**一、环境配置方面**

1）所有文件和工程源码都在D:\zhanghuan\_work文件夹中，只有数据集在路径D:\Downloads下

2）路径：D:\zhanghuan\_work\开题相关资料，包含了开题报告等相关资料

3）路径：D:\zhanghuan\_work\working，包含了相关SDK文档和资料

4）路径：D:\zhanghuan\_work\test\_sdk是pycharm项目工程文件夹，其他的项目工程都在D:\zhanghuan\_work\pycharmpro文件夹中

5）路径：D:\zhanghuan\_work\paper，包含了结构光技术应用的相关文献资料

6）cuda的版本是11.1

7）可适配pytorch的版本是1.8.0或1.9.0等版本，请不要使用pytorch>=1.10.0版本，torchvision=0.11.1+cu111无win环境下的whl资源，所以不可用。如何需要使用请采用conda的安装方法或者采取手动源码编译生成环境

8）一般使用的是1.9.0版本，命令如下：

pip install torch==1.9.0+cu111 torchvision==0.10.0+cu111 torchaudio==0.9.0 -f https://download.pytorch.org/whl/torch\_stable.html

9）VPN用的是clash for windows软件，具体VPN账户需要自行购买和使用

**二、结构光相机方面**

1）使用api接口控制结构光相机必须将防火墙关闭

3）设备启动连接较慢

4）结构光扫描时间间隔不能过短

5）使用mesh-eye-viewer可以将其软件设置为白名单，不关闭防火墙

6）激光条纹的参数信息如果调整，需要一同调整视场范围。否则会出现视场范围大，激光条纹扫描区域小，需要多次扫描才可以完成一次视场扫描。如果经过设置出现上述情况，可恢复的一种方法是，打开Mech-Eye Viewer，连接相机，调整右侧列表的参数组为默认，即可恢复默认出场

7）点云方面默认采用open3d库开发，还可以采用python-pcl或者PCL（C++库）

8）建议启用巨型帧，加快数据传输（目前暂时未启用）

9）标定，这里的标定指的是调整相机成像效果以及相机内参（在Mech-Eye Viewer软件中的内参工具中可手动调整）（目前暂时未标定，还需标定）

参考链接如下：

https://docs.mech-mind.net/2.1/zh-CN/MechEye/MechEyeViewer/Tools/CheckCameraIntrinsicParameters/CheckCameraIntrinsicParameters.html

10）调整参数范围（图像质量），可参考如下连接

参考链接如下：

https://docs.mech-mind.net/zh/eye-3d-camera/2.1.0/viewer/parameters-laser-v4.html

**三、数据集增强方法与标注**

3.1用于二维图像数据增强

1）imgaug扩增方法（离线增强）

网址链接：https://github.com/aleju/imgaug

2）albumentations数据增强

3）torchvision.transforms数据增强

3.2二维图像标注方法

1）Labelme已下载可用（发布的exe版本）

2）make-sense在线工具，可以使用

3）基于SAM的开源半自动标注方法

**四、场景图像分类**

本地Conda环境名称：classification

本地pycharm项目：classification

MobileViT已经在flower数据集上验证。

flower数据集遵循imagenet数据集格式。

**五、视觉缺陷检测**

5.1二维检测方法

1）anomalib集成方法（stfpm已实验）

链接：https://github.com/openvinotoolkit/anomalib

本地Conda环境名称：anomalib\_env

本地pycharm项目：anomalib

参考文档：https://openvinotoolkit.github.io/anomalib/tutorials/training.html

2）目标检测方法（YOLOv8）

本地Conda环境名称：test\_sdk

本地pycharm项目：test\_sdk

参考链接：https://github.com/ultralytics/ultralytics

data文件夹是采集的室内样例数据

Capture.py和now\_demo\_loop.py是用于结构光相机获取的demo

其他test\_line.py等文件是pyransac的测试demo

mecheye\_python\_samples包含了mecheye的python控制样例源码

5.2三维检测方法

1）M3DM（已配置环境和下载数据集）

本地Conda环境名称：M3DM

本地pycharm项目：M3DM

参考链接：https://github.com/nomewang/M3DM

2）Shape-Guided（已配置环境和下载数据集）

本地Conda环境名称：Shape-Guided

本地pycharm项目：Shape-Guided

参考链接：https://github.com/jayliu0313/Shape-Guided

3）3D-ADS（待实验）

https://github.com/eliahuhorwitz/3D-ADS

mvtec-3d-ad多模态异常检测数据集链接：

https://www.mvtec.com/company/research/datasets/mvtec-3d-ad

本地数据集原始压缩包位置：D:\Downloads

eyecandies多模态异常检测数据集链接：

https://eyecan-ai.github.io/eyecandies/download

包含多个压缩包，其中已经下载CandyCane，本地数据集原始压缩包：D:\Downloads

**六、其他图像处理算法**

1）主要参考M3DM项目中的demo1~demo6文件，还需要优化以及方法上的改进

2）现有的线段提取算法和区域分割算法采用的是基于边缘特征的提取方法，对于粗边缘线条不合适，建议采用基于颜色阈值或深度图阈值或者基于点云分割阈值的方法。

3）现有的平整度检测使用的是open3d的points.segment\_plane函数，建议实验使用pyRANSAC-3D库的效果(test\_sdk)

参考链接：https://leomariga.github.io/pyRANSAC-3D/

4）可以采用最小二乘法计算表面平面度