

2022 年全国大学生电子设计竞赛 信息科技前沿专题邀请赛（瑞萨杯） 作品设计报告

作品名称： 基于嵌入式深度学习的小物件表面缺陷检测

参赛学校： 上海理工大学

参赛学生： 葛弘毅、陆子康、简畅

电子邮箱： HongyiGE@outlook.com

提交日期： 2022.8.25

填写说明

1. 所有参赛项目必须为一个基本完整的设计。作品报告旨在能够清晰准确地阐述（或图示）该参赛队的参赛项目（或方案）。
2. 作品报告采用A4纸撰写。除标题外，所有内容一般为宋体、五号字、单倍行距。
3. 作品报告中各项目的说明文字部分仅供参考，作品报告撰写完毕后，请删除所有说明文字。(本页不删除)
4. 作品报告模板里所列内容仅供参考，作者可以在此基础上增加内容或对文档结构进行微调。
5. 作品报告从目录页开始，不得出现任何校名、参赛队代号、参赛队员姓名及其它暗记（否则视为无效）。

原创性声明

本团队谨声明本作品报告所阐述的参赛作品是本团队在指导教师的指导下，独立进行研究所取得的真实成果，从创意到实现均为原创。

除作品报告中已注明引用的内容外，参赛作品中不含任何其他个人或集体已经发表或撰写过的作品成果。本团队对该作品拥有完整、合法的著作权及其他相关权益。

本团队及作品严格遵守全国大学生电子设计竞赛组织委员会和信息科技前沿专题邀请赛组委会颁布的相关规定，并且无侵害他人合法权益行为和违反相关法律法规行为。

目 录

原创性声明	③
摘 要	1
1. 作品概述	3
1.1 背景分析	3
1.2 相关工作	3
1.3 特色设计	3
1.4 应用前景	4
2. 作品设计与实现	4
2.1 系统方案	4
2.2 硬件结构和软件体系	5
2.2.1 硬件结构	5
2.2.2 软件体系	7
2.3 待测物件的图像捕获与剪裁	8
2.3.1 实现原理	8
2.3.2 设计流程	8
2.4 基于嵌入式深度学习的缺陷检测	10
2.4.1 模型原理	10
2.4.2 设计计算	11
2.5 物件缺陷情况的可视化网页展示	14
2.5.1PC 端访问瑞萨 RZ/G2L 开发平台的文件	14
2.5.2Web 开发准备	15
2.5.3 网页的页面结构	15
2.5.4 网页的功能设计	17
2.6 功能与指标	19
2.6.1 功能	19
2.6.2 指标	19
3. 作品测试与分析	20
3.1 测试设备和环境搭建	20
3.2 测试方案	20
3.2.1 系统性能测试方案	20
3.2.2 系统功能测试方案	21
3.3 测试结果分析	21
3.3.1 系统性能测试结果分析	21
3.3.2 系统功能测试结果分析	23
4. 创新性说明	27
5. 总结	27
5.1 作品立意	27
5.2 优点与缺点	28
5.2.1 优点	28
5.2.2 缺点	28

摘要

本作品围绕大赛网络化语音图像检测与识别的主题，紧扣“网络化”和“图像检测与识别”两个关键词，立足于网络共享、机器视觉、嵌入式开发、深度学习等复杂工程背景，实现小物件产品的智能化表面缺陷检测。

作品的核心功能是对工业生产过程中的小物件进行表面缺陷检测，整体设计分为3个功能部分：待测物件的图像捕获与剪裁、基于嵌入式深度学习的缺陷检测、缺陷检测结果的可视化网页展示。

在第1部分中，操作 PyQt5图形交互页面，使连接到瑞萨 RZ/G2L 开发平台的摄像头实时获取画面，经过裁剪、缩放后输出待测物件图像，将待测图像保存至开发平台的指定文件夹；在第2部分中，首先利用 Jetson Orin，针对 MVTec 数据集的子集，训练和测试知识蒸馏模型，随后将 Jetson Orin 训练测试好的模型移植到瑞萨 RZ/G2L 开发平台，从瑞萨 RZ/G2L 开发平台指定文件夹抓取待测图像，输入移植到该开发平台的模型进行测试，最终输出特征金字塔多尺度特征融合前后的系列异常热力图以及综合热力实物合成图，并保存至开发平台的指定文件夹；在第3部分中，我们使用 HTML、CSS、JavaScript 设计开发了“质检反馈站”可视化网页，并使 PC 端与瑞萨 RZ/G2L 开发平台局域网互连，在 PC 端通过 samba 服务访问瑞萨 RZ/G2L 开发平台的指定文件夹，抓取缺陷检测的结果输出图像等，将缺陷检测结果展示在“质检反馈站”首页。

作品亮点在于数据的网络共享，嵌入式深度学习，算法的多尺度特征融合，“质检反馈站”网页的配套开发等，提高了检测任务的精度，解放人力并降低成本，正反馈调节前端生产，有较大的实用和推广价值。

关键词： 网络化 图像检测与识别 网络共享 机器视觉 嵌入式开发 深度学习 缺陷检测 知识蒸馏 特征金字塔 可视化网页

Surface Defect Detection of Small Objects Based on Embedded Deep Learning

Abstract:

This work focuses on the theme of networked speech image detection and recognition in the competition, closely following the two keywords of "network" and "image detection and recognition", and based on the complex engineering background of network sharing, machine vision, embedded development, deep learning and so on, realizing intelligent surface defect detection of small objects.

The core function of the work is to detect surface defects of small objects in the process of industrial production. The overall design is divided into three functional parts: image capture and clipping of objects to be tested, defect detection based on embedded deep learning, and visual web display of defect detection results.

In the first part, the PyQt5 graphical interaction page is operated to enable the camera connected to Renesas RZ/G2L development platform to obtain images in real time. After clipping and scaling, the image of the object to be tested is output, and the image to be tested is saved to the specified folder of the development platform. In the second part, Jetson Orin is first used to train and test the knowledge distillation model for a subset of the MVTec dataset. Then the model trained and tested by Jetson Orin is transplanted to Renesas RZ/G2L development platform, and the images to be tested are captured from the specified folder of Renesas RZ/G2L development platform. Input the model transplanted to the development platform for testing, and finally output the series of abnormal thermal maps before and after the multi-scale feature fusion of the feature pyramid and the comprehensive thermal physical synthesis map, and save to the designated folder of the development platform; In Part 3, we design and develop the visual web page of "Quality Control Feedback Station" using HTML, CSS and JavaScript, and connect the PC side with the LAN of Renesas RZ/G2L development platform, and access the designated folder of Renesas RZ/G2L development platform through Samba service on the PC side. Capture the output images of defect detection results, etc., and display the defect detection results on the home page of "quality control feedback station".

The highlights of the work are the network sharing of data, embedded deep learning, multi-scale feature fusion of the algorithm, and the supporting development of the web page of "quality inspection feedback station", etc., which improves the accuracy of the testing task, saves manpower and reduces costs, and adjusts front-end production with positive feedback, which has great practical and promotional value.

Keywords: network, image detection and recognition, network sharing, machine vision, embedded development, deep learning, defect detection, knowledge distillation, feature pyramid, visualized web pages

1. 作品概述

1.1 背景分析

本作品的设计构想源于工业生产中，机器视觉的技术背景下，零件等小物件产品的表面缺陷检测。

首先，缺陷检测是各行业产品质量管理体系中的重要一环，也是产品在正式投入市场应用前最后一道屏障，严格的质检把关在工业生产中极为重要。

其次，小物件产品的缺陷较小，很多情况下不能直接用肉眼判别检测；肉眼检测耗费时间和人力，完全取决于人工的检测也导致稳定性和一致性上下浮动，达不到预期效果；加之工业流水线上的产品缺陷检测任务大种类多，人力培训成本高，用工劳动强度大，寻求机器视觉的帮助已然成为一种趋势。

因此，智能化的小物件产品缺陷检测在工业生产中是有必要的，我们就基于网络化、机器视觉、嵌入式开发、深度学习等背景，设计实现了具备上述功能的作品。

1.2 相关工作

嵌入式开发在本作品的主要应用体现在嵌入式深度学习。随着技术发展，性能加速器能力的指数级增长使深度学习模型可以由远离云端的边缘嵌入式系统运行，带来低延迟、低能耗、高带宽的优势。与此同时，在嵌入式系统运行的深度学习模型必须考虑运行速度及能效比，因此模型的计算复杂度和尺寸都是越小越好。

深度学习领域的缺陷检测方法可以很大程度上克服人工检测的弊端，近年来得到越来越广泛的研究和应用。缺陷图像检测目前主要存在由这三种思想主导的技术：基于重构、基于分布、基于分类。基于重构：重构训练图像以获取正常数据流形，根据重构误差的大小来判定异常图像；基于分布：对图像的概率分布进行建模，根据概率密度值的大小来判定异常图像；基于分类：捕获图像数据特征，根据特征对应的类概率进行分类来判定异常图像，是应用最广泛的思想。

1.3 特色设计

待测物件的图像捕获与剪裁：使连接到瑞萨 RZ/G2L 开发平台的摄像头实时获取画面，经过裁剪、缩放后输出待测物件图像，将待测图像保存至瑞萨 RZ/G2L 开发平台的指定文件夹。

基于嵌入式深度学习的缺陷检测：首先利用 Jetson Orin，针对 MVTec 数据集的子集，训练和测试知识蒸馏^[1]模型，随后将 Jetson Orin 训练测试好的模型移植到瑞萨 RZ/G2L 开发平台。从瑞萨 RZ/G2L 开发平台的指定文件夹抓取待测图像，输入移植到瑞萨 RZ/G2L 开发平台的模型进行缺陷检测，最终输出特征金字塔多尺度特征融合前后的系列异常热力图^[2]以及综合热力实物合成图，并保存至瑞萨 RZ/G2L 开发平台指定文件夹。

缺陷检测结果的可视化网页展示：我们设计开发了“质检反馈站”可视化网页，并使 PC 端与瑞萨 RZ/G2L 开发平台局域网互连，在 PC 端通过 samba 服务访问瑞萨 RZ/G2L 开发平台的指定文件夹，抓取缺陷检测的结果输出图像等，点击“生成报告”后，将缺陷检测结果展示在“质检反馈站”首页。

