**项目任务书**

**目标软件系统：PDFPatcher**

**（PDF补丁丁）**

**团队成员：曾也桐**

**詹语欣**

**蔡沛霖**

**熊锦程**

**日期：2023年3月30日**

**目录**

**1 目标软件系统**3

**2 基本情况介绍**3

2.1 PDF领域3

2.2 PDFPatcher采用的PDF相关技术5

2.3 PDFPatcher解决的问题12

2.4 目标14

2.5 影响14

2.6 背景分析14

**3 内容**18

3.1 问题解决方案18

3.2 功能21

3.3 应用案例和具体场景23

**4 特征和影响**25

4.1 运行环境25

4.2 用户25

4.3 功能运行25

**5 验收建议**31

5.1 优化方向31

5.2 验收列表32

**1 目标软件系统**

PDF补丁丁——一款开源的PDF阅读、编辑软件，类似于PDF工具箱，可以编辑书签、剪裁旋转页面、解除限制、提取或合并文档，探查文档结构，提取图片、转成图片等等。

PDF补丁丁软件系统由C#、C++、C、html语言集成开发。软件架构可分为App 包（PDF 补丁丁主程序）、doc包（放置程序的使用文档）、JBig2包（放置 JBIG2 图像的编码和解码库代码）三个部分。

**2 基本情况介绍**

**2.1 PDF领域**

**2.1.1 PDF**

由于PDF补丁丁是PDF领域的开源应用软件，且PDF补丁丁处理的对象就是PDF文档。因此，需要对PDF进行调研和分析。

PDF，全称为“Portable Document Format”，是一种主流的电子文档格式。其具有许多良好的特性，比如：

①跨平台性：PDF文档在不同的操作系统和软件之间都可以保持一致的呈现效果，使得文档的可读性和可靠性更高。

②安全性：PDF文档可以设置密码、数字签名等安全措施，保护文档内容不被篡改或盗用。

③良好的排版效果：PDF文档能够保持原始文档的排版效果，包括文字字体、颜色、图片、图表等。

④可搜索性：PDF文档可以进行全文检索，提高文档检索的效率。

⑤支持多种媒体：PDF文档可以包含多种媒体形式，例如音频、视频等。

因功能丰富、性能良好，PDF文件被广泛应用于电子书籍、合同、报告、学术论文、演示文稿等领域。

**2.1.2 PDF处理软件**

PDF处理是指对PDF文档进行编辑、转换、合并、加密、解密等操作的过程。

PDF处理行业的发展可以追溯到20世纪90年代初，当时Adobe公司推出了PDF格式，并发布了第一款PDF编辑软件Acrobat。随着PDF格式的普及和应用场景的不断扩大，PDF处理行业也逐渐壮大。目前，PDF处理软件已经成为了各行各业必不可少的工具。

PDF处理行业的主要竞争者包括Adobe、Foxit、Nitro、Nuance等知名公司，以及一些小型的创业公司。这些公司通过不断推出新的功能和技术来提高产品的竞争力，满足用户的需求。

这些软件虽然与PDF补丁丁属于同类型软件系统，但究其功能着重点、开发技术、特点来说，都与PDF补丁丁（本次分析的开源软件）有所不同。

这些软件包括但不限于：

①Adobe Acrobat DC：一款由Adobe公司开发的付费PDF阅读软件。它支持在PDF文档中进行注释、编辑文字、旋转和缩放页面等基本的编辑和查看功能。采用了Adobe的PDF技术。

②WPS：涵盖对多种文件的操作，也包括面向PDF的阅读器和编辑器，拥有阅读PDF，修改PDF页面的文字、图像、表格和注释，对PDF文档进行加密、解密和签名，批量转换和合并PDF文件等主要功能。WPS的PDF采用先进的PDF解析和渲染技术，支持高效且精准地显示PDF文档的各项内容。其OCR技术基于深度学习和人工智能技术，能够快速准确地识别多种语言的文字，可识别扫描或图片格式的PDF文件。WPS的PDF基础功能是免费的，但一些高级功能（如PDF文档密码保护和OCR技术）需要付费解锁。

③Foxit Reader（福昕高级PDF编辑器）：由Foxit Corporation开发的PDF阅读、编辑和创建软件，支持添加注释、高亮文本、填写表单、转换文件为PDF格式等功能。其特点是速度快，体积小，界面简洁。Foxit Reader有免费试用版也有企业版，企业版提供更多增值功能，但需要付费。

④知网研学（原E-study）：简单好用，有网页端、PC端和单机端三个版本，集成了文献管理、文件整理、云同步等超额的功能。其中，PC端的功能最为丰富，支持抓图识别文字。亮点：云端同步、抓图识别文字。缺点：付费，阅读文献、文章可以赚取积分抵扣支付费用。

总体而言，目前PDF处理和文件处理软件的功能已经非常丰富，可以实现多种操作，例如：

- 创建、编辑和转换PDF文档；

- 添加、删除、合并和拆分PDF页面；

- 对PDF文档进行加密、签名和批注；

- 将PDF文档转换为其他格式，例如Word、Excel、HTML和图片等；

- 对PDF文档进行OCR识别、压缩和优化等。

Adobe最近推出的主要新技术为：

1. 自动OCR：Adobe Acrobat DC可以通过自动OCR（光学字符识别）技术将扫描的PDF文档转换为可编辑的文本，大大提高了编辑和搜索的效率。
2. PDF表格自动填充：Adobe Acrobat DC可以自动识别PDF表格中的字段，并自动填充相应的数据，减少了手动输入的工作量。
3. 云端存储：Adobe Acrobat DC可以将PDF文档存储到云端，方便用户随时随地访问和共享。
4. 数字签名：Adobe Acrobat DC支持数字签名，可以确保文档的安全性和完整性，防止篡改和伪造。

