空间会导致搜索难度增大，而过于简单的空间则可能导致搜索结果的欠佳。因此，在确定搜索空间时应该综合考虑搜索空间的大小、结构的合理性、可扩展性等多方面因素。

### 搜索算法

遗传算法（Genetic Algorithm，GA）：GA是一种基于进化论的优化算法，通过模拟生物进化过程来寻找最优解。在神经架构搜索中，GA通过交叉、变异等操作来产生新的网络结构，评估其性能并筛选出优秀的网络。

强化学习（Reinforcement Learning，RL）：RL是一种通过智能体与环境互动，获得奖励信号来学习最优策略的算法。在神经架构搜索中，RL可以将网络结构视为智能体，通过学习在不同结构下的性能奖励，来优化网络结构。

梯度下降（Gradient Descent，GD）：GD是一种常用的优化算法，通过沿着损失函数梯度方向更新网络参数，来逐步优化网络性能。在神经架构搜索中，GD可以通过对网络结构进行梯度下降优化来寻找最优网络。

贝叶斯优化（Bayesian Optimization，BO）：BO是一种基于贝叶斯理论的优化算法，通过构建目标函数的后验分布来寻找最优解。在神经架构搜索中，BO可以通过在搜索空间上构建高斯过程模型，来预测不同网络结构的性能，从而指导搜索过程。

### 训练和评估

在 NAS 中，训练和评估通常需要在多个步骤中进行。

首先，需要对每个候选架构进行训练，以评估其性能。训练可能需要在多个数据集上进行，以获得对候选架构性能的全面评估。通常，为了节省时间和计算资源，可以使用小型数据集（例如 CIFAR-10）来快速训练和评估候选架构，然后在更大的数据集上（例如 ImageNet）上对最终的候选架构进行训练和评估。

其次，需要对每个候选架构进行评估。评估可以使用各种指标，例如分类精度、检测精度或语义分割IoU等。需要评估不同的指标，以便找到最优的架构。

最后，需要使用最优的架构进行最终的训练和评估，以便获得最终的模型。在这一步骤中，可以使用更大的数据集和更长的训练时间来训练模型，并使用更多的评估指标来评估其性能。

## 代码系统设计

利用process库多进程运行自代

需要注意的是，当在多张GPU上并行运行时，需要注意每个子进程使用的GPU索引，避免多个子进程访问同一张GPU造成冲突。另外，还需要根据具体的深度学习模型和数据分布情况，合理划分数据和模型到不同的GPU上，以充分利用多张GPU的计算资源。同时，也需要考虑多进程间的数据通信和同步，确保计算结果的正确性。

首先，需要考虑GPU资源的分配和调度。在多GPU系统中，可以使用分布式训练的方法来将神经网络的计算任务分配到多个GPU上进行并行计算。分布式训练需要使用一些通信机制来同步不同GPU上的参数更新，例如AllReduce算法。为了保证系统的性能，需要对GPU资源进行合理的分配和调度，避免资源浪费和任务调度不平衡等问题。

其次，需要考虑数据的并行处理。在多GPU系统中，可以使用数据并行的方法将数据分割成多个小批次，并分别在不同的GPU上进行处理。数据并行需要使用一些数据通信机制来同步不同GPU上的计算结果。为了避免数据通信带来的性能瓶颈，可以采用异步通信或者局部同步的方法来减少通信开销。

此外，还需要考虑模型的拆分和重组。为了在多GPU上进行并行计算，可以将一个大型神经网络拆分成多个子模型，分别在不同GPU上进行计算，并在计算结果之间进行通信和合并。拆分和重组需要根据具体的神经网络结构和任务需求来进行设计，需要保证计算负载的均衡和计算结果的正确性。可以使用预训练模型或迁移学习来加速网络训练。在NAS中，可以使用预训练模型作为搜索空间中的初始模型，从而减少搜索时间和计算资源。

最后，还需要考虑系统的容错和可扩展性。在多GPU系统中，由于GPU硬件的故障和网络通信的不稳定性等原因，可能会出现计算任务失败和数据丢失等问题。为了保证系统的可靠性和可扩展性，需要采用一些容错机制和自适应调度算法来处理这些异常情况，并进行系统监控和诊断，及时发现和修复问题。