1.4 应用前景

首先，在工业生产中，本作品能够发挥智能化检测零件等产品缺陷的功能，检测并淘汰含缺陷的次品，从而提升对产品质量的可控性，使质检把关这一工业流程更加严格。

同时，本作品对缺陷产品进行信息的集成与留存，方便溯源，从而正反馈产品生产。缺陷检测在某种意义上有事后补救的效果，传统的人工检测对质检结果的记录常常不全面且分散，智能化缺陷检测统计并记录的产品缺陷信息，才能对产品生产进行反馈，促进精益化生产。

再者，由于人工检测的培训成本高，劳动强度大，岗位招工难，本作品的智能化功能属性在一定程度上也能起到解放人力，降低成本的效应。

2. 作品设计与实现

2.1 系统方案

本作品的整体设计按功能分割板块，可分为以下3个板块：待测物件的图像捕获与剪裁、基于嵌入式深度学习的缺陷检测、缺陷检测结果的可视化网页展示。

在待测物件的图像捕获与剪裁中（以下称图像捕获与剪裁），将罗技 C270或 C270i 摄像头通过 USB 接口连接到瑞萨 RZ/G2L 开发平台，操作 PyQt5图形交互页面，使摄像头实时捕获画面，并将捕获画面进行裁剪、等比例缩放至256px×256px 尺寸，输出作为待测物件图像，将待测图像保存至瑞萨 RZ/G2L 开发平台的指定文件夹。

在基于嵌入式深度学习的缺陷检测中（以下称缺陷检测），采纳深度学习中的知识蒸馏模型，以一个预训练的 ResNet-152网络作为教师模型，以一个初始化的同构 ResNet-152网络作为学生模型。利用 Jetson Orin 在 MVTec 数据集中该待测图像所属类别的数据子集上，对学生-教师模型进行训练和测试，学生模型通过与预训练的教师模型匹配特征金字塔层次特征^[3]来学习无异常图像的分布，不能抽取异常特征，但教师模型依然可以高质量抽取异常样本特征，两者的差异促成了对缺陷的检测。随后将 Jetson Orin 中训练测试后的模型嵌入移植到瑞萨 RZ/G2L 开发平台，再从瑞萨 RZ/G2L 开发平台指定文件夹抓有待测图像，输入移植到瑞萨 RZ/G2L 开发平台的模型，进行测试，最终输出特征金字塔多尺度特征融合前后的系列异常热力图以及综合热力实物合成图，并保存至瑞萨 RZ/G2L 开发平台指定文件夹。

在缺陷检测结果的可视化网页展示（以下称 Web 质检展示）中，我们设计开发了“质检反馈站”可视化网页，并使 PC 端与瑞萨 RZ/G2L 开发平台局域网互连，在 PC 端通过 samba 服务访问瑞萨 RZ/G2L 开发平台的指定文件夹，抓有待测图像、系列异常热力图以及综合热力实物合成图，点击“生成报告”后，将缺陷检测结果展示在“质检反馈站”首页。与此同时，我们集成收录了以往的12个缺陷检测案例，更直观和完整地展示了本作品对于小物件产品的缺陷检测性能。

总结下来，本作品系统设计框图如图1所示，同时，软硬件资源任务表如表1所示。

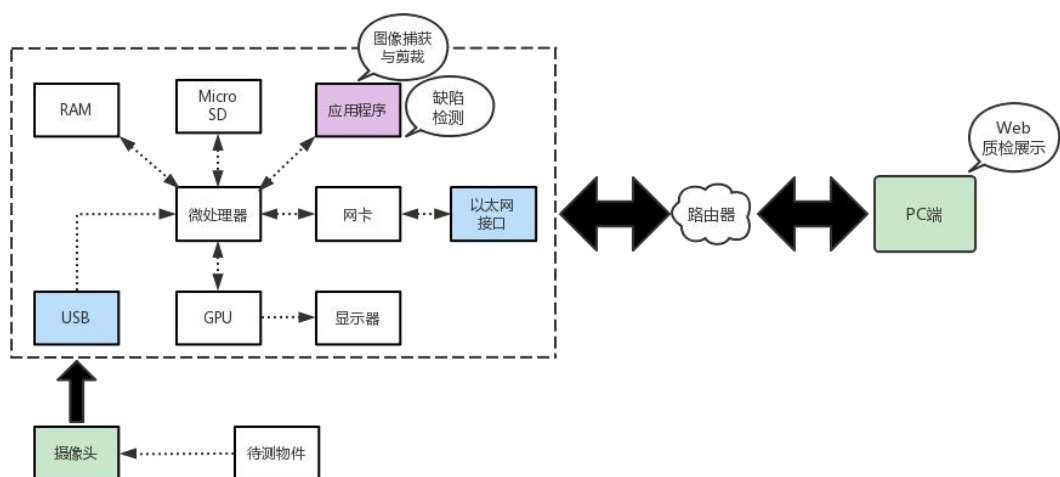


图 1系统设计框图

表 1 软硬件资源任务表

任务名称	详细功能
camera_task	捕获实时画面
cutting_task	对捕获图片进行裁剪缩放，获得待测物件图像
sharing_task	将图像等文件保存至指定文件夹
data_task	加载相应数据集
preparation_task	预训练教师模型，初始化学生模型
train-test_task	在相应数据集训练和测试学生-教师模型
test_task	对待测物件图像进行表面缺陷检测
anomaly_task	生成系列异常热力图以及综合热力实物合成图
visit_task	使用 samba 服务访问瑞萨 RZ/G2L 开发平台的指定文件夹
web_task	网页抓取测试结果，进行完整展示
screen_task	显示器抓取测试结果，展示原图像、缺陷热力图与分割图

2.2 硬件结构和软件体系

2.2.1 硬件结构

如图 2 所示，我们使用大赛官方提供的瑞萨 RZ/G2L 开发平台，并在已有硬件资源的基础上，做了一些选择和改良。

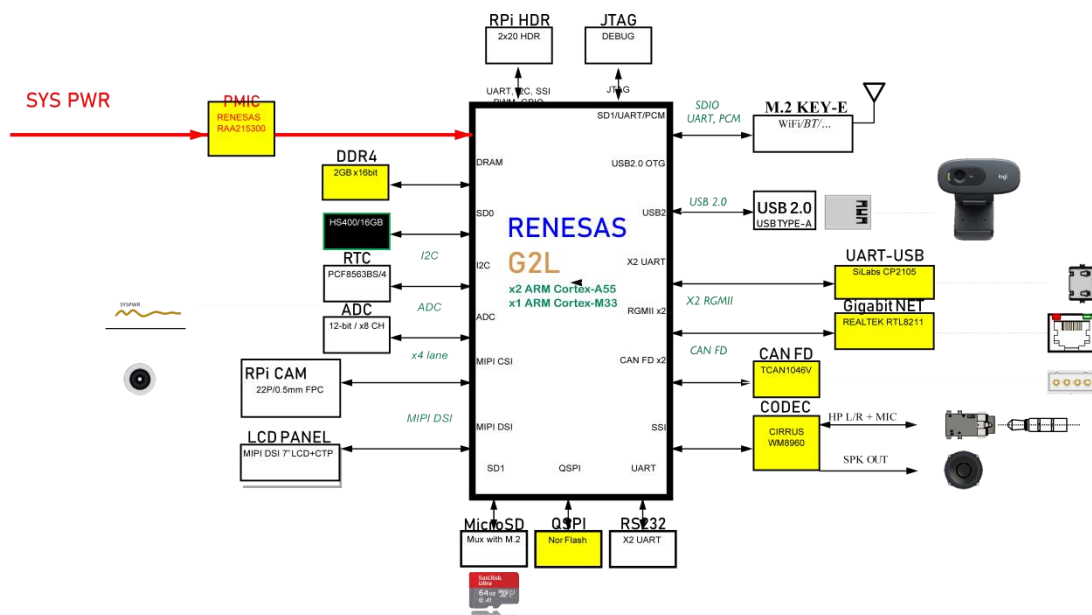


图 2 硬件结构改良图

我们利用了双核的 ARM Cortex 板载 2GB 800MHz 内存，TF 读卡器，MIPI-DSI，以太网和 USB 接口。

1.关于 TF 卡的设置及理由。

由于本作品的完成对内存消耗较大，需要使用交换空间（SWAP），且需要依赖较多的库文件；同时为降低系统兼容性带来的问题，我们采用与开发环境（Ubuntu 20.04）更为接近的 Ubuntu-base 20.04。由于系统体积可能较大，且使用 USB 将系统写入到内置的 EMMC 需要较长时间，我们采用将系统等文件写入到 TF 卡并通过 TF 卡启动的方式。

如图 3 所示，TF 卡通过硬盘精灵（DiskGenius）分为 IMAGE（500MB，FAT16，存放系统启动镜像），SWAP（8GB，交换空间），SHARE(9GB，FAT32，共享目录)，BASE(EXT4，系统分区)。



图 3 文件系统布局

板载内存 2GB，仅 Ubuntu（包括 Ubuntu-desktop）就需要 800MB，剩余空间无法满足缺陷检测程序的最小运行内存，因此使用交换空间辅助解决内存不足的问题。

由于 Windows 不支持直接读写 EXT4 格式的分区，故 SHARE 使用 FAT32 可方便 TF 卡插入计算机后直接复制文件。

2.通过 MIPI-DSI 连接 7 吋 LCD 触控屏。

3.USB 2.0 A 型接口连接 USB 扩展器，连接键盘和免驱动摄像头。

通过键盘的 trackpoint 或触控板控制鼠标，使用按键输入或完成快捷操作（亮度、锁屏、注销等）。

我们使用免驱动 USB 摄像头实时获取 1280px×720px 图像，并通过我们的图像获取程序裁剪至 256px×256px 并保存到指定文件夹内。

4.千兆以太网接口 1(eth0, 已设置动态 IP)与路由器使用跳线连接。

通过连接以太网，可以访问互联网自动更新系统、软件，还能同时同步时间。我们的项目使用 samba 服务可以不受 IP 变化的影响，精准地使计算机访问到指定的边缘设备即这里的瑞萨 RZ/G2L 开发平台。