这些新功能使Adobe Acrobat DC在PDF处理领域具有很强的竞争力。自动OCR和PDF表格自动填充功能提高了编辑和搜索的效率，云端存储功能方便用户随时随地访问和共享，数字签名功能保证了文档的安全性和完整性。这些功能都是用户普遍需要的，因此Adobe Acrobat DC在市场上具有很高的竞争力。

现有的PDF处理和文件处理软件的技术也在不断发展，大致可以归纳为以下三类：

- 基于云计算的PDF处理和文件处理服务，可以实现在线协作、自动化处理和无缝集成等；

- 基于人工智能的PDF处理和文件处理技术，可以实现自动识别、自动分类和自动提取等；

- 基于移动设备的PDF处理和文件处理应用程序，可以实现随时随地的操作和分享。

**2.2 PDFPatcher采用的PDF相关技术**

一般而言，PDF相关软件的开发都参照了PDF文档的规范（ISO 32000-1:2008 《Document management — Portable document format — Part 1:PDF 1.7》），它是 PDF 处理程序开发者的必读文献。PDF补丁丁的开发也不例外。

PDF 文档格式中，涉及印刷领域的多项技术，并有其独特的文档结构，还使用了多种数据压缩算法。正是因为，要从零开始编写 PDF 文档的处理程序通常过于困难且不切实际，PDF 补丁丁采用了 .NET Framework 框架进行开发，主要使用 iText 和 MuPDF 这两个开放源代码的组件库来处理 PDF 文档。

前者是 .NET 组件，与PDF主程序具有较好的互操作性，并且在解析、生成和修改PDF文档，以及嵌入TTF字体子集这些功能上，优胜于后者。

后者采用 C 语言开发并编译，与前者相比，其最大的优点是具有渲染 PDF 文档为位图的功能。MuPDF 编译出来的动态组件库可在作者另一个开放源代码库 [SharpMuPDF](https://github.com/wmjordan/SharpMuPDF) 下载。PDF 补丁丁通过 P/Invoke 技术调用该组件库的功能。

除了 PDF 开源组件之外，程序还使用了其它优秀开源组件。例如 ObjectListView 这个强大的列表控件、FreeImage 来读取和解码各种类型的点阵图像文件、Cyotek 的 ImageBox 用于显示渲染好的 PDF 文档页面、TabControlExtra 用于构建选项卡式文档界面、HTMLRenderer 用于显示 HTML 网页界面等等。

PDFPatcher在实现对PDF文件的编辑时，使用了多种技术和原理，包括PDF文件格式、PDF对象、PDF字体、PDF流、PDF渲染和PDF编辑器API等。同时，PDF编辑器采用树形结构模型，使用面向对象编程、事件驱动编程和函数式编程等编程思想，实现对PDF文件的操作。

1. PDF解析技术

PDF解析技术主要是通过解析PDF文件的二进制数据，获取其中的元数据、书签、页面内容等信息。PDF文件的结构非常复杂，可以通过调用PDF解析库或者自己编写解析器对PDF文件进行解析，然后将解析结果保存在内存中，供其他程序使用。PDF解析技术的方法包括：

- 使用PDF解析库：比如iText、PDFBox等PDF解析库，可以直接调用库中的API对PDF文件进行解析。

- 自己编写PDF解析器：通过读取PDF文件的二进制数据，解析文件结构，获取其中的元数据、书签、页面内容等信息。

PDF解析需要用到以下几种思想和模型：

（1）自动机模型：PDF 文件是由多个对象组成的，每个对象都有自己的类型和属性，这些对象之间的关系可以使用自动机模型来描述。自动机模型可以根据对象的类型和属性来进行状态转移，从而实现 PDF 文件的解析和处理。

自动机模型是一种描述有限状态机（Finite State Machine，FSM）的模型。在 PDF 文件解析中，可以使用自动机模型来描述 PDF 文件中各个对象之间的关系和状态转移规则，从而实现文件的解析和处理。

自动机模型由以下几个要素组成：

①状态集合：自动机模型中的状态表示对象的状态，状态集合是所有可能状态的集合。

②转移函数：转移函数表示状态之间的转移规则，即从一个状态转移到另一个状态所需要满足的条件。

③初始状态：初始状态是自动机模型开始执行时的状态。

④终止状态：终止状态是自动机模型执行结束时的状态。

自动机模型的工作原理如下：

①状态转移：自动机模型根据当前状态和输入，通过转移函数计算出下一个状态。状态转移可以是确定性的（Deterministic Finite Automaton，DFA）或非确定性的（Nondeterministic Finite Automaton，NFA）。

②输入处理：自动机模型根据输入，执行相应的操作。在 PDF 文件解析中，输入可以是文件中的字节流或者对象的属性值。

③状态判断：自动机模型根据当前状态和输入，判断是否满足转移条件。如果满足条件，则进行状态转移；否则，保持当前状态不变。

④终止条件：自动机模型在执行过程中，如果达到了终止状态，则停止执行。

自动机模型可以使用状态转移图（State Transition Diagram）来表示。状态转移图是一个有向图，其中节点表示状态，边表示状态之间的转移，标签表示转移条件。

在 PDF 文件解析中，可以使用自动机模型来描述 PDF 文件的结构和语法规则，从而实现文件的解析和处理。例如，可以使用自动机模型来解析 PDF 文件的头部、交叉引用表、对象等部分，从而提取出所需的数据。

（2）递归下降解析器：PDF 文件的结构比较复杂，需要用到递归下降解析器来实现文件解析。递归下降解析器是一种基于语法规则的解析器，可以通过递归地调用自身来解析复杂的语法结构。

递归下降解析器是一种常用的语法分析方法，它可以将一个文法转换为程序代码，从而实现对输入文本的解析和处理。在PDF文件处理中，递归下降解析器通常被用于解析PDF文件的语法结构和内容，以便进行相关的操作和转换。

递归下降解析器的原理是基于自顶向下的递归分析方法，它从文法的起始符号开始，逐步向下推导出文本的各个部分，并将其转换为程序代码。在这个过程中，解析器会根据文法规则和输入文本的结构，不断地调用自身或其他子程序，以便完成语法分析和处理。

