Software Engineering and Applications 软件工程与应用, 2022, 11(6), 1233-1240
Published Online December 2022 in Hans. http://www.hanspub.org/journal/sea
https://doi.org/10.12677/sea.2022.116125

Hans汉斯

基于图像匹配的GUI自动化测试技术研究

郝　琳，孔　婧，李美静，李　瑛*，李庆钰

北华航天工业学院，河北　廊坊

收稿日期：2022年10月11日；录用日期：2022年11月29日；发布日期：2022年12月8日

摘　要

当前软件更新迭代频率不断加快，但对软件的质量和稳定性要求却依然没有因此降低，测试工作就显得更加重要，而人与软件的交互大部分都是通过GUI来完成的，因此GUI测试是测试工作不可缺少的一环。传统测试工具存在一些典型问题，例如只适应于单一开发框架、断言方式简单、缺乏综合性的测试报告等。为解决上述问题，本文设计并实现了一种基于Airtest的GUI自动化测试框架，通过图像匹配技术来进行控件的定位，完成自动化测试的录制与回放，同时实现多种模式的测试断言以及测试报告的自动生成，提高了测试脚本的可重用性和测试效率。

关键词

GUI测试，自动化测试，图像匹配，Airtest，断言

Research on GUI Automated Test Technology Based on Image Matching

Lin Hao, Jing Kong, Meijing Li, Ying Li*, Qingyu Li

North China Institute of Aerospace Engineering, Langfang Hebei

Received: Oct. 11th, 2022; accepted: Nov. 29th, 2022; published: Dec. 8th, 2022

Abstract

At present, the update iteration frequency of software is accelerating, but the quality and stability requirements of software are still not reduced, the test work is more important, and most of the interaction between people and software is completed through the GUI, so GUI testing is an indispensable part of the test work. There are some typical problems with traditional testing tools, such as only adapting to a single development framework, simple assertion methods, and the lack of comprehensive test reports. In order to solve the above problems, a GUI automated test frame-

*通讯作者。

文章引用: 郝琳, 孔婧, 李美静, 李瑛, 李庆钰. 基于图像匹配的 GUI 自动化测试技术研究[J]. 软件工程与应用, 2022, 11(6): 1233-1240. DOI: 10.12677/sea.2022.116125

work based on Airtest is designed and implemented, which locates the control through image matching technology, completes the recording and playback of the automated test, and realizes the test assertion of various modes and the automatic generation of test reports, thus improving the reusability of the test script and test efficiency.

Keywords

GUI Testing, Automated Testing, Image Matching, Airtest, Assertion

1. 引言

GUI (Graphical User Interface)，即图形用户界面，它允许用户通过图像元素与计算机、笔记本、智能手机和平板电脑等电子设备进行交互[1]。GUI 的广泛应用，极大地方便了用户操作，同时也给测试人员带来了相应的挑战。通常软件中，GUI 相关的代码量在整个项目中占了非常大的比重，同时 GUI 的质量直接影响着整个系统的运行。因此，GUI 测试在确保软件质量方面起着至关重要的作用。

GUI 测试经历了从传统的手工测试阶段到录制回放的自动化测试阶段，由于 GUI 自动化测试的测试效率高且测试成本低，目前已成为各大公司的首选。GUI 自动化测试最关键的点在于如何识别控件对象，主要方法有基于关键字驱动的方法、基于 UI 控件搜索的方法、基于图像匹配的方法，相对应的较为成熟的 GUI 自动化测试工具有 Mercury Interactive 公司的 QTP，Microsoft 的 Coded UI、WinAppDriver 以及 MIT 的 SikuliX 等。现有的这些测试工具存在着对不同 GUI 开发框架的支持程度不同，测试断言少，且生成的测试报告需人工汇总等问题[2] [3] [4]。

针对上述问题，本文提出一种基于 Airtest 的 GUI 自动化测试框架。该框架使用图像匹配技术来识别控件对象，以此实现对不同框架下开发的应用程序的测试支持，并提出几种新的断言以支持判断文件修改正确与否和页面预期结果正确与否，同时在测试报告的生成方面进行优化，实现测试执行完成后无需人为干预自动输出汇总报告。