# 实验

实验环境配备2\*NVIDIA RTX309 GPU、Intel Core i7-9700 CPU、16GB内存、Windows 10操作系统和PyTorch深度学习框架。实验中调整图像大小为224×224像素，模型训练的mini-batch为10，整个训练为100个epochs，学习率设置为0.0001，采用Adam优化算法优化模型。

我们使用 X-SDD 中 70% 的数据作为训练集，使用 X-SDD 中 30% 的数据作为测试集。因此，训练集包含 952 张图像，而测试集包含 408 张图像。

对VGG16和ResNet50进行性能测试, 并采用在miniImageNet数据集上预训练(pre)和从头学习2种方式测试网络性能

学习率下降策略，找到最合适lr，应该从小到大

## 可视化分析

为了更好地对比分析特征嵌入模型能否有效表达各个类别的特征, 以及从不同领域学习的方法在NEU数据集上的表现差异, 在NEU数据集上使用t-分布随机近邻嵌入(t-Distribution Stochastic Neigh-bour Embedding, t-SEN)可视化6个类别、每个类别100个样本的特征嵌入, 如图7所示。

由图7可知, 2个不同的源域训练的方法都能较好地区分每个类别, 这说明FmTLNet可正确区分样本.从这2个可视化图对比中可看到明显差异, 在miniImage-Net数据集上学习的方法的区分度优于NEU数据集.

综合上述分析可知, 小样本度量学习通过在非相关的大型数据集上学习, 使网络能利用较少的样本(5幅或10幅图像)获得较优性能, 在缺陷检测过程中减少对缺陷样本量的需求, 提升工作效率, 满足快速应用的需求.然而, 从实验过程上看, 小样本度量学习对源域较敏感.源域复杂度、类别数及样本量的不同都可能造成性能的明显差异.不同特征嵌入模块的对比实验表明, 模型规模的大小影响性能。

### t-SNE

t-SNE (t-Distributed Stochastic Neighbor Embedding) 是一种非线性降维算法，可以将高维数据映射到低维空间（通常是二维或三维），并尽可能地保留原始数据之间的局部相似性关系。

下面是使用 t-SNE 算法将数据集降维并进行可视化的基本步骤：

准备数据集：准备一个数据集，该数据集应该包含至少两个特征（或变量）和相应的标签。这些特征可以是连续型变量、离散型变量或混合型变量。

对数据进行标准化：由于 t-SNE 是基于距离的算法，因此需要将数据进行标准化，以确保每个特征的尺度相等。一种常见的方法是将每个特征缩放到 [0, 1] 的范围内。

选择 t-SNE 的参数：t-SNE 有两个主要参数需要调整：学习率（learning rate）和困惑度（perplexity）。学习率控制每个点移动的速度，而困惑度则控制了每个点周围所考虑的相邻点的数量。这两个参数的值可以通过试验来确定。

运行 t-SNE 算法：使用所选参数运行 t-SNE 算法，将数据集从高维映射到低维空间。

可视化结果：使用可视化工具（如 Matplotlib、Seaborn 或 Plotly）将 t-SNE 映射的结果可视化。在二维空间中，每个点表示数据集中的一个观测值，点的颜色或形状可以表示观测值的标签。

## 指标

Accuary (准确率):

Accuary = (TP + TN) / (TP + TN + FP + FN)

Macro-recall (宏平均召回率):

Macro-recall = Σ\_{i=1}^{n} Recall\_i / n

其中，Recall\_i 是第 i 类的召回率，n 是类别的总数。

Macro-precision (宏平均精确率):

Macro-precision = Σ\_{i=1}^{n} Precision\_i / n

其中，Precision\_i 是第 i 类的精确率，n 是类别的总数。

Macro-F1 (宏平均F1分数):

Macro-F1 = 2 \* Macro-precision \* Macro-recall / (Macro-precision + Macro-recall)

Macro - Recall = \frac{{TP0}}{{TP0 + FN0}} + \frac{{TP1}}{{TP1 + FN1}} + \ldots + \frac{{TPN-1}}{{TPN-1 + FNN-1}}} \times \frac{1}{N}

参考文献目录：

1. Kechen Song, Yunhui Yan. A noise robust method based on completed local binary patterns for hot-rolled steel strip surface defects, Applied Surface Science, Volume 285, Part B, 2013, Pages 858-864.

2. Y. He, K. Song, Q. Meng and Y. Yan. An End-to-End Steel Surface Defect Detection Approach via Fusing Multiple Hierarchical Features. IEEE Transactions on Instrumentation and Measurement, vol. 69, no. 4, pp. 1493-1504, April 2020.