递归下降解析器的思想是将文法规则转换为程序代码，从而实现对输入文本的解析和处理。这种思想是基于面向对象编程和函数式编程的思想，它将文法规则看作对象或函数，将输入文本看作参数或输入，通过递归调用和函数调用来实现文本的解析和处理。

递归下降解析器的模型是一个树形结构，它将文本的各个部分看作树的节点，将文法规则看作树的分支，通过递归调用和函数调用来实现树的遍历和节点的处理。在PDF文件处理中，递归下降解析器通常将PDF文件的语法结构和内容转换为树形结构，以便进行相关的操作和转换。

递归下降解析器的具体应用过程包括以下几个步骤：

①定义文法规则：根据PDF文件的语法规范，定义相应的文法规则，包括PDF文件头、对象、字典、数组、流等。

②实现解析器：根据文法规则，实现相应的解析器，包括PDF文件头解析器、对象解析器、字典解析器、数组解析器、流解析器等。

③解析PDF文件：将PDF文件读入解析器中，调用相应的解析器进行解析，将PDF文件转换为树形结构。

④处理PDF文件：根据树形结构，进行相关的操作和转换，例如提取文本、转换成图片、合并PDF文件等。

⑤输出PDF文件：将处理后的PDF文件输出到指定的位置，完成PDF文件处理过程。

（3）二进制解析器：PDF 文件是一种二进制文件，需要使用二进制解析器来读取和处理其中的数据。二进制解析器可以将文件中的二进制数据转换为可读的文本格式，从而实现文件的解析和处理。

二进制解析器是一种常见的解析二进制数据的方法。在PDF文件处理中，二进制解析器通常被用于解析PDF文件中的二进制数据，如图像、字体、流等。

二进制解析器的原理是将二进制数据转换为程序代码，从而实现对输入数据的解析和处理。二进制解析器通常采用位运算、字节序等方法对二进制数据进行处理，以实现对二进制数据的解析和转换。

二进制解析器的思想是将二进制数据看作程序代码中的数据结构，通过位运算、字节序等方法对其进行处理，从而实现对二进制数据的解析和转换。

二进制解析器的模型通常采用树形结构，将二进制数据看作树的节点，将二进制数据的结构和类型看作树的分支，通过遍历树的节点和调用相应的函数来实现对二进制数据的解析和转换。

二进制解析器的具体应用过程包括以下几个步骤：

①定义二进制数据结构：根据PDF文件格式规范，定义相应的二进制数据结构，如字体文件、图像文件、流等。

②实现解析器：根据二进制数据结构，实现相应的解析器，包括字体解析器、图像解析器、流解析器等。

③解析二进制数据：将二进制数据读入解析器中，调用相应的解析器进行解析，将二进制数据转换为程序代码中的数据结构。

④处理二进制数据：根据程序代码中的数据结构，进行相关的操作和转换，例如将字体转换为PDF格式、将图像提取为PNG格式等。

⑤输出结果数据：将处理后的数据输出到指定的位置，完成二进制数据处理过程。

2. XML文件处理技术

XML文件处理技术主要是通过将PDF文件的元数据、书签、页面设置等信息导出成可编辑的XML文件，并能将XML文件导入到PDF文件中。XML文件是一种标记语言，可以很方便地描述数据结构和数据内容，通过调用XML解析库或者自己编写解析器对XML文件进行解析，然后将解析结果保存在内存中，供其他程序使用。XML文件处理技术的方法包括：

- 使用XML解析库：比如DOM、SAX、JDOM等XML解析库，可以直接调用库中的API对XML文件进行解析。

- 自己编写XML解析器：通过读取XML文件的文本数据，解析文件结构，获取其中的元数据、书签、页面设置等信息。

3. PDF文件处理技术

PDF文件处理技术主要是能够对PDF文件进行修改、添加、删除等操作，包括修改文档属性、添加、删除、修改书签、替换字体、添加链接、更改逻辑页码编号、调整页面尺寸、裁剪页面、调整页面旋转方向、删除XML元数据、删除自动执行动作等。PDF文件处理技术的编程思想是通过调用PDF处理库或者自己编写处理器对PDF文件进行处理，然后将处理结果保存在内存中或者写回到文件中。PDF文件处理技术的方法包括：

- 使用PDF处理库：比如iText、PDFBox等PDF处理库，可以直接调用库中的API对PDF文件进行处理。

- 自己编写PDF处理器：通过读取PDF文件的二进制数据，解析文件结构，获取其中的元数据、书签、页面内容等信息，然后根据需要进行修改、添加、删除等操作。

4. 光学字符识别技术

光学字符识别技术主要是调用微软Office 2003或2007的光学字符识别引擎，使用OCR API接口将PDF文档的图片转换为文字，然后将识别结果保存在内存中或者写回到PDF文件中。光学字符识别技术的方法包括：

- 使用OCR库：比如Tesseract、GOCR等OCR库，可以直接调用库中的API将图片转换为文字。

- 使用OCR API接口：比如微软Office 2003或2007的OCR API接口，可以通过调用API将图片转换为文字。

5. 文件处理技术

文件处理技术主要是能够生成PDF文件、拆分、合并PDF文件、无损导出PDF文件的图片、提取PDF文件中指定的页面、根据PDF文档元数据重命名PDF文件名等。通过调用文件处理库或者自己编写文件处理器对文件进行处理，然后将处理结果保存在内存中或者写回到文件中。文件处理技术的方法包括：

- 使用文件处理库：比如Apache Commons IO、Java File API等文件处理库，可以直接调用库中的API对文件进行处理。

- 自己编写文件处理器：通过读取文件的二进制数据，解析文件结构，获取其中的元数据、页面内容等信息，然后根据需要进行生成PDF文件、拆分、合并PDF文件、无损导出PDF文件的图片、提取PDF文件中指定的页面、根据PDF文档元数据重命名PDF文件名等操作。