2. 相关工作

2.1. 基于关键字驱动的 GUI 自动化测试

基于关键字驱动的 GUI 自动化测试，也被称为表格驱动测试或基于动作字的 GUI 自动化测试。它由 Test Step、Test Object、Action、Test Data 四部分组成，其主要思想为将关键字从测试用例和测试步骤中分离出来存放到对应的 Test Step、Test Object、Action、Test Data 中，测试发生变更时只需修改关键字即可，实现了测试数据与测试脚本的分离。

基于关键字驱动的测试工具中典型的代表是 QTP。QTP 将每个控件(如窗口、按钮、文本框等)都视为一个对象，同时构建一个对象库，把每个对象相应的属性和方法一并存入，脚本录制时将每一步操作的操作对象、操作类型和输入数据提取出来并进行记录，测试人员根据需要对其进行修改，增加检查点；脚本回放时按照记录文件中的内容依次调用相应对象的属性和方法进行录制还原，同时根据设置的检查点判断测试是否通过。

QTP 的应用大幅提高了测试脚本的可重用性，同时使得编程能力不强的测试人员加快了测试进度。但 QTP 仍存在一些缺点：某些控件对象不易识别、不开源免费、只为每个测试用例单独生成测试结论，无法生成整体的测试结论。同时 QTP 只能识别 Windows 标准的控件(按钮，滚动条，静态控件，列表框，编辑框，组合框)，不能识别自定义控件[5] [6] [7]。

2.2. 基于 UI 控件搜索的 GUI 自动化测试

基于 UI 控件搜索的 GUI 自动化测试[8]，通过给定的控件属性(如 id、name、xpath 等)来进行控件元素的搜索定位，获得要进行操作的控件元素的相关信息，对控件进行操作，与预期结果进行比较，判断是否一致，从而完成测试。通过该方式来完成测试的工具，Web 端是 Selenium，移动 App 端是 Appium，而 Windows 桌面端上实现较好的是 AutoIt [9]。

AutoIt 通过内置的 Finder Tool 获取被测应用程序中的控件，然后通过模拟键盘敲击、鼠标移动等操作来完成整个测试过程。它拥有类 BASIC 的语言表达式，同时支持多种语言以及多种 Windows 版本，使得新旧测试人员都可以很快上手。同时支持 Win32 和第三方 DLL API 的调用，脚本还可以编译成 exe 文件单独运行，在跨设备的测试工作中表现良好。

虽然 AutoIt 很强大，但它也存在一些缺点：对于不同桌面浏览器可能存在一些兼容性问题，没有断言，没有测试报告，每次测试通过与否需凭人工判断，不能做到无人值守。

基于关键字驱动和基于 UI 控件搜索的测试方法，在定位元素方面出现一点误差就可能导致控件对象定位不到，因此提出了基于图像匹配的 UI 自动化测试方法。

3. 基于图像匹配的 GUI 自动化测试

3.1. 图像匹配在 GUI 自动化测试的可行性分析及优势

基于图像匹配的 GUI 自动化测试，通过寻找当前界面中与给定图片相同或相似的图像区域，返回该图像区域的中心点坐标，再根据该坐标对控件操作从而完成测试。屏幕大小有限，屏幕上显示的内容也是有限的，这使得图像匹配的复杂度控制在了一定范围内[10]。

由于不同 GUI 开发框架的底层架构不同，使用不同框架所开发的功能相同的应用程序，控件结构属性也相应会存在差异。这样就导致关键字驱动和 UI 控件搜索的测试方法需要为不同框架下开发的应用分别编写测试脚本来识别控件，造成了测试成本高昂。而基于图像匹配的测试方法由于应用的控件图像不会因开发框架的变化而发生太大变化，所以测试时无需对脚本进行大量改动，极大地降低了测试成本，更符合生产需要。

3.2. 图像匹配算法

在常用于自动化测试方法的诸多算法中，主要有以下两个分支：基于模板匹配的算法和基于特征点匹配的算法。

基于模板匹配的算法，即实现在当前屏幕图像上绘制一个与给定模板图像相同大小的矩形框，将该矩形框内的图像与给定的模板图像进行像素点级别的对比，如果该图像与给定模板图像的相似程度很高(即两张图片中相同的像素点数与总像素点之商大于给定的阈值)，就认为找到了该图像，同时返回该图像在当前屏幕上的坐标，从而可以对该坐标上的控件进行操作，完成自动化测试；否则该框向前移动继续比较，直到找到相似度极高的区域或当前屏幕图像遍历完成。