3. Feng X, Gao X, Luo L. X-SDD: a new benchmark for hot rolled steel strip surface defects detection[J]. Symmetry, 2021, 13(4): 706.

4. Y. Bao et al. Triplet-Graph Reasoning Network for Few-Shot Metal Generic Surface Defect Segmentation. in IEEE Transactions on Instrumentation and Measurement, vol. 70, pp. 1-11, 2021, Art no. 5011111.

5.Song G, Song K, Yan Y. Saliency detection for strip steel surface defects using multiple constraints and improved texture features[J]. Optics and Lasers in Engineering, 2020, 128: 106000.

6. Alexey Grishin, BorisV, iBardintsev, inversion, Oleg. Severstal: Steel Defect Detection. Kaggle, 2019, <https://kaggle.com/competitions/severstal-steel-defect-detection>.

7. Chiahaol, IceLotus, Tsai Yi Lin, Yuchi Liu. AIA Manufacturing Project - Defect inspection, Kaggle, 2022, <https://kaggle.com/competitions/manufacturing-project-defect-inspection>.

8. Tabernik, Domen, et al. Segmentation-Based Deep-Learning Approach for Surface-Defect Detection. Journal of Intelligent Manufacturing, vol. 31, no. 3, May 2019, pp. 759–76.

9. Božič J, Tabernik D, Skočaj D. Mixed supervision for surface-defect detection: From weakly to fully supervised learning[J]. Computers in Industry, 2021, 129: 103459.

10. Mishra P, Verk R, Fornasier D, et al. VT-ADL: A vision transformer network for image anomaly detection and localization[C]//2021 IEEE 30th International Symposium on Industrial Electronics (ISIE). IEEE, 2021: 01-06.

11. 29th Annual Symposium of the German Association for Pattern Recognition. Online Reference Documentation [EB/OL]. <https://conferences.mpi-inf.mpg.de/dagm/2007/prizes.html>.

12. Feng X, Gao X, Luo L. X-SDD: a new benchmark for hot rolled steel strip surface defects detection[J]. Symmetry, 2021, 13(4): 706.

13. Yang L, Huang X, Ren Y, et al. Steel Plate Surface Defect Detection Based on Dataset Enhancement and Lightweight Convolution Neural Network[J]. Machines, 2022, 10(7): 523.

14. He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]. Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

15. Ge Z, Liu S, Wang F, et al. Yolox: Exceeding yolo series in 2021[J]. arXiv preprint arXiv:2107.08430, 2021.

16. Chen L C, Zhu Y, Papandreou G, et al. Encoder-decoder with atrous separable convolution for semantic image segmentation[C]. Proceedings of the European conference on computer vision (ECCV). 2018: 801-818.

17. Zoph B, Le Q V. Neural architecture search with reinforcement learning[J]. arXiv preprint arXiv:1611.01578, 2016.

18. Liashchynskyi P, Liashchynskyi P. Grid search, random search, genetic algorithm: a big comparison for NAS[J]. arXiv preprint arXiv:1912.06059, 2019.

19.张浩宇. 基于进化计算的深度神经网络架构搜索算法研究[D].东华大学,2022.DOI:10.27012/d.cnki.gdhuu.2022.000020.

20. Holland J H. Genetic algorithms[J]. Scientific american, 1992, 267(1): 66-73.

21. Kennedy J, Eberhart R. Particle swarm optimization[C]//Proceedings of ICNN'95-international conference on neural networks. IEEE, 1995, 4: 1942-1948.

22. O’Neill M. Riccardo Poli, William B. Langdon, Nicholas F. McPhee: a field guide to genetic programming[J]. 2009.

23. Sun Y, Yen G G, Yi Z. IGD indicator-based evolutionary algorithm for many-objective optimization problems[J]. IEEE Transactions on Evolutionary Computation, 2018, 23(2): 173-187.

24. Darwish A, Hassanien A E, Das S. A survey of swarm and evolutionary computing approaches for deep learning[J]. Artificial intelligence review, 2020, 53(3): 1767-1812.

25. Karp, R.. Reducibility among combinatorial problems. In R. Miller & J. Thatcher (ed.), Complexity of Computer Computations , 1972, pp. 85-103.

26. Floreano D, Dürr P, Mattiussi C. Neuroevolution: from architectures to learning[J]. Evolutionary intelligence, 2008, 1(1): 47-62.