6. 大文件处理技术

大文件处理技术主要是支持打开并修改超过2G字节的超大PDF文档。大文件处理技术的编程思想是通过分块读取文件的二进制数据，分段处理文件内容，减少一次性读取文件的内存占用。大文件处理技术的方法包括：

- 分块读取文件：将文件分成多个块，每次读取一个块的数据进行处理，减少一次性读取文件的内存占用。

- 分段处理文件内容：将文件内容分成多个段，每个段进行单独处理，减少一次性处理文件的内存占用。

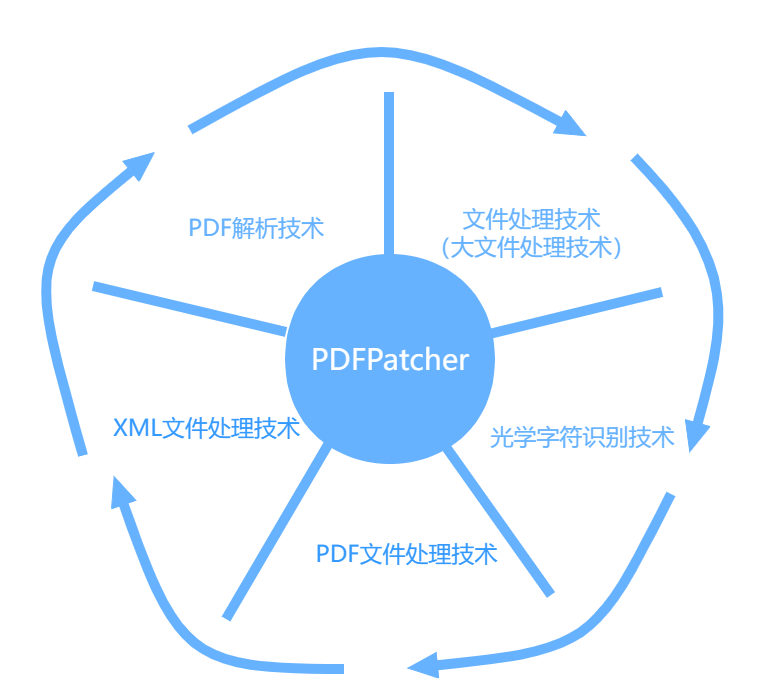


图 PDFPatcher使用的技术总图

**2.3 PDFPatcher解决的问题**

**2.3.1 问题需求分析**

当今数字时代，电子文档的阅读和处理已经成为了我们日常工作中不可或缺的一部分。而PDF格式作为一种跨平台、可靠性高的文件格式，越来越广泛地被使用。因此，一个多功能的PDF 处理工具能够满足用户在各种工作和生活场景中的需求，为用户提供更为便捷、高效、智能的PDF文件阅读、编辑、管理等功能。

本部分旨在通过现实实例充分分析PDF作为一个常用的文件格式的日常使用场景，阐释PDF使用刚性、需求广度，进而证明一个功能强劲、综合的PDF处理集成工具软件存在的必要性。

PDF格式的使用需求逐渐增大的证据主要包括以下几个方面：

1. 科研和学术领域的发展

PDF格式具有保留原始格式和内容的特点，使得科研和学术领域越来越多地采用PDF格式来存储和传输文献、报告和论文。根据Adobe Systems的报告，截至2018年底，全球每年使用PDF格式的科学论文数量已经超过3亿篇。此外，不仅是论文，PDF格式还广泛用于书籍、期刊、会议材料和演示文稿等各种学术资源的制作和发布。

1. 文件共享和在线传输的便捷性符合当前的互联网信息交流需求

PDF格式具有跨平台、易于共享、可读性好等特点，越来越多的人选择使用PDF格式来分享和传输文件。据统计，2020年全球每月有近2亿次关于PDF文件的搜索，而PDF文件的下载量更是呈现指数级增长。

1. 电子商务和办公需求的增加

随着电子商务和在线办公的发展，PDF格式的应用范围也在不断扩大。PDF格式可以用于制作电子合同、发票、报价单和各种表格，存储效果要优于一些普通的图片格式。同时，由于PDF特有的不易编辑性，使得其可以被添加各式各样的“签名”，进行各种类型的“加密”来满足特殊的业务要求。

1. PDF格式在其他各行业中的广泛应用

PDF格式已经在各行业中得到广泛应用，如教育、金融、医疗、政府等。在教育领域，教材、课件、考试试卷等都采用PDF格式。在金融领域，各种报表、合同、账单等也采用PDF格式。在医疗领域，病历、检查报告、处方等也采用PDF格式。在政府领域，各种公文、政策法规等也采用PDF格式。

综上所述，PDF格式的使用需求逐渐增大的证据是多方面的，包括PDF文件数量的增加、PDF软件的广泛应用、PDF格式的可靠性和便捷性以及PDF格式在各行业中的广泛应用。

**2.3.2 问题需求解决**

PDF补丁丁软件旨在解决现有PDF软件在处理某些PDF文件时出现的问题，比如复制限制、打印限制、文档隐藏垃圾数据、页面格式、文本乱码等问题，这些问题会导致PDF文件无法正常显示和编辑。PDF补丁丁软件在解决这些问题的基础上，还增加了方便用户查阅、操作PDF的工具，这些工具在其它PDF编辑软件中通常是收费的，且往往分散在不同PDF编辑软件中，而PDF补丁丁不仅集成了多种功能，还免费开放使用，使得用户能够更加方便、更加高效、更加省心、更加省钱地处理和编辑PDF文件。

**2.4 目标**

PDF补丁丁的目标是使PDF文件的处理和编辑更加准确、高效和方便，提高用户的工作效率。通过PDF补丁丁软件，用户可以对PDF文件进行字体更改、文本替换、页面旋转等操作，同时保持PDF文件的格式和布局不变，或者灵活多变地调节。