基于特征点匹配的算法，先对模板图像进行特征提取，得到一定数量的特征点及描述子，再对当前屏幕图像进行特征提取，得到当前屏幕图像的一系列特征点以及特征点描述子，计算模板图像的特征描

述子与屏幕图像的特征描述子之间的距离，若两个特征点描述子之间的向量距离(常见的距离有欧式距离、汉明距离等)最小，则将这两个描述子视为它们所对应特征的最佳匹配。根据所有最佳匹配特征点对的横纵坐标，计算出模板图像转化到屏幕图像上的中心点、左上点、右下点坐标，取出屏幕图像中的该部分图像，与模板图像进行相似度计算，若相似度大于阈值返回中心点坐标，否则认为当前屏幕图像上不存在与模板图像相似的图像区域。

基于模板匹配的算法相对简单，运行速度快，但它不具有旋转不变性和尺度不变性，即当待匹配图像发生旋转或者缩放时，该算法就可能无法找到正确的匹配图像对。因此基于模板匹配的算法常用于对固定设备上的静态元素编写测试脚本的场景；基于特征点匹配的算法相对复杂，但它提取到的特征在旋转不变性和尺度不变性方面有着良好的表现，当待匹配图像发生旋转或者缩放时，该算法对正确匹配和错误匹配的图像对仍具有一定的辨别能力。真实生产环境中常常需要对软件在不同环境下的兼容性和稳定性进行测试，从而测试脚本所运行的设备可能会发生更改，设备发生更改时待匹配图像也可能会发生变化，所以基于特征点匹配的算法更适用于实际测试活动中。

4. 技术实现

4.1. 系统框架

本自动化测试框架以 Airtest 为基础框架并遵循维护性强和可复用性强的原则进行设计，同时集成 Pytest 测试框架和 Allure 测试报告框架对 Airtest 进行了扩展。采用图像匹配技术实现控件对象的识别，使得控件识别不受开发框架的限制，新增的断言模式和测试报告使得测试人员对测试工作更加得心应手，从而提高测试效率。框架采用分层设计，分为四层：公共库层、控件识别层、测试管理层、表现层。

图 1 是本测试框架的系统架构图。

Figure 1. Diagram of the system architecture
图 1. 系统架构图

第一层是公共库层，该层主要实现一些通用功能，包含常用方法封装、基础配置、文件解析、截图等功能；第二层是控件识别层，该层包含了常用的图像匹配算法，为测试脚本执行过程中的控件对象识别提供支持；第三层是测试管理层，该层根据测试项目与功能点一对多的需求构建了脚本树用以管理脚本，同时使用 Pytest 来实现多个测试脚本的执行，并新增了测试断言以及相应的异常捕获；第四层是表现层，该层主要展示测试执行结果，测试人员可以通过测试报告与用例截图对照着查看每个测试脚本的

执行细节，并根据测试日志定位失败用例中存在的问题。

4.2. 测试脚本自动运行

基于特征点匹配的图像匹配算法种类有很多，如 SIFT、ORB、BRISK、AKAZE 等。为了挑选出效果更好、更适合用于测试环境中的匹配算法，本文从一些应用程序中截取了部分控件图像，并对 SIFT、ORB、BRISK、AKAZE 等算法的特征点检测能力进行了对比，从图 2 (左上右下依次为 SIFT、ORB、BRISK、AKAZE 算法检测到的特征点示意图)中可以得出，BRISK 算法检测出的特征点基本覆盖了所有控件且误检概率较低；AKAZE 和 SIFT 均检测到了大量特征点，但 SIFT 检测到的特征点中有不少误检，而 AKAZE 对部分控件的特征点检测有所遗漏；ORB 检测到的特征点数量远少于其他算法且对勾选框的检测效果很差，所以，本框架选择 BRISK 作为图像匹配的主算法，并以 SIFT 算法为辅，当使用 BRISK 算法未在待匹配图像中找到与模板图像相似的图像块时，再改用 SIFT 算法继续执行相同操作。

基于图像匹配算法的控件识别方法，在面对不同开发框架所开发软件进行测试时，有很强的适应能力，我们对其进行了验证，如图 3 所示，左为对 Tkinter 所编写的软件进行测试时的截图，右为对 Qt 所编写的软件进行测试时的截图，图中红圈指代测试过程中鼠标的光标所在位置。