2.2.2 软件体系

表 2 软件体系表

应用软件	应用程序		
系统软件	文件系统 FAT16, SWAP, FAT32, EXT4	嵌入式 GUI	Linux 任务管理
	嵌入式实时操作系统 Ubuntu-Base 20.04		
中间层	板级支持包/硬件抽象层（BSP/HAL）		
硬件层	定时器	微处理器 CPU	64GB Micro SD
	以太网接口		DDR4 800MHz 2GB
	USB		MIPI-DSI 接口显示器
外设	1 转 4USB 扩展器，USB 键盘，罗技 C270i		

本作品主要利用了双核 cortex A55，定时器，以太网接口，USB 2.0 Type-A， MIPI-DSI 接口触控显示器以及 DDR4 800MHz 2GB 内存。通过 USB2.0 和 USB 扩展器，连接 USB 键盘和罗技 C270i 摄像头。

启动时使用 gx-qsb-rzg2l.dtb 和 image.g2l 镜像文件引导。

系统内核使用 Ubuntu-base 20.04，gnome 作为图形化界面。

文件系统包括 FAT16 的镜像分区（存放 gx-qsb-rzg2l.dtb,image）。内存交换空间 SWAP（共 8GB）。为了便于将 TF 卡插入计算机后可以直接传输文件（Windows 系统下无法直接将文件读写到 EXT4 分区），设置 FAT32 格式的 share 共享分区。EXT4 存放系统文件。

本项目的应用程序分为两个。

程序 1：图像获取软件基于 PyQt5 编写，具有可视化图形界面，可以实时获取 1280px×720px 的图像，并在点击“截取图片”后保存实时画面中红色方框框出的正方形区域，并以 256px×256px 保存为 png 文件，输出为待测物件图像。

程序 2：基于 Python3.8 编写。通过读取保存的待测图片并加载与训练的权重文件，生成缺陷检测的热力图。

2.3 待测物件的图像捕获与剪裁

2.3.1 实现原理

操作 PyQt5 图形交互页面，通过连接到瑞萨 RZ/G2L 开发平台的罗技 C270 摄像头初步获取待测物件图像，经过裁剪、等比例缩放等一系列工作将图像调整为 256px×256px 尺寸，输出作为待测物件图像。

在 PyQt5 上显示捕获图片需要将 OpenCV 中捕获的 `mat` 图片类型转换成 `QImage` 图片类型。

$$bytesPerLine = channels \times cols$$

其中，RGB 彩色图像通道数 `Channels` 为 3，`rows` 为 Mat 矩阵的行数，`cols` 为 Mat 矩阵的列数。

可以通过 `QImage` 函数改变类型：

```
QImage(self.frame.data, cols, rows, bytesPerLine, QImage.Format_RGB888)
```

其中，`data` 为 `uchar` 类型的指针，指向 Mat 数据矩阵的首地址。

2.3.2 设计流程

1.硬件设备：瑞萨 RZ/G2L 开发平台、罗技 C270 摄像头、Rocktech Displays RK070CU05H-CTG 显示屏

2.硬件连接：Rocktech Displays RK070CU05H-CTG 显示屏通过 mipi DSI 接口与瑞萨 RZ/G2L 开发平台连接，罗技 C270 摄像头通过 USB 接口与瑞萨 RZ/G2L 开发平台连接，如图 4 所示。

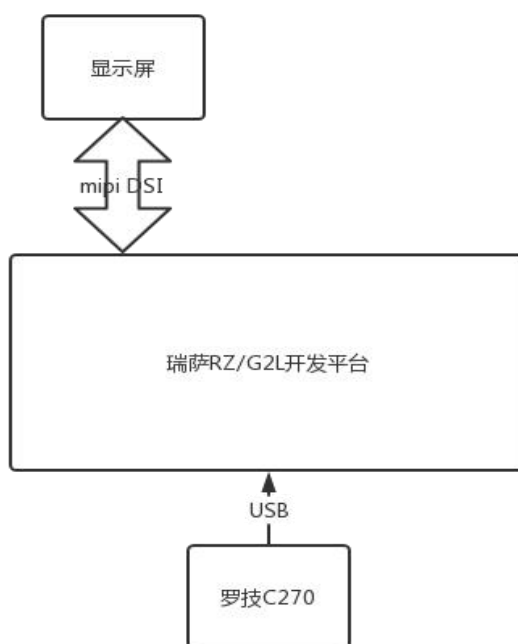


图 4 图像捕获硬件框图

3.软件环境：Pycharm、 Opencv 4.1.1、 Python 3.8、 Pyqt5 5.15.2

4.软件流程设计：

1)摄像头调取

采用 OpenCV 和 Python 中的相关操作函数调用摄像头；并通过 set()函数改变摄像头参数，设置摄像头捕获画面长为 1280px，宽为 720px。

在这里，选择罗技 C270 摄像头的理由是其自动对焦，为了方便截取出中心区域较为清晰的图像画面。

2)设置定时器用于刷新获取

设置定时器，使得每 30ms 捕获 1 帧画面，并循环捕获画面，以输出图片。

3)操作显示屏上的 PyQt5 交互页面

在 PyQt5 上显示图片需要将 OpenCV 中捕获的 mat 类型图片转换成 QImage 类型图片，可以通过 QImage()函数将 mat 类型图片改变成 QImage 类型图片。

如图 5 所示，用 PyQt5 作为图形交互页面，在摄像头所捕获图片的画面窗口之中，坐标（440，160）处显示一个大小为 400px×400px 的红色矩形框。

点击开启摄像头按钮即可开始图像捕获功能。

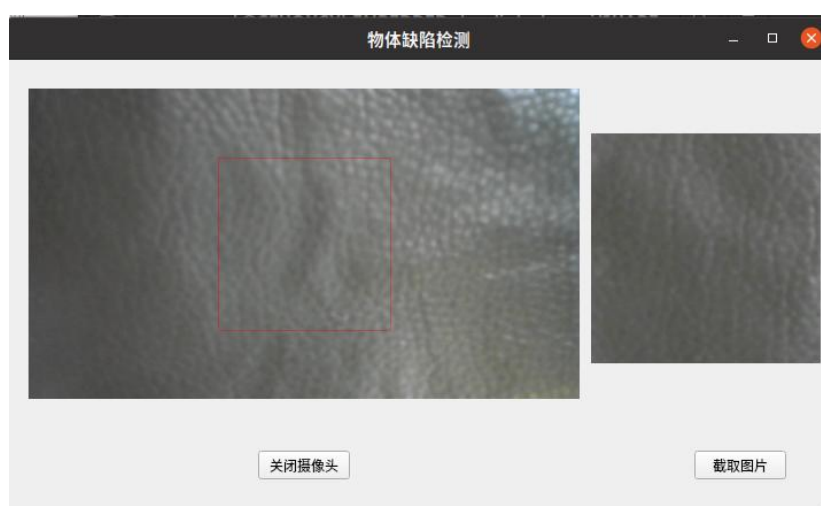


图 5PyQt 图形交互界面

点击“截取图片”按钮后，对当前帧的全部画面进行 BGR 转 RGB。截取裁剪出矩形框内的 400px×400px 图像，再将矩形框内 400px×400px 的图像等比例缩放转换为 256px×256px 的图像，作为待测物件图像输出，并保存至瑞萨 RZ/G2L 开发平台的指定文件夹内。由于缺陷检测时会自动下采样至统一尺寸，长宽比保持一致，能够尽可能保证特征不失真。

对于待测物件的图像捕获与剪裁，我们的设计流程图如图 6 所示。

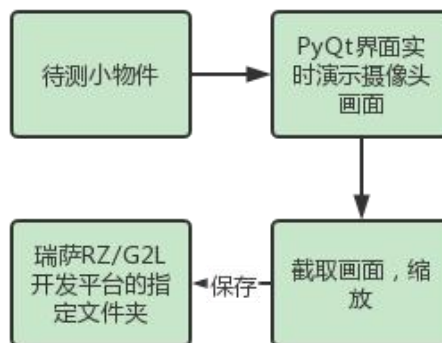


图 6 设计流程图：待测物件的图像捕获与剪裁

2.4 基于嵌入式深度学习的缺陷检测

2.4.1 模型原理

在这里，我们利用知识蒸馏原理的教师-学生框架进行缺陷检测，其中教师模型和学生模型均采用 ResNet-152 架构，但教师模型在训练时预先加载 ImageNet 预训练模型，学生模型则为初始化状态；同时，为提高模型检测的鲁棒性，引入特征金字塔。

训练过程中，只采用正常样本用于模型训练，旨在让学生模型学习到样本的无异常表示。当一张正常图像同时输入到师生网络时，教师网络得益于预训练模型能很好的抽取正常样本的特征，学生网络在训练中逐渐学习抽取更好的特征。模型训练过程中，教师模型冻结参数，学生模型不断的更新参数以寻求模型收敛的最优解。最后，当师生网络之间的差异逐渐减小到可接受范围内，我们停止训练。

测试过程中，模型测试既包含异常图片也包含正常图片。学生模型在训练过程中能够抽取正常样本特征，不能抽取异常特征，但教师模型依然可以高质量抽取异常样本特征。因此，当一张异常图像输入模型，不同尺度师生特征图进行余弦相似度计算得到不同尺度异常图，通过规范化和双线性插值后对应乘积得到融合后的异常图，与实际异常位置图片进行指标评估计算。

位于特征金字塔的不同层，提取的特征分辨率或尺度不同，低层对应高分辨率特征提取，在测试过程中生成高分辨率异常热力图像；高层对应低分辨率特征提取，在测试过程中生成低分辨率异常热力图像。不同层次生成不同尺度的异常热力图，同时，通过规范化和双线性上采样进行多尺度特征融合，得到融合后的综合异常热力图，综合异常热力图吸收迁移了来自特征金字塔不同层次的不同尺度特征，使得多种尺度特征融合互补。