**2.5 影响**

PDF补丁丁软件的影响：①减少由于PDF文件格式问题导致的沟通和交流的困难。②减少了用户对当前PDF修改到理想效果所耗费的时间和精力。从前用户需要打开多个软件，甚至付费才能完成的操作，现在只需打开PDF补丁丁一个软件，就可以免费地使用。③同时，提高了用户在工作中的效率，进一步提高了办公效率和工作质量。

总之，PDF补丁丁软件填补了市场上现有PDF软件的不足，使PDF文件的处理和编辑更加方便、高效和准确，提高了用户的工作效率和办公效力，同时也为开源软件的发展和推广做出了贡献。

**2.6 背景分析**

**2.6.1 现实情况**

在工作场景中，PDF 处理工具能够帮助用户更好地处理和管理各种文档。例如，一份工作合同需要签署，用户可以使用PDF 处理工具对该合同进行电子签名，从而避免打印、签字、扫描等繁琐步骤；当需要将多个PDF文档合并为一份文档时，用户可以使用PDF 处理工具的合并功能，轻松实现合并操作；同时，PDF 处理工具还可以支持用户对文档进行标注、批注、划重点等操作，提高文档阅读和处理效率。

在学习场景中，PDF 处理工具软件也能够为用户提供很大的帮助。例如，在进行文献阅读时，用户可以使用PDF 处理工具的标注功能，将自己的想法、感受记录在文档中，方便后续查找和回顾；在进行学术写作时，用户可以使用PDF 处理工具的引用功能，方便地引用文献并生成参考文献列表；在进行英语学习时，用户可以使用PDF 处理工具的翻译功能，轻松阅读英文文档。

在生活场景中，PDF 处理工具软件同样有很多的应用场景。例如，在旅游时，用户可以使用PDF阅读器查看各种旅游手册、地图等信息；在进行健身时，用户可以使用PDF阅读器查看各种健身指南、食谱等信息；在进行家庭管理时，用户可以使用PDF阅读器查看各种账单、合同等信息。

综上所述，一个多功能的PDF处理工具可以帮助用户更好地处理和管理各种文档，提高文档阅读和处理效率，同时在学习、工作和生活等各个场景中提供便利。

**2.6.2 未来趋势**

然而可以预见，电子文档作为一个现代信息几乎唯一的载体的出现，在未来很长一段时间内不会被其他介质所取缔。因此，电子文档的典型类型PDF的各方面处理需求也一定会只增不减。目前没有及时实现的一些功能，在未来也会成为迫切的需要。

我们可以针对未来的发展趋势做出如下的分析和展望：

随着科技的不断发展，PDF处理工具软件的应用也在不断拓展和创新。

（1）客观上分析，未来，PDF处理工具软件将会有以下几个趋势：

云端服务：PDF处理软件将越来越多地向云端转移，以便用户可以更轻松地在不同的设备上使用和访问它们的PDF文档。此外，通过将PDF文档上传到云端，用户可以更方便地与他人共享和协作。

AI和自动化：PDF处理软件将越来越多地集成人工智能（AI）和自动化技术，以提高处理效率和准确性。包括以下几点：①自动化处理：将PDF文档的创建、编辑、转换和管理等过程自动化，提高效率和准确性；②智能化识别：将PDF文档中的文字、图像和表格等元素进行自动识别和分类，提高数据分析和应用的效率；③个性化定制：根据用户的需求和习惯，提供个性化的PDF文档模板、样式和功能，满足用户的个性化需求。例如，自动化工具可以自动识别文本、图像和表格，并将其转换为可编辑的格式。

移动应用：随着移动设备的普及，PDF处理工具软件将越来越多地面向移动应用程序。这将使用户可以随时随地查看、编辑和共享PDF文档，从而提高生产力和灵活性。

跨平台功能嫁接：PDF处理软件将会越来越多地支持跨平台功能，以便用户可以在不同的操作系统和设备上无缝地使用和访问PDF文档。

安全性细粒度强化和隐私化：由于PDF文档可能包含敏感信息，因此PDF处理软件将越来越多地注重安全性和隐私保护。这可能包括加密和权限控制功能，以确保PDF文档只能被授权的人员访问。可以将以下三种技术应用于PDF处理，使得更加高效、安全和可靠：①基于区块链技术的PDF文档管理和安全验证，保证PDF文档的真实性和完整性；②基于人工智能和自然语言处理技术的PDF文档自动翻译和语音合成，提高PDF文档的跨语言和无障碍使用；③基于云计算和大数据技术的PDF文档分析和应用，挖掘PDF文档中的价值和信息。

总的来说，未来PDF处理工具软件将会更加注重用户体验、效率、安全性和隐私保护。而这些功能和上面所分析的现状所需求的功能中，有一部分已经在我们所分析的项目——PDF 补丁丁（PDFPatcher）中得以实现。

（2）从用户角度而言，我们希望未来的PDF处理器能够实现以下几个功能：

自动识别和转换PDF文件中的表格：目前，处理PDF文件中的表格需要手动进行识别和转换，非常繁琐。希望未来的PDF处理器可以自动识别PDF文件中的表格，并将其转换为Excel或其他格式，以便进行更方便的数据处理。

实时协作和编辑功能：希望未来的PDF处理器可以实现实时协作和编辑功能，多人可以同时对PDF文件进行编辑和注释，以提高工作效率和协作效果。

自动化批量处理功能：希望未来的PDF处理器可以实现自动化批量处理功能，可以一次性处理多个PDF文件，如批量转换、批量压缩、批量加密等，以提高工作效率和节省时间。

自动化OCR识别功能：希望未来的PDF处理器可以实现自动化OCR识别功能，可以自动识别PDF文件中的文字内容，并将其转换为可编辑的文本格式，以便进行更方便的编辑和处理。

