测试说明书

1. 测试目标

本次软件测试的主要目标是检查程序是否存在严重的错误（bug），并确保程序能够正确执行各种功能，以达到预期结果。我们期望程序能够正确处理图像文件并将处理后的文件正确显示在指定区域。此外，程序应能够正确接收用户的输入，并在用户输入非法信息时给出相应的警告或提示。对于关键功能，还应确保在短时间内做出响应，以保证用户的良好体验。

本次软件测试最关键的测试目标是测试本程序中的车牌识别功能，其中包括对车牌识别功能在给定不同情况下图片的正确性进行测试，GUI测试，在用户没有选择图片时是否能给出正确提示的测试以及能否在1.5s内完成车牌识别任务的测试。除了对于车牌识别功能的测试，本次测试还涉及对于本程序中所有图像处理功能的测试，例如卷积、图像分割、图像模糊等。

1. 测试计划

采用单元测试的策略对本系统进行测试，本次测试采用全自动的形式进行，利用patch方法对用户输入进行模拟，使用QTest结合unittest编写测试用例，并使用XTestRunner生成测试报告。选取10张不同的图片作为测试资源，其中包括经过模糊处理的图片，车牌较小的图片，经过旋转后的图片，车牌位置较偏的图片，性能源汽车车牌图片，有多个车牌的图片，正常图片等等。

测试工作由许钰卓完成，对所有系统功能进行测试，对于特变简单的功能，只需要编写一个测试用例测试该功能是否能够正确完成目标。对于普通功能，需要编写至少三个测试用例，测试在不同情况下功能是否能够正确执行。对于重点功能（如车牌识别功能），需编写至少7个测试用例。测试时间为第15、16周，需要在第16周周三之前完成所有测试并生成测试报告。

具体的测试用例编写计划如下：

**Open\_image**:

1. 测试该功能能否正确打开.jpg文件并将图片正确显示在对应位置
2. 测试该功能能否正确打开.png文件并将图片正确显示在对应位置
3. 测试该功能能否正确打开.jpeg文件并将图片正确显示在对应位置
4. 测试该功能能否正确打开.webp文件并将图片正确显示在对应位置
5. 测试该功能能否正确打开中文路径下的文件
6. 测试该功能能否打开非图像文件，如果不能，是否给出提示
7. 测试该功能能否打开较大图像

**Save image：**

1. 测试该功能能否将图像保存为.jpg格式
2. 测试该功能能否将图像保存为.png格式
3. 测试该功能能否将图像保存为.jpeg格式
4. 测试该功能能否将图像保存在中文路径下
5. 测试该功能是否允许用户将保存为非图像文件，是否给出提示

**Vertical flip**：

1. 验证是否可以将图像进行垂直翻转，图像大小640\*480
2. 验证是否可以将图像进行垂直翻转，图像大小1600\*1200
3. 验证是否可以将图像进行垂直翻转，图像大小557\*373

**Horizontal flip**：

1. 验证是否可以将图像进行水平翻转，图像大小640\*480
2. 验证是否可以将图像进行水平翻转，图像大小1600\*1200

3）验证是否可以将图像进行水平翻转，图像大小557\*373

**Rotate image**：

1. 验证该功能是否可以将图像旋转正角度
2. 验证该功能是否可以将图像旋转负角度
3. 验证该功能是否可以将图像旋转浮点数角度

**Crop image**：

1. 验证该功能是否能正确裁剪图像，裁剪比例（100，100，200，200）
2. 验证该功能是否能正确裁剪图像，裁剪比例（10，20，30，40）
3. 验证该功能是否能正确裁剪图像，裁剪比例（10，10，20，20）

**Convert to gray**：

验证该功能是否能正确将图片转为灰度图像

**Apply gaussion blur**：

1. 检查再输入正确卷积核时能否运用该功能怎加图片的模糊程度
2. 检查在输入卷积核为浮点数和整数时能否弹出警告对话框

**Sharpen**：

1. 检查图像是否被正确锐化，图像大小640\*480
2. 检查图像是否被正确锐化，图像大小1600\*1200
3. 检查图像是否被正确锐化，图像大小557\*373

**Apply histogram equalization**：

1. 检查直方图均值化是否可以正常执行，图像大小640\*480
2. 检查直方图均值化是否可以正常执行，图像大小1600\*1200

3）检查直方图均值化是否可以正常执行，图像大小557\*373

**Apply gamma transform:**

1. 测试输入浮点数时gamma变化能否正确执行
2. 测试输入整数时gamma变化能否正确执行
3. 测试输入负数时gamma变化能否正确执行

**Conv**：

1. 测试输入整数时卷积操作能否正确执行
2. 测试输入浮点数时卷积操作能否正确执行
3. 测试输入负数时卷积操作能否正确执行
4. 测试在输入非法卷积核时能否正确弹出警告框

**Upload image**：

1. 测试在窗口2中能否正确使用该功能加载图片并保存在相应路径下。

**Detect image：**

1) This function can correctly identify the license plate number of different types of pictures.