简要归纳的原理图如图7^[4]所示。

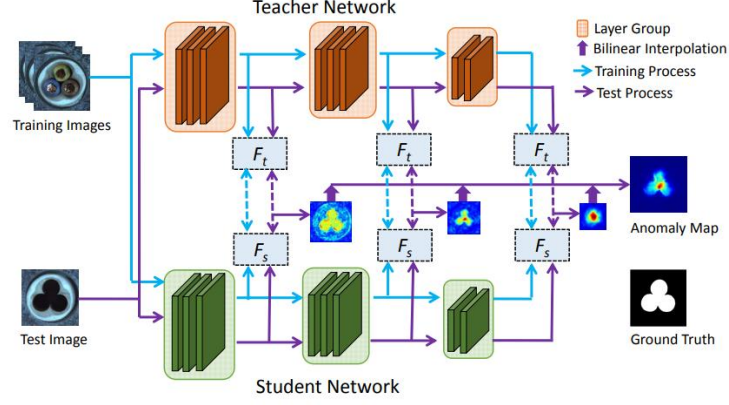


图 7 基于深度学习的物件缺陷检测实现原理图

2.4.2 设计计算

1. 模型的设计计算

1) 模型训练的设计计算

给定预训练的教师模型，初始化学生模型，目标是获得充分吸收教师模型多尺度层次特征知识的学生模型。其中，教师模型和学生模型具有相同的架构，都是残差网络 ResNet-152。训练数据集图像全部采用无缺陷图像，旨在使学生模型能够较好地模仿教师模型的特征金字塔输出。

(1) 系列预先定义。

给定无异常图像的训练数据集 $D = \{I_1, I_2, \dots, I_n\}$ ，匹配教师模型和学生模型特征金字塔对应的 L 层组提取的特征来捕获正常数据分布流形。

对于任一输入的训练图像 $I_k \in R^{w \times h \times c}$ ，其中， w 表示图像宽度， h 表示图像高度， c 表示图像颜色通道数。教师模型和学生模型的特征金字塔第 l 层组输出的特征图分别为 $F_t^l(I_k) \in R^{w_l \times h_l \times c_l}$ 和 $F_s^l(I_k) \in R^{w_l \times h_l \times c_l}$ 。

由于没有和物体的外观及位置相关的先验知识，现在假设训练集中所有图像的所有区域都是无异常的。

来自教师模型特征图中 (i, j) 坐标的特征向量为 $F_t^l(I_k)_{ij} \in R^{d_l}$ ，来自学生模型特征图中 (i, j) 坐标的特征向量为 $F_s^l(I_k)_{ij} \in R^{d_l}$ 。

(2) 计算教师-学生模型总损失函数。

我们对特征图中 (i, j) 坐标的损失函数定义为教师和学生模型中 L2 归一化后特征向量间的欧氏距离。

$$\ell^l(I_k)_{ij} = \frac{1}{2} \left\| \hat{F}_t^l(I_k)_{ij} - \hat{F}_s^l(I_k)_{ij} \right\|_{\ell^2}^2 \quad (1)$$

$$\hat{F}_t^l(I_k)_{ij} = \frac{F_t^l(I_k)_{ij}}{\|F_t^l(I_k)_{ij}\|_{\ell_2}}, \quad \hat{F}_s^l(I_k)_{ij} = \frac{F_s^l(I_k)_{ij}}{\|F_s^l(I_k)_{ij}\|_{\ell_2}} \quad (2)$$

在完成上面的工作后，定义输入图像 $I_k \in R^{w \times h \times c}$ 的总损失函数为特征图上每个坐标的平均损失。

$$\ell^l(I_k) = \frac{1}{w_l h_l} \sum_{i=1}^{w_l} \sum_{j=1}^{h_l} \ell^l(I_k)_{ij} \quad (3)$$

训练过程中，教师-学生模型总损失函数是不同金字塔层次损失的加权平均和。

$$\ell^l(I_k) = \sum_{l=1}^L \alpha_l \ell^l(I_k), s.t. \alpha_l \geq 0. \quad (4)$$

其中， α_l 是特征金字塔第 l 层的特征尺度权重，表征该层特征对缺陷检测的影响。

(3) 最小化损失函数，寻求模型收敛的最优解。

以最小化教师-学生模型损失为目标，不断更新学生模型，而不改变教师模型。在这里，定义数据集 θ 为训练集 $D = \{I_1, I_2, \dots, I_n\}$ 的非空子集，通过最小化损失

$$\ell_\theta = \frac{1}{|\theta|} \sum_{k \in \theta} \ell(I_k) \quad (5)$$

不断更新学生模型，从而使学生模型充分吸收教师模型多层次尺度特征知识。

2) 模型测试的设计计算

将测试图像同时输入教师和学生模型，比较两者过程中的特征金字塔。如若差异较大，则为缺陷图像，将各层特征差异多尺度融合，生成异常图。

(1) 系列预先定义。

令同时输入教师模型和学生模型的测试图像为 $J \in R^{w \times h \times c}$ ，令目标期望得到的异常图为 $\Omega^{w \times h \times c}$ ，令缺陷为 Ω_{ij} ，其中 $\Omega_{ij} \in [0,1]$ ，表示的坐标 (i, j) 的像素与训练数据流形的偏离程度。

教师模型和学生模型的特征金字塔第 l 层的特征图分别为 $F_t^l(J) \in R^{w_l \times h_l \times c_l}$ 和 $F_s^l(J) \in R^{w_l \times h_l \times c_l}$ 。

(2) 计算异常评分。

通过之前的公式我们可以计算得到第 l 层异常图 $\Omega^l(J)$ 的元素 $\Omega_{ij}^l(J)$ 的损失，异常图 $\Omega^l(J)$ 通过双线性插值上采样得到 $w \times h$ 的大小，生成的异常图定义为 L 等距上采样异常图的元素积。

$$\Omega(J) = \prod_{l=1}^L \text{Upsampled} \Omega^l(J)$$

如果测试图像中的任何像素异常，则将该图像指定为异常。最后，选择异常图中的最大值 $\max(\Omega(J))$

作为测试图像 $J \in R^{w \times h \times c}$ 异常评分。

2.在 Jetson Orin 中完成模型的训练和测试

1) 环境设置与搭建

Jetson Orin, 瑞萨 RZ/G2L 开发平台, Rocktech Displays RK070CU05H-CTG 显示屏, opencv-python4.5.2.54, Pytorch 1.12.0, protobuf3.9.2, scikit-learn 0.24.2, pytorch-lightning 1.3.3, torchmetrics 0.5

2) 数据集

MVTec 异常检测数据集包括15个类别, 有3629张图像用于训练和1725张图像用于测试。训练集只包含没有缺陷的正常图像。测试集含有各种缺陷和无缺陷的图像。

3) 模型在数据集上的训练和测试。

利用 Jetson Orin, 在 MVTec 异常检测数据集^[5]上找到与待测物件类别一致的数据子集, 选择随机梯度下降 (SGD) 作为优化器, 首先在数据子集的训练集上进行训练, 设置 epoch 为1000, batch size 为2。由于程序的设置, 在模型训练结束后, 自动转入在数据子集的测试集进行测试, 在此过程中, batch size 设置为2。

3.模型的嵌入式移植

将 Jetson Orin 中训练测试后的模型 (即 checkpoint 文件), 复制到瑞萨 RZ/G2L 开发平台的与训练测试时相对路径一致的文件夹内 (例如, 皮革的权重文件位置: /test/leather/ checkpoint), 将模型移植到开发平台。

4.利用训练测试后的模型执行小物件的缺陷检测

在瑞萨 RZ/G2L 开发平台的指定文件夹抓取待测物件图像, 输入移植到开发平台的模型, 进行测试, 执行待测小物件的缺陷检测。

通过多次实践, 发现在启用8GB 的 SWAP 交换空间后, 瑞萨 RZ/G2L 开发平台可以支持 batch size=1, 2, 4, 其中 batch size 设置为2时可以最快速地计算出结果。因此, 测试过程中, 设置 batch size 为2。

特征金字塔的不同层次提取的特征分辨率或尺度不同, 低层对应高分辨率特征提取, 生成高分辨率异常热力图像; 高层对应低分辨率特征提取, 生成低分辨率异常热力图像。

不同层次生成不同尺度的异常图, 我们对特征金字塔选取3层, 从上往下各层次分别生成 16px×16px 尺度异常图、32px×32px 尺度异常图、64px×64px 尺度异常图。通过规范化和双线性上采样进行多尺度特征融合, 得到融合后的256px×256px 综合异常热力图。此外, 将综合异常热力图和待测物件图像进行合成, 得到综合热力实物合成图。最终, 将系列缺陷检测的结果输出图像全部保存至瑞萨 RZ/G2L 开发平台的指定文件夹, 可以通过显示屏, 查看文件夹中缺陷检测的结果图像文件。

如图8所示, 简洁说明了基于嵌入式深度学习的缺陷检测的设计计算。

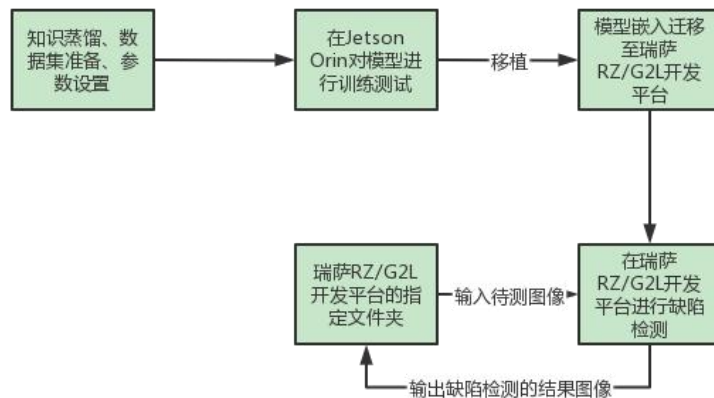


图 8设计计算图：基于嵌入式深度学习的缺陷检测

2.5 物件缺陷情况的可视化网页展示

2.5.1PC 端访问瑞萨 RZ/G2L 开发平台的文件

我们将瑞萨 RZ/G2L 开发平台的以太网接口通过 ENET 网线连接到 PC 端所连接的路由器，使 PC 端和瑞萨 RZ/G2L 开发平台局域网互连。当 PC 端欲从瑞萨 RZ/G2L 开发平台获取系列缺陷检测的结果输出图像，可以通过 samba 服务，搜索目标设备的指定文件夹，定位访问，进行抓取。