手写笔记和标注功能：希望未来的PDF处理器可以实现手写笔记和标注功能，可以使用手写笔对PDF文件进行标注和注释，以方便进行更直观的沟通和交流。

（3）从开发者角度而言，我们会使用以下技术和方法来实现上述用户设想的功能：

自动识别和转换PDF文件中的表格：可以使用OCR技术对PDF文件进行识别，并使用表格识别算法将表格转换为可编辑的格式。可以使用Python的Pillow库进行图像处理和OCR识别，使用Python的pandas库进行表格处理和转换。

实时协作和编辑功能：可以使用WebSocket技术实现实时协作和编辑功能，多人可以同时对PDF文件进行编辑和注释。可以使用JavaScript的WebSocket API实现客户端和服务器之间的通信，使用Node.js和Express框架实现服务器端。

自动化批量处理功能：可以使用Python的PyPDF2库和其他相关库实现自动化批量处理功能，可以一次性处理多个PDF文件，如批量转换、批量压缩、批量加密等。

自动化OCR识别功能：可以使用Python的PyPDF2库和OCR库实现自动化OCR识别功能，可以自动识别PDF文件中的文字内容，并将其转换为可编辑的文本格式。可以使用Python的Tesseract OCR库进行OCR识别。

手写笔记和标注功能：可以使用JavaScript的Canvas API实现手写笔记和标注功能，可以使用HTML5的Canvas元素绘制PDF文件，并使JavaScript的事件监听器和绘图函数实现手写笔记和标注功能。

（4）从细节功能上看，以对书签功能和PDF合并功能优化为例：

如果要对书签功能进行优化，需要从以下几个方面考虑：

①提高自动书签的准确性和智能性：通过加强OCR技术的应用，提高自动书签的识别准确性和自动化程度，使得用户无需手动添加书签，即可快速生成准确的书签信息。

②增加书签功能的多样性：除了基本的书签功能外，还可以增加类似于书签绑定、书签分享、书签评论等多样化的功能，让用户更加方便地管理书签信息。

③提高书签的可视化效果：通过增加书签的颜色、图标、标签等视觉元素，使得书签信息更加直观和易于辨认，提高用户的使用体验。

④改善书签的操作流程：通过优化书签的添加、编辑、删除等操作流程，减少用户的操作步骤和时间，提高用户的使用效率。

⑤加强书签与其他功能的整合：通过将书签与其他功能进行整合，如书签与目录、书签与注释、书签与搜索等，使得用户更加方便地使用各种功能，提高软件的综合实用性。

若要对现有的pdf合并功能做出优化，可以从以下方面考虑：

①提高合并速度：优化算法、增加计算资源等措施可以提高合并速度，提高用户体验。

②增加合并方式：除了简单的合并多个pdf文件，可以增加按页码、按书签等方式进行合并。

③支持多种文件格式：除了pdf文件，可以支持其他常见文件格式的合并，如doc、xls、ppt等。

④提供在线编辑功能：用户可以在线编辑已合并的pdf文件，包括添加书签、插入页码、添加水印等。

⑤增加批量处理功能：支持同时合并多个文件夹中的pdf文件，提高处理效率。

⑥提供安全保障：确保合并过程中用户数据的安全性和隐私性，防止信息泄露。

⑦提供用户反馈机制：通过用户反馈，及时了解用户需求，优化合并功能，提高用户满意度。

⑧支持自动去重：让用户合并pdf文件时，自动去除重复的页面，避免合并后出现重复页面。

⑨支持加密合并：让用户可以对合并后的pdf文件进行加密，保护文件的隐私性。

**3 内容**

**3.1 问题解决方案**

**3.1.1 内容**

对于2.3节中所描述的PDF文档相关的各种现实问题与需求，应考虑设计一个PDF 处理工具，使它具有以下待设计模块：

**界面设计**

为了提高用户体验，可以采用现代化的界面设计。界面中应该包括菜单、工具栏、快捷键等，以便用户方便地访问和使用功能。界面应该简洁明了，符合用户的习惯和预期。

**查看PDF文档**

为了查看PDF文档，可以使用开源的PDF渲染库，如MuPDF、Poppler等。这些库可以将PDF文档渲染成图片，以便用户进行查看。同时，还可以提供缩放、旋转、平移等功能，以方便用户浏览文档。

**转换PDF文档**

为了转换PDF文档，可以使用开源的PDF转换库，如PDFBox、Ghostscript等。这些库可以将PDF文档转换成其他格式，如Word、Excel、图片等。同时，还可以将其他格式的文档转换成PDF格式。

**修改 PDF 文档**

修改文档属性、页码编号、页面链接；统一页面尺寸；删除自动打开网页等动作；去除复制及打印限制；设置阅读器初始模式；清理文档隐藏垃圾数据；重新压缩黑白图片；旋转页面。为了编辑PDF文档，可以使用PDF编辑库，如iText、PDFsharp等。这些库可以提供添加、删除、移动、修改文本、图片、表格等内容的功能。同时，还可以提供注释、书签、目录等功能，以便用户更方便地阅读文档。

**贴心 PDF 书签编辑器**

带有阅读界面（具有便于阅读竖排文档的从右到左阅读方式），可批量修改 PDF 书签属性（颜色、样式、目标页码、缩放比例等），书签可精确定位到页面中间；在书签中执行查找替换（支持正则表达式及 XPath 匹配、可快速选择篇、章、节书签），自动快速生成文档书签。

**制作 PDF 文件**

合并已有 PDF 文件或图片，生成新的 PDF 文件；合并后的 PDF 文档带有原文档的书签，还可挂上新书签（或根据文件名生成），新书签文本和样式可自定义；合并的 PDF 文档可指定统一的页面尺寸，以便打印和阅读。