2) The middle picture and the final result picture can be correctly displayed.

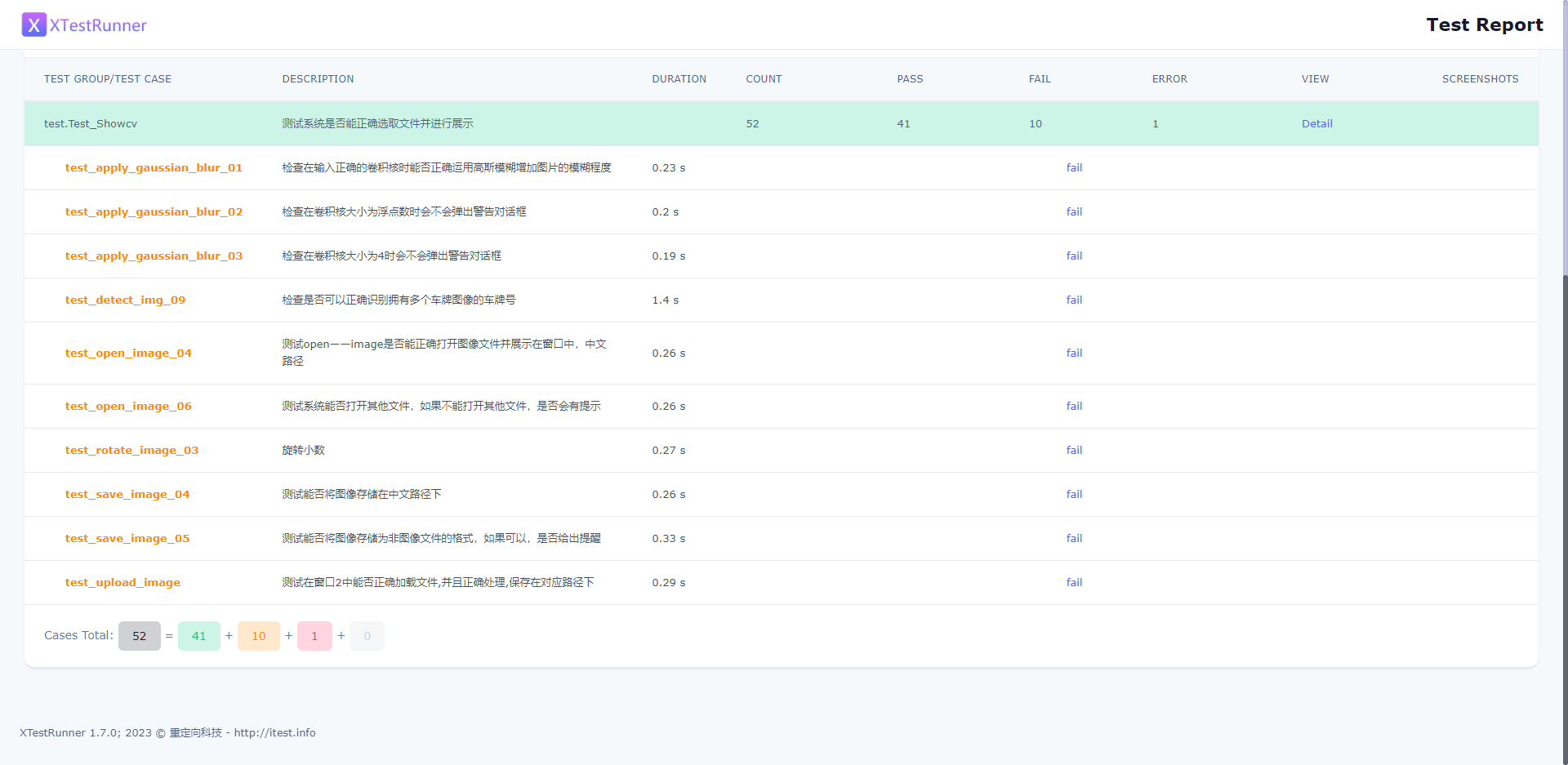
3) The identification time is within 1.1s.

4) When the user does not select the picture, will the prompt dialog box pop up

1. 测试执行

使用如下图片作为测试用例的图片资源：

在测试执行过程中，需要将所有测试用例使用unittest中的TestSuite方法包装成测试套件，并使用runner.run方法执行测试套件。在所有的测试用例中有10项测试没有通过，具体如下：