1.硬件设备：瑞萨 RZ/G2L 开发平台、路由器、ENET 网线

2.硬件连接：如图 9 所示，将开发平台通过 ENET 网线连接到路由器。

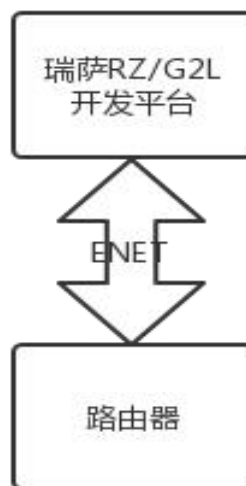


图 9 网络共享硬件框图

3.软件使用：samba

4.软件流程设计：

1)配置网络

基于 Ubuntu-base 20.04 的瑞萨 RZ/G2L 开发平台提供两个千兆以太网接口。

为了方便将本产品轻松接入互联网，我们将第一端口(eth0)设置为静态；此外为方便在无互联网的局域网内使用 IP 地址控制或访问本产品，我们将端口二(eth1)设置为静态。

我们通过在 /etc/systemd/network/80-dhcp.network 中设置 eth0 为动态 IP；在 /etc/systemd/network/50-static.network 中设置 eth1 为静态 IP。

通过命令“systemctl enable systemd-networkd.service”设置开机自动配置网络。

2)开发平台的指定文件夹访问

利用 samba 服务，通过“file://设备名/共享目录/目标文件” (适用于 Google chrome 核心的浏览器)或“\\设备名\共享目录\目标文件” (适用于微软的 explorer 或 Internet Explorer)来访问目标设备（此处即瑞萨 RZ/G2L 开发平台）的目标文件。

在这里，通过边缘设备的网络设备名进行访问，进而替代直接 IP 地址访问，有效解决了使用动态 IP 时，IP 地址变化导致访问出错等情况。

2.5.2Web 开发准备

为方便测试结果的条理化、可视化展示，我们访问抓取到瑞萨 RZ/G2L 开发平台中指定文件夹中的系列图像数据后，利用 Web 前端知识设计网页，用于现场展示待测物件的缺陷检测以及以往测试案例的输入输出系列图（即输入的待测物件图像、输出的综合异常热力图、输出的综合热力实物合成图像），特征金字塔各尺度层次异常热力图等。

语言版本：HTML 5、CSS 3、JavaScript ECMAScript 2022(调用了 JavaScript 的 JQuery 库)

编程工具：Visual Studio Code 1.70

浏览器：360 极速 X 21.0.1130.0

2.5.3 网页的页面结构

如图 10 所示，我们总共设计了 16 页网页，通过链接跳转的方式相互连接，统一集成为一个本地网站。



图 10 网页文件总览

以首页（00-index.html）为例，各页面的设计结构从上往下分为 4 个模块：

头部（header），尺寸为 1265px×42px；

横幅（banner），尺寸为 1265px×510px；

主体（bodybox），尺寸为 1265px×900px；

尾部（footer），尺寸为 100%×250px。

如图 11 所示，首页的完整页面结构如下，其他网页页面结构与此类似，只是各模块的尺寸参数有所调整，填充内容有所改变，不另进行一一赘述。

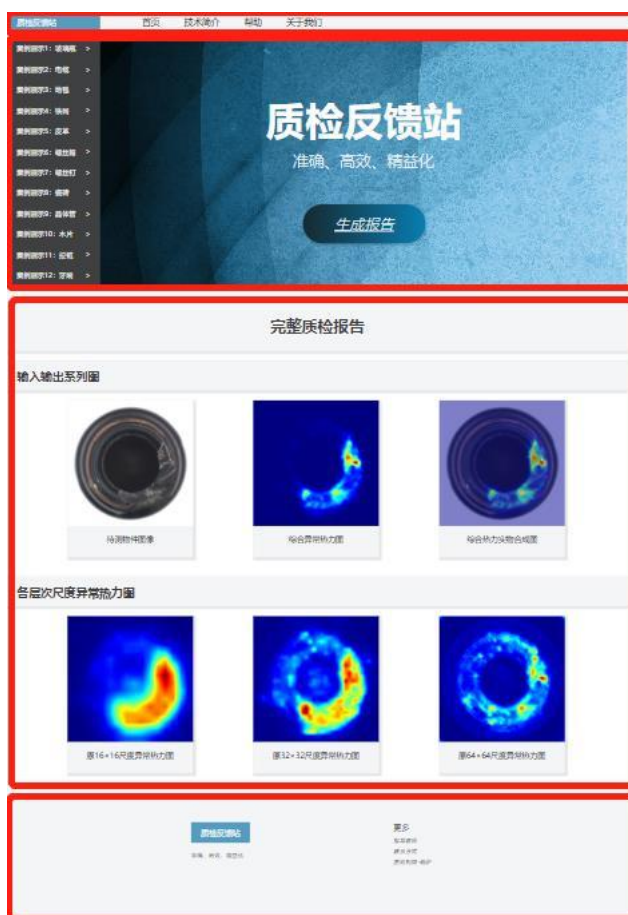


图 11 首页的页面结构

2.5.4 网页的功能设计

Web 质检展示的功能使用流程如下：

首页（00-index.html）部分界面如图 12 所示。该页主要功能为：完成待测物件的嵌入式深度学习缺陷检测后，点击首页“生成报告”按钮，首页将会生成待测物件的缺陷质检报告。生成的缺陷质检报告，包含有输入输出系列图，特征金字塔不同层次尺度的异常热力图。



图 12 “质检反馈站” 首页部分界面

点击首页左侧竖栏的案例展示，能够看到历史案例的缺陷质检结果展示，例如点击“案例展示 1”，会跳转到案例展示 1 的页面（01-案例展示 1.html），如图 13 所示。与首页的缺陷质检报告结果一样，我们的案例展示同样包括输入输出系列图与特征金字塔不同层次尺度的异常热力图。

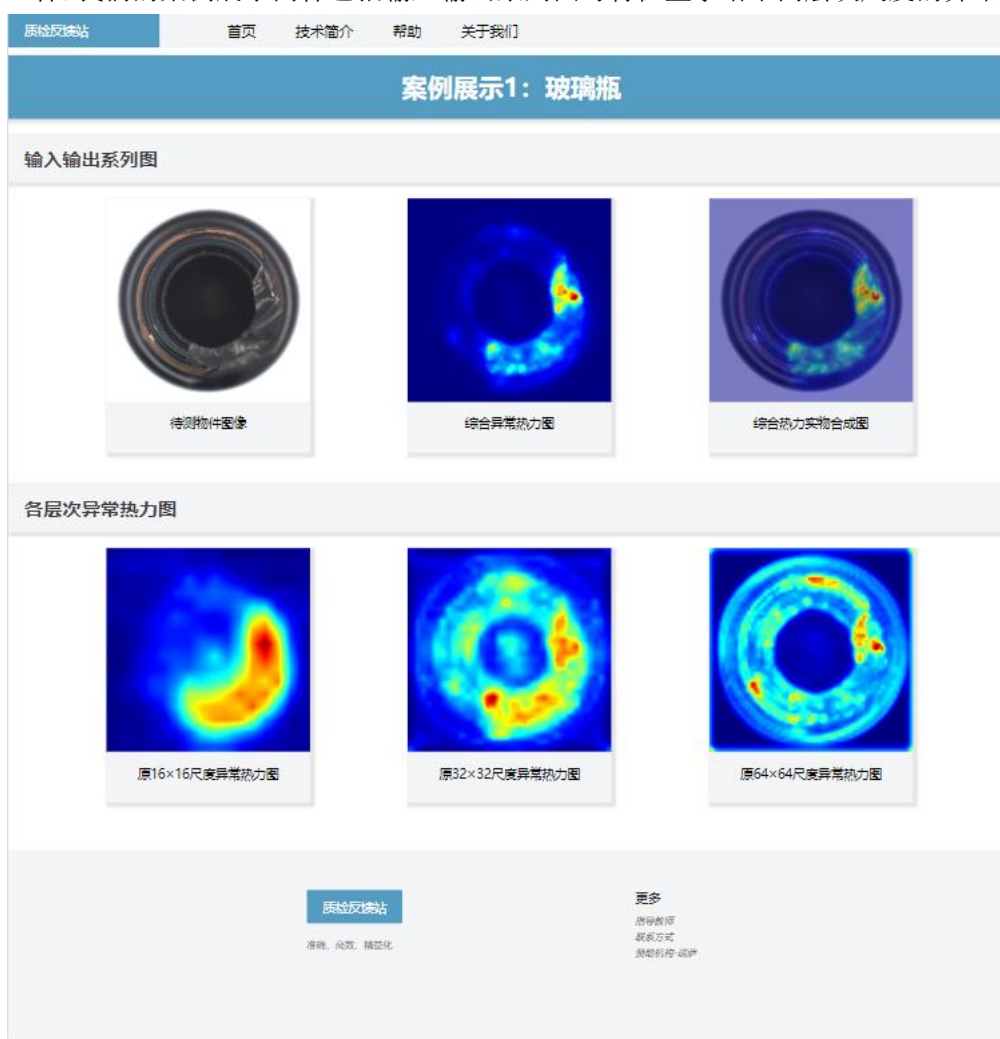


图 13 案例展示 1 的页面

图 14 简要总结了关于缺陷检测结果的可视化网页展示的设计实现思路。

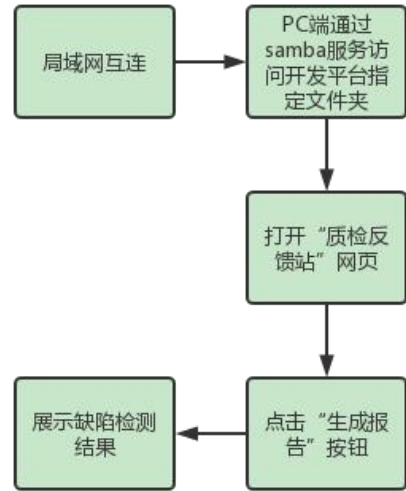


图 14 设计实现思路：缺陷检测结果的可视化网页展示

2.6 功能与指标

2.6.1 功能

在2.1系统方案已经提到过，本作品按功能划分板块进行设计，3个板块分别为：待测物件的图像捕获与剪裁、基于嵌入式深度学习的缺陷检测、缺陷检测结果的可视化网页展示。

第1个板块的功能是图像捕获与剪裁，使连接到瑞萨 RZ/G2L 开发平台的摄像头实时获取画面，经过裁剪、缩放后输出待测物件图像，将待测图像保存至开发平台的指定文件夹。