拆分或合并 PDF 文件，并保留原文件的书签或挂上新的书签。

高速无损导出 PDF 文档的图片。

将 PDF 页面转换为图片。

提取或删除 PDF 文档中指定的页面，调整 PDF 文档的页面顺序。

根据 PDF 文档元数据重命名 PDF 文件名。

调用微软 Office 的图像识别引擎分析 PDF 文档图片中的文字；将图片 PDF 的目录页转换为 PDF 书签。识别结果可写入 PDF 文件。

**替换字体**

替换文档中使用的字体；嵌入字库到 PDF 文档，消除复制文本时的乱码，使之可在没有字库的设备（如 Kindle 等电子书阅读器）上阅读。

**分析文档结构**

以树视图显示 PDF 文档结构，可编辑修改 PDF 文档节点，或将 PDF 文档导出成 XML 文件，供 PDF 爱好者分析、调试之用。

设计一个PDF处理工具软件，可以实现PDF文档的查看、编辑、转换等功能。下面是一个可能的设计方案：

**综合功能**

除了上述功能之外，还可以提供OCR识别、合并PDF文档、拆分PDF文档、加密PDF文档等其他功能，以满足用户的需求。

**3.1.2 方案**

那么具体进行的方案应该包括以下步骤：

**确定需求**

首先需要确定所设计的PDF处理工具软件的具体需求。例如，是否需要支持PDF文件的编辑、转换、加密、解密、签名、压缩等功能，以及是否需要支持批量处理等。这些内容我们已经在上文中讨论清楚。

**界面设计**

根据需求，设计软件的界面。界面应该简洁明了，易于操作。可以采用图标、菜单、工具栏等方式，使用户能够轻松地找到所需功能。本次项目，我们选用C#作为基础语言，方便我们进行交互窗口的设计与图形化处理。

**PDF文件处理**

实现PDF文件的处理功能。可以采用现有的PDF处理库或者编写自己的PDF处理代码。例如，可以使用iText、PDFBox、Ghostscript等PDF处理库，或者使用Java、Python等语言编写自己的PDF处理代码。

**软件测试与维护**

完成软件开发后，进行软件测试，并在之后进行长期的维护。软件测试是软件质量保证的主要手段之一，也是在将软件交付给客户之前所必须完成的步骤。目前，软件的正确性证明尚未得到根本的解决，软件测试仍是发现软件错误和缺陷的主要手段。软件经过测试，交付给用户后，在使用和运行的过程中可能对于软件产品进行的修改就是维护。

在本项目中，我们无法获得到海量的测试机会，故测试与维护将局限在我们组内的测试和代码审查之中。

**3.2 功能**

PDFPatcher是一个用于修改PDF文件信息的工具。它具有以下功能：

（1）修改文档：

* + 修改文档属性（如作者、主题、关键词等）。支持通过文件名重写文档属性。
  + 书签编辑器：添加、修改或删除PDF书签，设置书签的文字颜色、打开或收拢状态、点击后的跳转位置及页面缩放比例等修改PDF书签的功能。
  + 替换文档中使用的字体，或将字体嵌入到原本没有嵌入字体的PDF文档。
  + 添加或修改页面内的链接。
  + 添加或更改PDF文档的逻辑页码编号。
  + 更改阅读器的初始设置（如显示比例、界面等）。
  + 统一页面尺寸、裁剪或扩大页面尺寸。
  + 调整页面旋转方向。
  + 删除文档或页面内嵌的XML元数据。
  + 删除打开文档或页面时自动执行的动作。
  + 去除PDF文件复制、打印限制。

（2）高级补丁修改功能

* + 首先导出信息文件（将PDF文档属性、阅读器初始状态、页码设置、页面设置、书签等信息导出成可编辑的XML文件。
  + 导入信息文件生成新文件：将上述信息文件和已有PDF文件合并，生成新的PDF文档，该PDF文档具有XML信息文件的设置（如页面设置、书签等）。