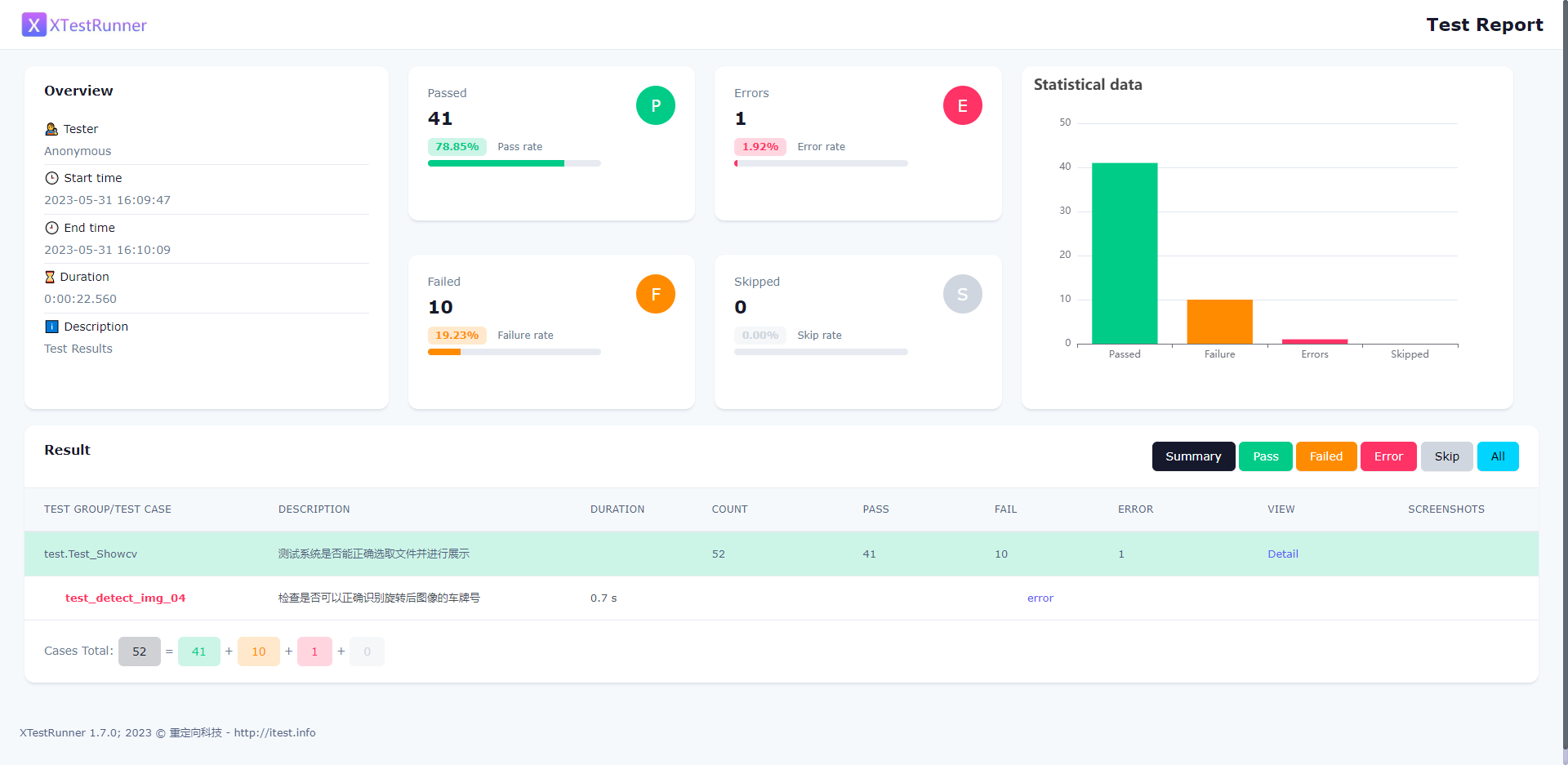


其中对于高斯模糊功能的测试出现了重大的bug，该功能没有正确的执行图像模糊功能，也没有跳出用户输入窗口。在所有打开和保存图片的功能中，用户都不可以选择中文路径，这会导致程序崩溃，具体原因在于Dialog中的getOpenFileName方法无法正确读取中文路径，用户需要将待打开图片保存在英文路径下才可以正确加载，但是该系统没有此提示。在旋转图片的功能中，用户无法请输入浮点数，在用户在键盘输入‘.’时，窗口不会读取，只会读取数字。此外，对于最重要的功能，车牌识别，我们采用了各种各样的图像来判断识别功能的正确性，其中，在有多个车牌同时存在的情况下，系统无法正确识别，但是会输出结果图像，给出输出内容。但是对于旋转后的图像，系统无法正确识别，也无法生成最终的结果图像，大概原因可能是在图像缩放的过程中，系统将具有车牌的部分剔除所导致的。



1. 测试报告

使用XTestRunner生成整个系统的单元测试报告，统计所有测试用例的通过率、失败率和错误率，并以.html文件的格式保存在指定文件夹下，具体的测试报告如以下图片所示：



在Overview中记录了测试人员和测试时间和相关描述，中间模块记录了本次测试的通过次数和通过率，错误次数和错误率以及失败次数和失败率等。右边的表格中展示了本次测试的通过次数、失败次数和错误次数的直方图。在result中存储了本次测试的全部测试用例并将其划分到pass、failed和error中。

本次测试的通过率为78.85%，有41个案例通过了测试，在上一节中已经相对具体的分析了导致其他用例没有通过或者错误的原因。总体来说，本次测试的大部分功能已经具有相对完备的功能，但是系统中还有一些重大的bug需要修改，也有一些小型的bug需要进行改进。