

说明：

1、由于数据集的获取采用的是手机拍照，获取的照片分辨率是 3000X4000，直接用该分辨率进行训练的话，参数太多，因此用 pytorch 自带的 `transform.Resize()` 函数把图片分辨率降到 (224, 224) 大小在输入到网络中训练，用该函数处理后，宽度方向上是等比缩小，没失真，而高度方向上会被压缩，有失真。下图图 1 是用来测试的原图。



图 1

预测结果：

1、下图为采用原始网络的预测结果

图 2 和图 3 均为模型的预测结果，不同的是，图 2 在预测时模型输入输出的分辨率为 (224, 224)，而图 3 在预测时模型输入输出的分辨率为 (1568, 1568)。

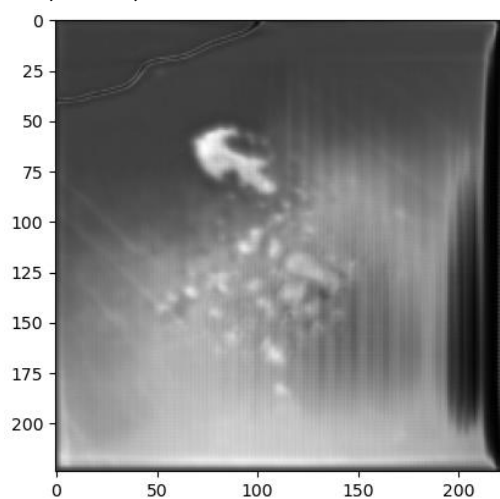


图 2

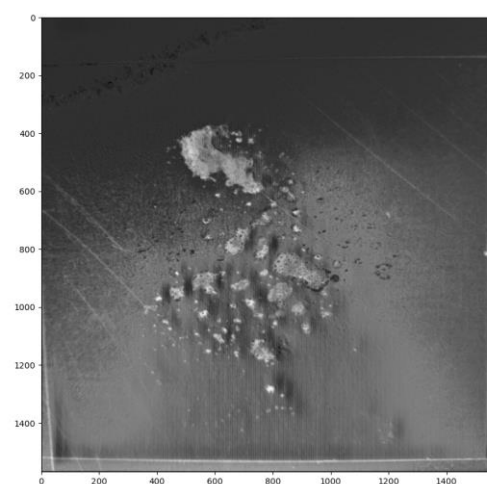


图 3

2、可能影响因素记录

- ①使用 `transform.Resize()` 函数导致的高度方向上的失真。
- ②拍摄照片时光照的影响。
- ③网络架构的影响。（不知道用 resnet 网络作为下采样的特征提取网络会不会好一些）。