Figure 2. Comparison chart of feature point detection
图 2. 特征点检测对比图

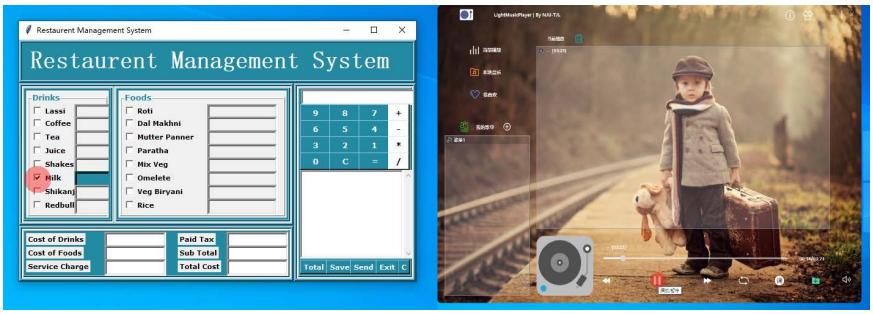

Figure 3. Testing software developed in different frameworks
图 3. 对不同框架所开发软件进行测试

4.3. 测试断言模式

断言用来验证应用程序执行的正确性，快速定位错误。本测试框架除了支持目标图像是否存在于当前页面的断言、目标图像与模板图像是否一致的断言等这些常规的断言以外，还根据文件操作需求以及判别页面正确需求，新增了以下几种测试断言：目标文件是否存在、目标文件的最后 n 行是否为预期文字、OCR (Optical Character Recognition，光学字符识别)结果与预期是否一致。

支持目标文件是否存在的断言。一些测试场景下(如对下载文件、删除文件和导出文件等功能进行测试)经常会生成或修改一些文件，此时文件存在与否关系着测试用例的通过与否。考虑到有些软件生成的文件名字会很长，因此本框架还加入了断言文件是否存在的模糊选项，测试断言输入的文件名是给定路径下某个文件的文件名一部分即可视为通过。

支持目标文件的最后 n 行是否为预期文字的断言。许多软件都有日志功能，测试时可以通过其日志来判断测试是否通过。具体实现：先判断文件字符数是否小于读取字符数 m，若小于，从文件开头读取所有字符，否则从文件末尾读取 m 个字符；随后判断读取到的文件行数是否大于 n，若不大于 n 则判断读取到的文件行数是否等于 n 且文件读取指针移动到了文件起始位置，两个条件满足其一，便可将读取到的文件内容与预期结果进行比较从而完成断言，若两个条件均不满足则将读取字符数 m 翻倍，重复执行上述操作直至条件满足。流程图如图 4 所示。