第2个板块的功能是缺陷检测，将 Jetson Orin 中训练测试后的模型嵌入式移植到瑞萨 RZ/G2L 开发平台，随后从开发平台指定文件夹抓取待测图像，输入移植到开发平台的模型进行测试，最终输出综合异常热力图、综合热力实物合成图、各尺度层次缺陷热力图，并保存至开发平台的指定文件夹。

第3个板块的功能是 Web 质检展示，使 PC 端和瑞萨 RZ/G2L 开发平台局域网共享，在 PC 端通过 samba 服务访问开发平台的指定文件夹，抓取系列缺陷检测结果的输出图像，点击“生成报告”后，展示在“质检反馈站”首页。

2.6.2 指标

工业质检中的性能指标一般是漏检率、误检率、准确率。但由于现目前我们没有工业级流水的环境，待测产品的数目过少，所以只能对几件物件样品进行多次重复的缺陷检测，从而获得准确率等指标。

本作品的主要功能是对小物件产品进行缺陷检测，缺陷检测模型的性能是影响本作品性能的核心因素，我们可以根据模型性能来评判作品性能。因此，我们采用评判模型性能的图像级 AU-ROC，像素级 AU-ROC，PRO-score 作为缺陷检测结果的评估指标，衡量本作品的性能，

其中，3个指标均属于正向型指标；同时，异常热力图、热力实物融合图也可以直观反映待测物件图像是否存在缺陷，可以作为检测结果的可视化指标。

AUROC 通过 ROC 曲线与坐标轴之间的面积大小来反应的性能，这是一个与阈值无关的指标，常用于二分类算法的评估。

PRO-score 统计的是不同阈值下的区域重叠率，以二值化后连通域和真值图之间的相对重叠率作为每一个阈值下的模型分类性能。

热力图通过对像素点或像素块着色来显示异常评分数据，按颜色映射的指定规则进行绘图。缺陷检测生成输出16px×16px 尺度异常热力图、32px×32px 尺度异常热力图、64px×64px 尺度异常热力图、多尺度特征融合后的256px×256px 综合异常热力图。此外，我们将综合异常热力图和待测物件图像进行合成，得到综合热力实物合成图。

3. 作品测试与分析

3.1 测试设备和环境搭建

1. 测试设备

瑞萨 RZ/G2L 开发平台，罗技 C270摄像头，Jetson Orin，Rocktech Displays RK070CU05H-CTG 显示屏，路由器，Thinkpad P15v(Windows 10 Pro)，1转4USB 扩展器，USB 键盘，64GB Micro SD，DDR4 800MHz 2GB

2. 测试环境搭建

瑞萨 RZ/G2L 开发平台的嵌入式实时操作系统：Ubuntu 20.04 L4T

电脑操作系统：Ubuntu 20.04 L4T 虚拟机

深度学习环境：opencv-python4.5.2.54, Pytorch 1.12.0, protobuf3.9.2, scikit-learn 0.24.2 , pytorch-lightning 1.3.3, torchmetrics 0.5

Web 浏览器环境：360 极速 X 21.0.1130.0

3.深度学习模型数据集及评估指标

关于待测物件，我们选择了有缺陷的木块、皮革、拉链这3个小物件。

关于深度学习模型涉及到的数据集，我们采用上文提到的 MVTec 异常检测数据集。

数值评估指标：图像级 AU-ROC ， 像素级 AU-ROC ， PRO-score。

可视化评估指标：各尺度层次缺陷热力图、综合异常热力图、综合热力实物合成图。

3.2 测试方案

3.2.1 系统性能测试方案

系统性能体现在深度学习模型的性能，对缺陷检测系统的性能进行测试，在于对其中深度模型的性能进行测试。

准备工作：教师模型和学生模型均采用 ResNet-152架构，选择 ResNet-152的前3个块(conv2_x,

conv3_x, conv4_x)作为教师和学生模型的金字塔特征提取器。教师模型在训练时预先加载 ImageNet 预训练模型,学生模型则为初始化状态。选择随机梯度下降(SGD)作为优化器,设置 epoch 数为1000, batch size 为2, 将 MVTec 数据集中所有图像的大小调整为256px×256px。

在 MVTec 数据集寻找待测物件(木块、皮革、拉链)的同类物体图像数据集,在 Jetson Orin 用其中只包含无缺陷图像的训练集对学生-教师模型进行训练至收敛,再使用无缺陷和缺陷图像混合的测试集进行测试,得到模型的指标评估数据图像级 AU-ROC ,像素级 AU-ROC ,PRO-score。

3.2.2 系统功能测试方案

系统功能测试方案与系统设计方案保持架构一致,都按照功能划分为3个板块。在这里,我们的系统功能测试方案由3部分组成:图像捕获与剪裁的测试、缺陷检测测试、Web 质检展示测试。

1.图像捕获与剪裁的测试

所有设备就绪,准备待测物件,将待测物件摆放至罗技 C270摄像头画面的中央红色边框内。

按下“截取图片”,捕获当前画面,应用程序将裁剪出红色边框内的图像,并将其等比例缩放为256px×256px 尺度,作为待测物件图像000.png。

在瑞萨 RZ/G2L 开发平台中创建一个指定文件夹“SHARE”,将000.png 按指定路径保存至该文件夹。

2.缺陷检测测试

将 Jetson Orin 中训练测试后的模型(即 checkpoint 文件),复制到瑞萨 RZ/G2L 开发平台的与训练测试时相对路径一致的文件夹内,将模型移植到瑞萨 RZ/G2L 开发平台。

将瑞萨 RZ/G2L 开发平台“share”文件夹中的 test.jpg,输入移植到开发平台的模型,进行测试,执行小物件的缺陷检测,输出多尺度特征融合前,特征金字塔各尺度层次的不同尺度热力图,以及多尺度特征融合后,与输入待测物件图像尺寸一致的异常热力图及热力实物融合图。

将缺陷检测过程的输出图像等数据,按指定路径保存至开发平台的“share”文件夹。

3.Web 质检展示测试

将瑞萨 RZ/G2L 开发平台的以太网接口通过 ENET 网线接入路由器,使开发平台和 PC 端局域网互连,PC 端通过 samba 服务,搜索目标设备的目标文件,访问瑞萨 RZ/G2L 开发平台“share”文件夹。

提前打开“质检反馈站”网页,进入网页首页。点击网页首页右上角窗口缩小按钮,隐藏网页后等待。

当待测物件的缺陷检测完成后,显示被隐藏的网页,点击网页首页正中央的“生成报告”按钮,下滑查看生成的完整质检报告,包括输入输出系列图像,特征金字塔各尺度层次的不同尺度热力图,在这里,所有图像都被等比例放大为256px×256px。

3.3 测试结果分析

3.3.1 系统性能测试结果分析

在 MVTec 异常检测数据集选择与待测物件属于同一类别的数据子集,在 Jetson Orin 中将准备好的学生-教师模型在该数据子集的训练集上进行训练,直至损失收敛至最优,木块、皮革、拉

链损失函数值随时间变化的曲线如图 15、图 16、图 17 所示。



图 15 模型训练的损失函数曲线图（木块）



图 16 模型训练的损失函数曲线图（皮革）

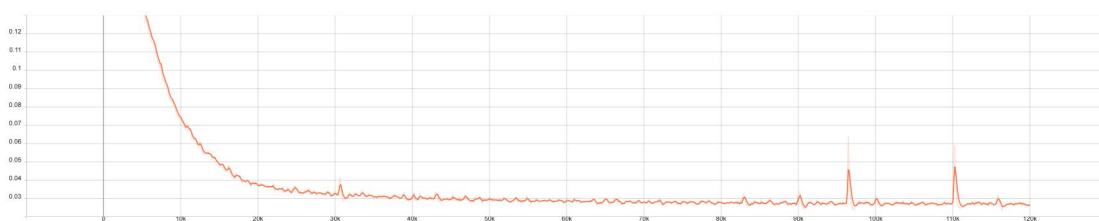


图 17 模型训练的损失函数曲线图（拉链）

在 Jetson Orin 中将训练之后的学生-教师模型在该数据子集的测试子集测试。学生-教师模型经过以上的训练测试过程后,最终得到表 3 所示的,本模型在待测物件图像类别的图像级 AUROC、像素级 AUROC、PRO-score。这 3 个指标都属于正向型指标,数值越大,表征评价越好。

表 3 本模型在待测物件类别实验的评估指标数据

检测物件名称	图像级 AUROC	像素级 AUROC	PRO-score
木块	0.976	0.988	0.951
皮革	0.988	0.993	0.980
拉链	0.974	0.985	0.952

在这里,为了方便对结果指标有更加清晰的认识,验证我们的模型方法具有较好的性能,我们做了额外的大体量工作:在选取本模型方法的同时,额外选取 Cut-Paste^[6]、Patch-SVDD^[7]、SPADE^[8]这 3 种常见缺陷检测模型方法,将每一种方法在 MVTec 异常检测数据集对全部 15 个类别分别进行训练测试,得到 15×4 组图像级 AUROC、像素级 AUROC、PRO-score 数据。对每一个模型方法的每一个指标,我们取该方法上该指标在全部 15 个类别下结果的平均值,代替该方法的该指标结果值。最终得到 4 组图像级 AUROC、像素级 AUROC、PRO-score 数据,如表 4 所示。

表 4 各模型的评估指标数据比较

模型方法名称	图像级 AUROC	像素级 AUROC	PRO-score
Cut-Paste	0.952	0.960	-
Patch-SVDD	0.921	0.957	-
SPADE	0.855	0.965	0.917
Ours	0.955	0.970	0.921

对表 4 所展示的 4 种模型方法的 3 种指标进行比较分析，可以得出结论：本模型的图像级 AUROC、像素级 AUROC、PRO-score 数据皆优于其他 3 种模型方法，证明本模型的综合性能较为优越。其中，本模型图像级 AUROC 值的优越性说明本模型的缺陷图像分类能力强，多尺度缺陷检测的效果较好，在图像层面缺陷检测取得了预期效果；像素级 AUROC 值、PRO-score 值的优越性证明本模型的缺陷识别精度高，缺陷定位能力强，在像素层面缺陷检测取得了预期效果。