27. Real E, Moore S, Selle A, et al. Large-scale evolution of image classifiers[C]//International Conference on Machine Learning. PMLR, 2017: 2902-2911.

28. Darwish A, Hassanien A E, Das S. A survey of swarm and evolutionary computing approaches for deep learning[J]. Artificial intelligence review, 2020, 53(3): 1767-1812.

29. Stanley K O, Clune J, Lehman J, et al. Designing neural networks through neuroevolution[J]. Nature Machine Intelligence, 2019, 1(1): 24-35.

30. Huang G, Liu Z, Van Der Maaten L, et al. Densely connected convolutional networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 4700-4708.

31.Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 1-9.

32. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.

33. Sun Y, Xue B, Zhang M, et al. Evolving deep convolutional neural networks for image classification[J]. IEEE Transactions on Evolutionary Computation, 2019, 24(2): 394-407.

34. Fujino S, Mori N, Matsumoto K. Deep convolutional networks for human sketches by means of the evolutionary deep learning[C]//2017 Joint 17th World Congress of International Fuzzy Systems Association and 9th International Conference on Soft Computing and Intelligent Systems (IFSA-SCIS). IEEE, 2017: 1-5.

35. Xie L, Yuille A. Genetic cnn[C]//Proceedings of the IEEE international conference on computer vision. 2017: 1379-1388.

36. Sun Y, Xue B, Zhang M, et al. Evolving deep convolutional neural networks for image classification[J]. IEEE Transactions on Evolutionary Computation, 2019, 24(2): 394-407.

37. Liu S, Zhang H, Jin Y. A survey on computationally efficient neural architecture search[J]. Journal of Automation and Intelligence, 2022, 1(1): 100002.

38.王立中,管声启.基于深度学习算法的带钢表面缺陷识别[J].西安工程大学学报,2017,31(05):669-674.

39.梁日强,胡燕林,蒋占四.基于改进的残差收缩网络的带钢表面缺陷识别[J].组合机床与自动化加工技术,2022(06):82-85.

40.李维刚,徐康,李金灵,赵云涛.热轧带钢表面缺陷识别算法研究与应用[J].钢铁,2022,57(10):139-147.

41.翟丹娜. 基于图像处理的带钢表面缺陷识别方法研究[D].河北工业大学,2015.

42.陈小改.基于PCA-SVM的带钢表面缺陷识别[J].机械设计与制造工程,2017,46(06):100-103.

43. Challenge I N L S V R. Olga russakovsky, jia deng, hao su, jonathan krause, sanjeev satheesh, sean ma, zhiheng huang, andrej karpathy, aditya khosla, michael bernstein, alexander c. berg, li fei-fei. 2014[J]. Computing Research Repository, Vol. abs/1409.0575.

44. Krizhevsky A, Hinton G. Learning multiple layers of features from tiny images[J]. 2009.

45. Gibb S, La H M, Louis S. A genetic algorithm for convolutional network structure optimization for concrete crack detection[C]//2018 IEEE Congress on Evolutionary Computation (CEC). IEEE, 2018: 1-8.

46. Zhou B, Khosla A, Lapedriza A, et al. Learning deep features for discriminative localization[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 2921-2929.

47. 单东日,童灿,乃学尚,高立营,王玉伟.基于小波和灰度共生矩阵的带钢表面缺陷识别[J].制造技术与机床,2020(02):120-123.DOI:10.19287/j.cnki.1005-2402.2020.02.025.

48. 孙梓钧. 基于AdaBoost分类器的带钢表面缺陷识别研究[D].河北科技大学,2016.

49. 陈小改.基于PCA-SVM的带钢表面缺陷识别[J].机械设计与制造工程,2017,46(06):100-103.

50. 杨水山,何永辉,赵万生.Boosting优化决策树的带钢表面缺陷识别技术[J].红外与激光工程,2010,39(05):954-958.

51. 陈永胜. 基于机器视觉的带钢表面缺陷检测系统的研究[D].武汉科技大学,2010.

52. Zhang Z, Yu S, Yang S, et al. Rail-5k: A real-world dataset for rail surface defects detection[J]. arXiv preprint arXiv:2106.14366, 2021.

53. Niu M, Song K, Huang L, et al. Unsupervised saliency detection of rail surface defects using stereoscopic images[J]. IEEE Transactions on Industrial Informatics, 2020, 17(3): 2271-2281.