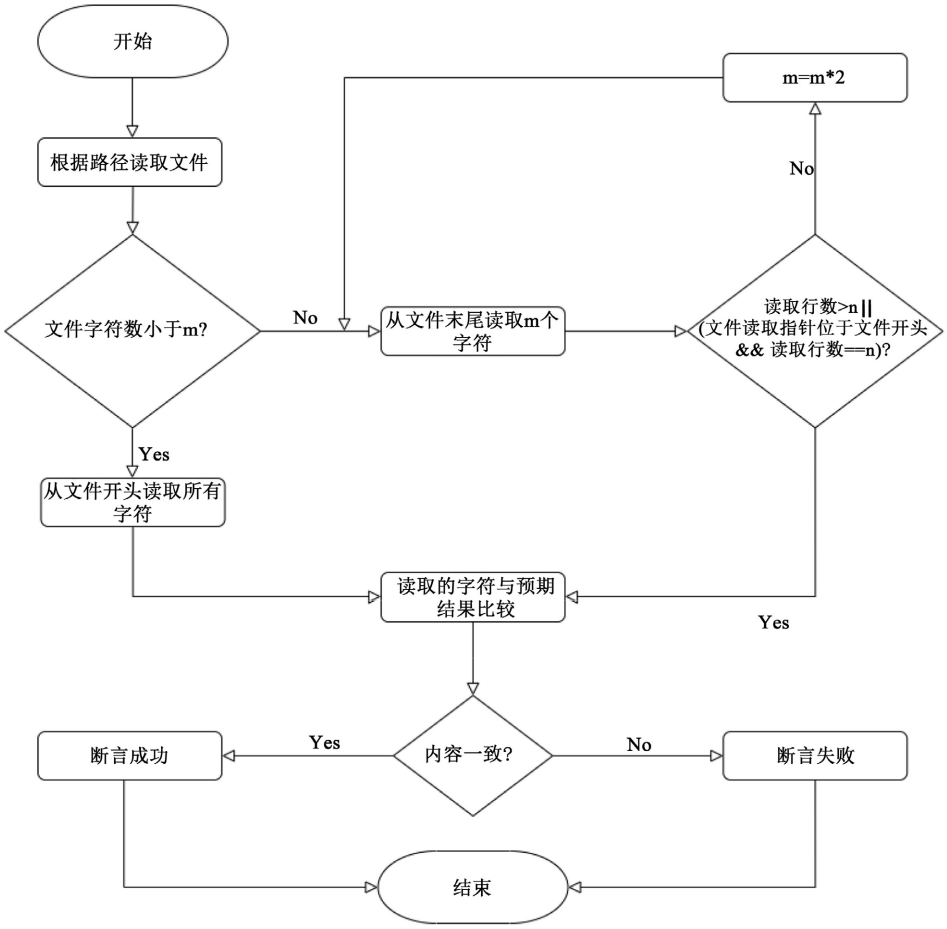

Figure 4. Program flow chart
图 4. 程序流程图

支持 OCR 结果与预期是否一致的断言。一些软件操作过程中会在界面中显示大量的文字，用 OCR 便可判断预期结果与实际结果是否一致。由于 OCR 识别过程中可能会存在部分文字识别错误，因此本断言支持以实际结果与预期结果之间的编辑距离来决定断言是否通过。同时考虑到大量文字中重要的信息只有小部分，本断言还支持了判断预期结果是否包含于实际结果之中。图 5 为 OCR 断言失败时测试系统的输出。

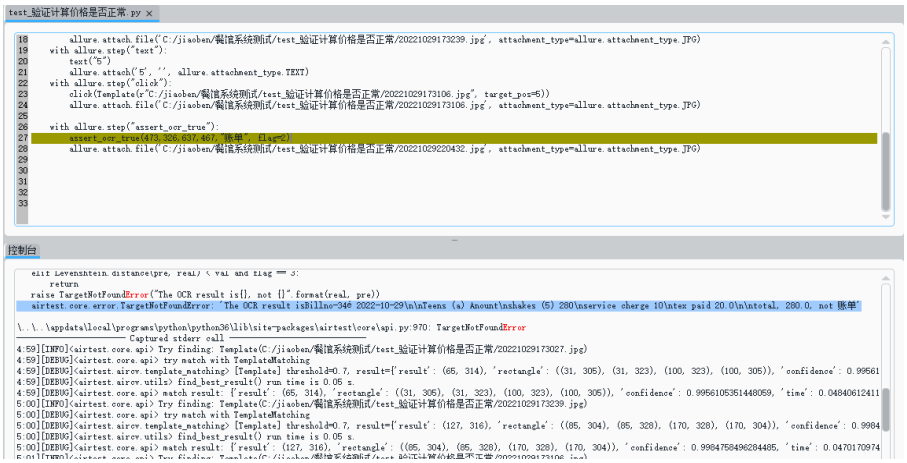

Figure 5. Diagram of assertion failure
图 5. 断言失败图

4.4. 测试报告

Allure 是一种灵活的轻量级多语言测试报告工具，能以简洁的报告形式显示已执行测试的结果，而且还允许参与开发过程的人员从执行过程中提取有用信息[11]。本框架以 Allure 为主，脚本树为辅，来完成测试报告的生成工作。待测试完成后，可以先从脚本树中根据脚本颜色来直观地查看各脚本的通过情况，再根据失败用例的名字从 Allure 生成的详细报告中查看执行失败的原因，判断是否为 bug 所致，再决定下一步是提交问题还是修改脚本。