3.3.2 系统功能测试结果分析

遵循系统功能测试方案，我们对作品进行功能测试，得到了以下结果。在这里，我们的功能测试结果分析对应上文的系统功能测试方案，依旧划分为 3 个板块：图像捕获与剪裁的结果分析、缺陷检测的结果分析、Web 质检展示的结果分析。

1. 图像捕获与剪裁的结果分析

我们将待测物件调整置于罗技 C270 摄像头视野画面中央的红色边框内。

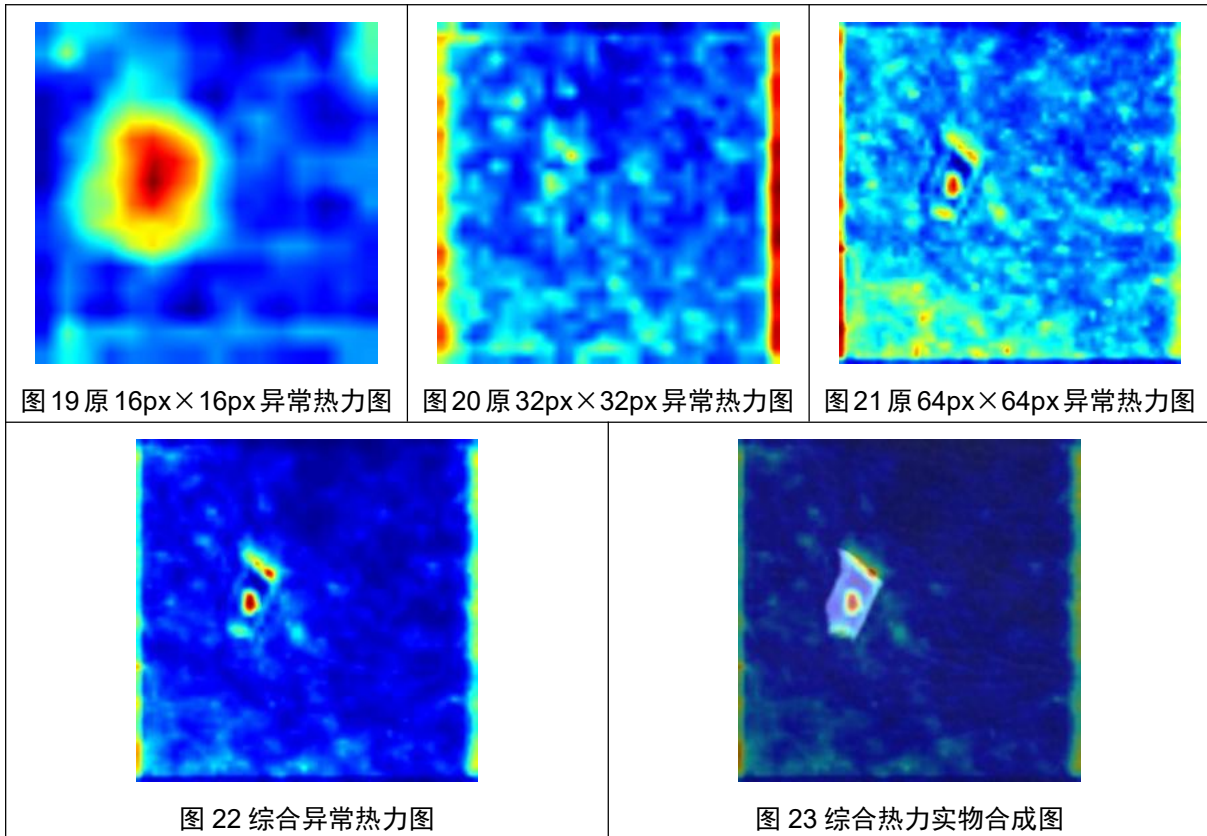
按下“裁取图片”后，如图 18 摄像头实时画面被捕获，应用程序裁剪出 400px×400px 红色边框内的图像，并将其等比例缩放为 256px×256px 尺寸，作为待测物件图像 000.png。



图 18 待测物件图像 000.png

2. 缺陷检测的结果分析

将 Jetson Orin 中训练测试后的模型嵌入式移植到瑞萨 RZ/G2L 开发平台，将 000.png 输入移植到瑞萨 RZ/G2L 开发平台的模型进行缺陷检测，输出一系列结果图像。包括多尺度特征融合前特征金字塔各尺度层次的不同尺度热力图（下面已将不同尺度异常热力图全部等比例扩大为 256px×256px 尺度），如图 19、20、21 所示；此外，还包括多尺度特征融合后，256px×256px 尺度综合异常热力图（如图 22 所示），以及综合异常热力图转换而来的综合热力实物合成图（如图 23 所示）。



根据一系列的输出结果图像进行分析，可以直观认识到，特征金字塔的低层对应高分辨率特征提取，生成的高分辨率异常热力图像（例如 64px×64px 尺度异常热力图）能够对缺陷进行更加精确的定位，尽管也会包含较多的背景噪声。而特征金字塔的高层对应低分辨率特征提取，生成的低分辨率异常热力图像（例如 16px×16px 尺度异常热力图）只能对缺陷区域进行大致定位，但能过滤更多背景噪声。

而多尺度特征融合之后的 256px×256px 尺度综合异常热力图，有效吸收迁移了特征金字塔不同层次的不同分辨率特征，使得多种分辨率特征即多种尺度特征融合互补，最终呈现的缺陷检测效果最好。此外，由 256px×256px 尺度综合异常热力图衍生转换而来的综合热力实物合成图（如图 23 所示），增强和突出了待测物件缺陷区域与正常区域的对比，更加直观清晰。

3.Web 质检展示的结果分析

设置瑞萨 RZ/G2L 开发平台和 PC 端通过路由器实现局域网互连，如图 24、图 25 通过在开发平台端使用 ifconfig 命令查看当前 ip 地址，并在 windows 端 CMD 内使用命令 ping+开发平台 IP 地址，若显示接入的延时，则说明两台设备都接入到同一局域网内。

如图 26 所示，使用 samba 服务，搜索//GEHONGYI-EMBEDDED/SHARE 定位访问“SHARE”文件夹的系列图像。

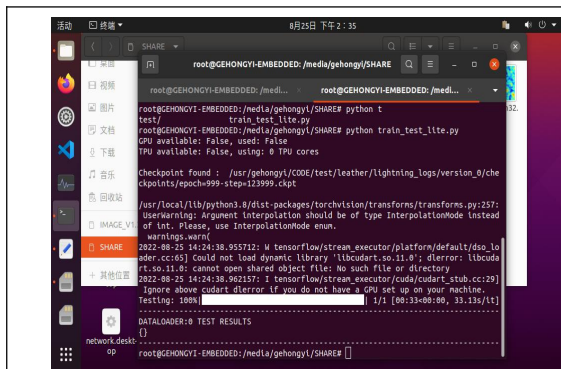


图 24 开发平台端查看 IP 地址

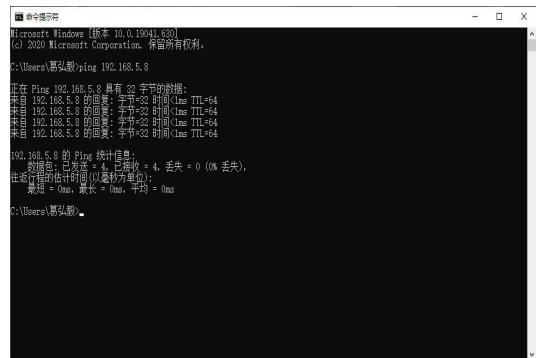


图 25 Windows 端查看 IP 地址

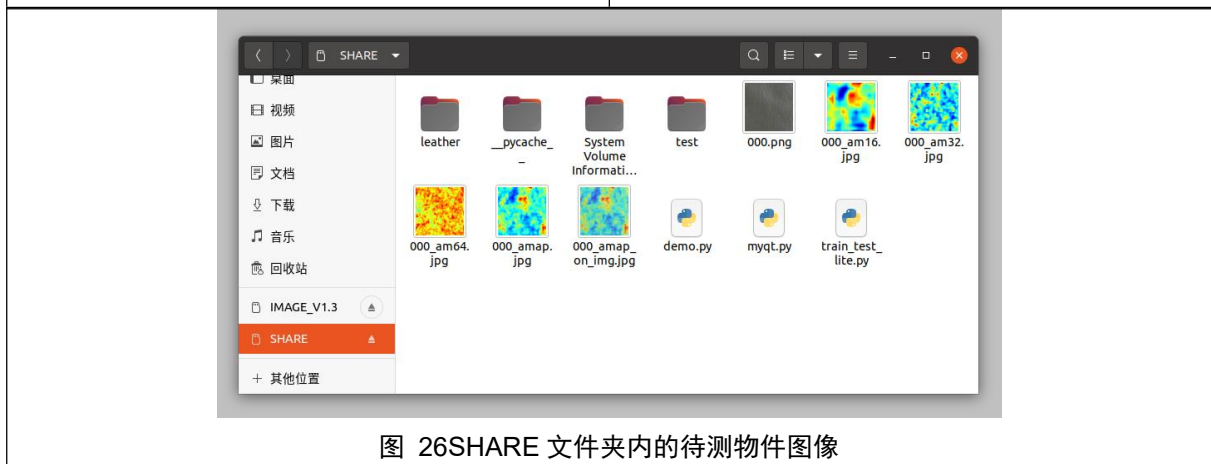


图 26 SHARE 文件夹内的待测物件图像

在整个测试过程开始前，提前打开“质检反馈站”网页，进入首页，此时图像捕获与剪裁、缺陷检测暂未进行，首页“完整质检报告”部分尚且空白，如图 27、图 28 所示，隐藏浏览器窗体进行等待。



图 27 刷新前的空白“完整质检报告”（上）



图 28 刷新前的空白“完整质检报告”（下）

如图 29、图 30 所示，在图像捕获与剪裁、缺陷检测完成后，点击“生成报告”按钮，生成质检报告，包括输入输出系列图（待测物件图像、综合异常热力图、综合热力实物合成图），各层次尺度异常热力图（16px×16px 尺度异常热力图、32px×32px 尺度异常热力图、64px×64px 尺度异常热力图）。