图 6 为测试执行后的展示，左为脚本树情况，右为生成的测试报告。

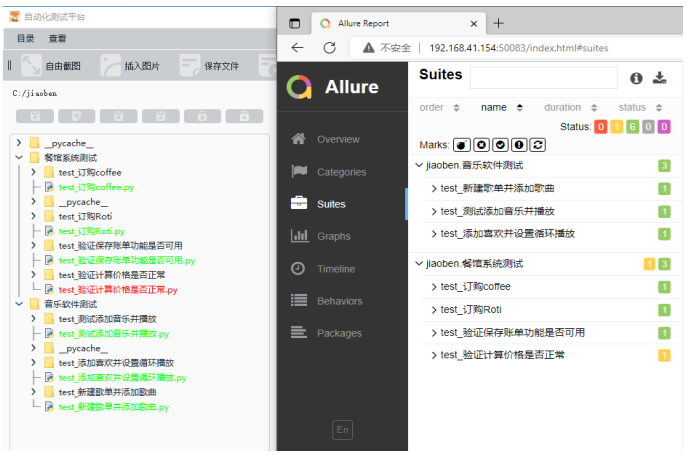

Figure 6. Diagram of test report
图 6. 测试报告图

5. 结束语

本文提出了一种基于 Airtest 的 GUI 自动化测试框架，解决了测试工具对不同 GUI 开发框架的兼容性问题，支持更多种类的断言模式，并且可以自动生成汇总后的测试报告，一定程度上提高了测试人员的测试效率。但本框架还存在一些问题，如跨分辨率匹配不稳定，而深度学习图像匹配技术相较于传统的图像匹配技术有着更高的鲁棒性和实时性[12] [13] [14] [15] [16]，因此下一步将尝试在自动化测试框架中使用一些深度学习的技术，提高自动化测试的准确率和效率。

基金项目

校级研究生创新资助项目(项目编号：YKY-2021-21)；廊坊市科技支撑计划项目(项目编号：2021011029)。

参考文献

[1] Indeed Editorial Team (2020) What Is a GUI (Graphical User Interface)? Definition, Elements and Benefits. https://www.indeed.com/career-advice/career-development/gui-meaning

[2] 邱长云. 基于 WPF 的 UI 自动化测试系统[J]. 电脑知识与技术, 2020, 16(11): 49-50.

[3] 朱方祥. Windows 应用程序自动化测试关键技术设计与实现[D]: [硕士学位论文]. 合肥: 中国科学技术大学, 2018.

[4] 余锦润, 杨丹君, 李波波. 基于位图识别的 UI 自动化测试研究和应用[J]. 自动化与仪表, 2021, 36(3): 90-94.

[5] 王健, 李亚伟, 朱璇. 使用 QTP 对 Silverlight 应用进行自动化测试的研究与实践[J]. 软件, 2014(4): 18-20.

[6] 钱汉伟. GAT:Windows 平台下 GUI 软件自动化测试框架研究[J]. 软件, 2018, 39(3): 72-76.

[7] 吴琼. 基于 QTP 自动化测试框架的研究与应用[D]: [硕士学位论文]. 合肥: 中国科学院大学, 2015.

[8] 侯津, 顾乃杰, 丁世举, 杜云开. 基于控件路径的跨设备 UI 自动化测试方法[J]. 计算机系统应用, 2018, 27(10): 240-247.

[9] Denisov, E.Y., Voloboy, A.G., Biryukov, E.D., *et al.* (2021) Automated Software Testing Technologies for Realistic Computer Graphics. *Programming and Computer Software*, **47**, 76-87. https://doi.org/10.1134/S0361768820080034

[10] 李昕宇, 侯春萍, 王宝亮, 等. 基于图像匹配的移动应用自动化测试方法研究[J]. 计算机工程与应用, 2016, 52(13): 43-47.

[11] 彭新宇. 基于 Selenium 的 Web 自动化测试框架研究与实现[D]: [硕士学位论文]. 廊坊: 北华航天工业学院, 2021.

[12] 魏泓安, 单小军, 郑柯, 等. 基于深度学习的 SAR 与光学影像配准方法综述[J]. 无线电工程, 2021, 51(12): 1361-1372.

[13] 李昌昊. 基于深度学习的图像配准方法研究[D]: [硕士学位论文]. 南昌: 南昌大学, 2021.

[14] 刘凯宇. 面向图像匹配的神经网络优化研究[D]: [硕士学位论文]. 哈尔滨: 哈尔滨工业大学, 2020.

[15] 哈尔滨工业大学(深圳). 一种结合深度学习的图像匹配方法[P]. 中国, CN202110540653.9. 2021-08-10.

[16] 西安理工大学. 基于深度学习的相似图像匹配方法[P]. 中国, CN202010131479.8. 2020-06-30.