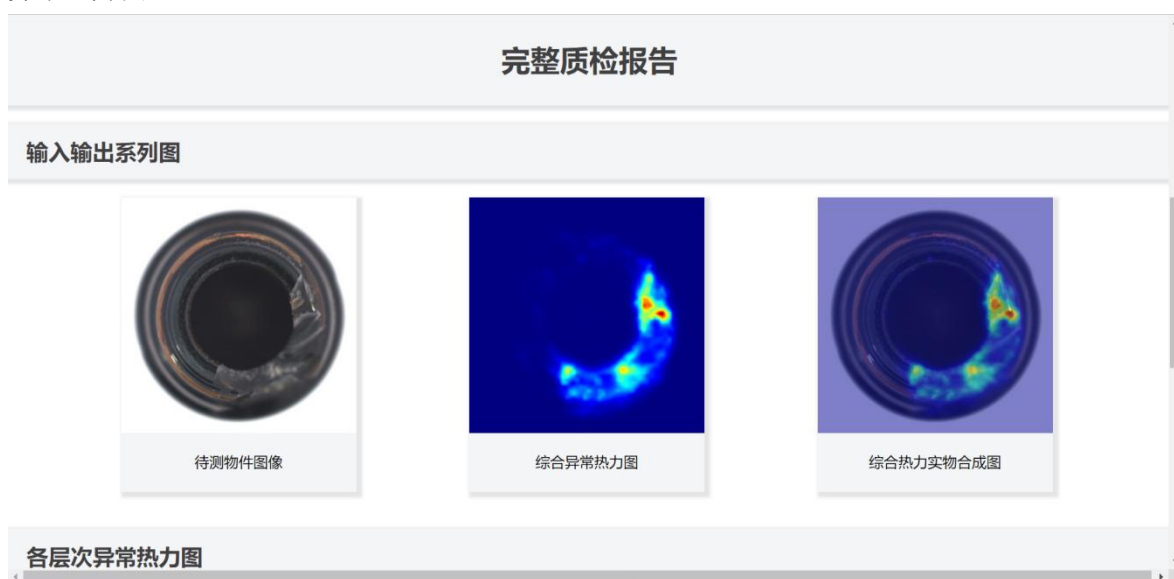


图 29 刷新后生成的“完整质检报告”（上）

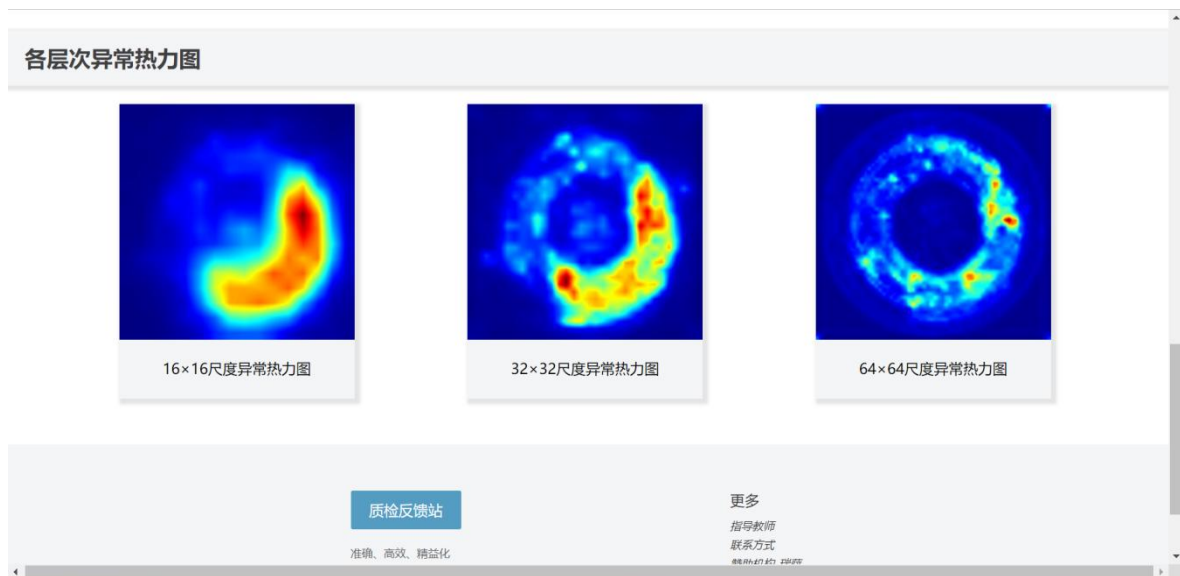


图 30 刷新后生成的“完整质检报告”（下）

4. 创新性说明

本作品的主题是零件等小物件产品的表面缺陷检测，我们的设计在应用和技术上分别都有一些创新点。

本作品在**技术**方面的创新点主要包括：

- 1.符合网络化主题，实时在 PC 端网页查看缺陷检测的结果。
- 2.将深度学习置于嵌入式系统训练测试。
- 3.模型在嵌入式设备的移植。
- 4.归属于边缘计算，降低了能耗和延迟。

本作品在**应用**方面的创新点主要包括：

- 1.针对不同产品的不同缺陷进行表面缺陷检测，质检把关效果较好，推广和普适性较好。
- 2.集成及留存产品缺陷的信息，方便人员溯源，正反馈促进精益化生产。
- 3.配套设计了显示待测物件测试结果的可视化网页，使非开发人员更易理解。

5. 总结

5.1 作品立意

本作品紧扣大赛“网络化”、“图像检测与识别”的主题，立足于机器视觉领域的相关技术，设计实现了小物件产品的智能化表面缺陷检测。

而我国正处于由劳动密集型向技术密集型转型的重要时期，因此，本作品的设计立意从长远来看，符合技术发展和当下国情，有较大的应用和推广价值。

5.2 优点与缺点

5.2.1 优点

总结下来，本作品的优点如下：

- 1.本作品的设计实现立足于机器视觉领域，在大批量重复性工业生产过程中，提高了缺陷检测工作的灵活性和自动化程度。
- 2.将深度学习置于嵌入式系统训练测试，减轻了云端计算的限制和约束，更加便捷。
- 3.统计并记录产品的缺陷信息，对产品生产进行正反馈调节，充分发挥缺陷检测这一任务的价值。
- 4.配套设计的可视化网页展示测试取得的缺陷信息、数据图表，方便非开发人员查看和理解。

5.2.2 缺点

与此同时，本作品需要优化的缺点如下：

- 1.本作品的训练测试过程对样本集的要求较高。
- 2.由于时间较短，暂时未专门设计后端数据库存放历史测试结果，所有测试结果保留在本地磁盘。
- 3.本作品只是我们设计的智能化小物件产品表面缺陷检测的一个初步成品，在真正投入工业生产过程的实际应用之前，还需要大量的改良和维护。

附录

```
/*部分源代码*/
import argparse
import torch
from torch.nn import functional as F
from torch import nn
from torchvision import transforms, datasets
from torch.utils.data import Dataset, DataLoader
import cv2
import numpy as np
import os
import glob
import shutil
import time
from torchvision.models import resnet152
from PIL import Image
from sklearn.metrics import roc_auc_score
from torch import nn
import pytorch_lightning as pl
import string
import random
from sklearn.metrics import confusion_matrix
if __name__ == '__main__':

    device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
    args = get_args()

    trainer = pl.Trainer.from_argparse_args(
        args,
        default_root_dir=os.path.join(args.project_path, args.category),
        max_epochs=args.num_epochs,
        gpus=[0]
    ) #, check_val_every_n_epoch=args.val_freq, num_sanity_val_steps=0) #,fast_dev_run=True)

    if args.phase == 'train':
        model = STPM(hparams=args)
        trainer.fit(model)
        trainer.test(model)
    elif args.phase == 'test':
        # select weight file.
        weights_file_path = auto_select_weights_file(
            args.weights_file_version
        ) # select latest weight if args.weights_file_version == None
```

```
if weights_file_path != None:
    model = STPM(hparams=args).load_from_checkpoint(weights_file_path)
    trainer.test(model)
else:
    print('Weights file is not found!')
```

参考文献:

- [1] 王林和尹国珍.知识蒸馏和师生学习视觉情报:回顾和新的前景.arXiv 预印本 arXiv:2004.05937, 2020.
Lin Wang and Kuk-Jin Yoon. Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks. arXiv preprint arXiv:2004.05937, 2020.
- [2] 基于生成对抗网络的无监督异常检测指导标记发现.IPMI,2017.
Thomas Schlegl, Philipp Seeböck, Sebastian M. Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In IPMI, 2017.
- [3] 安镇元和赵成宗.基于重构概率的变分自编码器异常检.2015,首尔大学数据挖掘中心技术报告.
Jinwon An and Sungzoon Cho. Variational autoencoder based anomaly detection using reconstruction probability. Technical report, SNU Data Mining Center, 2015.
- [4] 王刚.韩松.丁娥等.学生-教师特征金字塔匹配的无监督异常检测[J].
Wang G, Han S, Ding E, et al. Student-teacher feature pyramid matching for unsupervised anomaly detection[J].
- [5] 保罗·伯格曼,迈克尔·福泽,大卫·萨特莱格,卡斯滕·斯泰格.不知情的学生:基于判别潜嵌入的师生异常检测.2020,CVPR
Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In CVPR, 2020.
- [6] 李春良,孙启赫,尹晋成,托马斯·菲斯特.Cutpaste:用于异常检测和定位的自我监督学习.2021,CVPR.
Chun-Liang Li, Kihyuk Sohn, Jinsung Yoon, and Tomas Pfister. Cutpaste: Self-supervised learning for anomaly detection and localization. In CVPR, 2021.
- [7] 李智勋,尹圣罗.Patch svdd:用于异常检测和分割的 Patch 级 svdd.2020,ACCV.
Jihun Yi and Sungroh Yoon. Patch svdd: Patch-level svdd for anomaly detection and segmentation. In ACCV, 2020.
- [8] 科恩,叶迪德·何申.基于深度金字塔对应的子图像异常检测.arXiv: 2005.02357,2020.
Niv Cohen and Yedid Hoshen. Sub-image anomaly detection with deep pyramid correspondences. arXiv:2005.02357, 2020.