（3）制作PDF文件：

通过导入一批图片或已有PDF文件，生成包含图片及已有PDF文件指定页面范围的PDF文件（可挂上书签）。此功能还可拆分或合并PDF文件，并保留原文件的书签。

（4）自动生成PDF书签：

分析PDF文档的文本，自动生成书签。

（5）文字识别：调用微软Office 2003或2007的光学字符识别引擎，将PDF文档的图片转换为文字。识别结果可写入PDF文档。

（6）无损导出PDF文件的图片。

（7）提取PDF文件中指定的页面。

（8）根据PDF文档元数据重命名PDF文件名。

（9）分析文档结构：以树节点形式查看PDF文档结构，可将页面或文档内容导出成XML文件、二进制文件供PDF爱好者分析、调试之用。

（10）支持打开并修改超过2G字节的超大PDF文档。

综合上述分析，可将软件功能分为大致以下几类：

（1）修改文档属性、更改页码、替换字体等文档编辑操作

（2）自定义样式、自动生成书签等书签相关操作

（3）导出并编辑xml文件，将xml文件与已知文件合并的文档移植操作

（4）更改阅读器的初始设置、删除自动执行的动作等设置操作

（5）文字识别、无损导出图片等图片操作

（6）分析、调试文档结构等结构操作

（7）制作PDF文件、修改超大pdf文件的文件操作

（8）替换字体、更改样式的风格操作

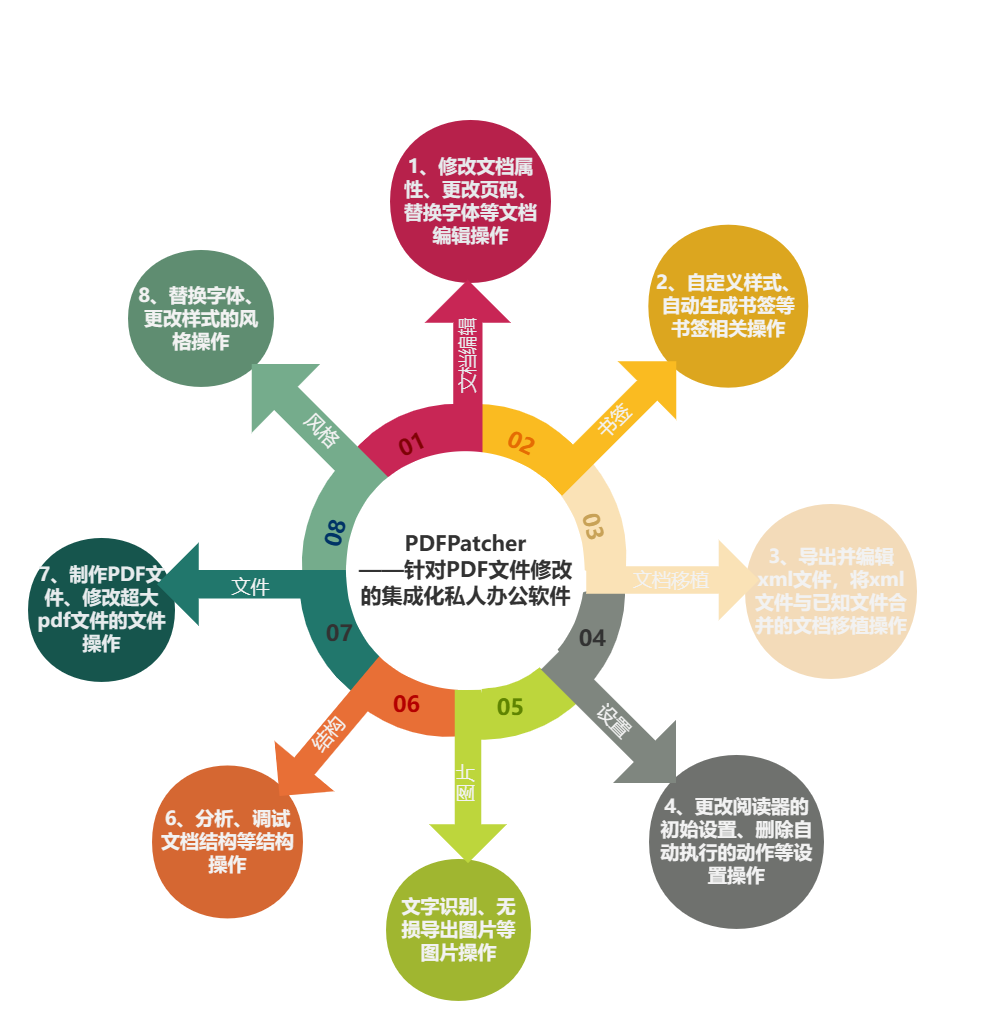


图 PDFPatcher功能示意图

**3.3 应用案例和具体场景**

**3.3.1 样例一**

A公司兹要求员工S提交100个分部财政文档PDF的合并版本，通过采用我们所设计的工具，可以在一个交互框中选中所有列出的PDF文件，上传到工具中后，工具进行一次集体合并，以上传顺序为依据，将所有PDF文件拼合成一个文件。

软件内部的流程图类似：

Start ->

（通过交互窗口集体选定）输入要合并的PDF文件 ->

（通过交互窗口的选定）输入合并后的文件名和路径 ->

（通过工具“后端”）调用PDF处理库将多个PDF文件合并成一个文件 ->

输出合并后的文件到指定路径 ->

End

**3.3.2 样例二**

由于工作需要，甲需将一个文件夹中的数个PDF文件中的一些表格进行编辑。

需要编辑PDF文件中的文本或表格的情况中，由于PDF格式的文档是不可编辑的，我们难以直接改变PDF中的内容。因此，常需要将PDF文件转换为可编辑的文档格式，如Word、Excel、PowerPoint等。当然，也可以通过工具里的编辑功能，直接实现对PDF的删改。这个需求可以通过我们项目的PDF处理工具软件来实现。

软件内部的流程图类似于：

Start ->

（通过交互窗口集体选定）选择要转换的PDF文件 ->

（通过交互窗口的选定）选择输出文件的格式和路径 ->

（通过工具“后端”）调用PDF处理库将PDF文件转换为可编辑的文档格式 ->

（与上类似）调用文档格式转换库将文档格式转换为指定格式 ->

输出转换后的文件到指定路径 ->

对生成的可编辑文档进行编辑 ->

End

或者

Start ->

（通过交互窗口集体选定）选择要转换的PDF文件 ->

（通过交互窗口的选定）选择输出文件的格式和路径 ->

（通过工具“后端”）调用PDF读取库将PDF文件批量载入交互端 ->

（通过工具的编辑功能实现对于文档的直接增删改查）直接进行编辑 ->

输出转换后的文件到指定路径 ->

End

**4 特征和影响**

**4.1 运行环境**

* Windows 7 以上版本的操作系统。
* .NET Framework 4.0 到 4.8 版本。
* 使用文字识别功能需要安装 Microsoft Office 2003（或 2007）的 Document Imaging 组件（MODI）。
* 编译程序源代码，建议使用 Visual Studio 2019 或更新版本，并安装“.NET 桌面开发”（用于编译 PDFPatcher源代码）和“C++ 桌面开发”（用于编译 JBIG2 编码组件）两个工作负载。如使用 Visual Studio 2022，可能会遇到项目“面向不再受支持的 .NET Framework”、需要“将目标更新为 .NET Framework 4.8”的问题。简单方法是将目标更新为 .NET Framework 4.8。

**4.2 用户**

PDFPatcher的特征是安装简单，上手容易。只需要下载exe文件和以来的.dll文件就可以直接运行。对于用户来说，PDFPatcher使用十分简单，非常符合使用直觉，而且功能强大。例如，PDFPatcher不仅仅有鼠标点击操作，还支持很多鼠标拖动操作，极大方便了用户的使用。

PDFPatcher的另一个特征就是面向群体广泛，既有普通用户，也面向专业的开发人员，一方面开发人员可以使用PDFPatcher帮助整理PDF结构，另一方面PDFPatcher结构完善，很有利专业人员的软件开发的学习。

PDFPatcher的影响：PDFPatcher在用户群体中广受好评，很多功能只此一家独有，例如无损把PDF导出为图片。稳定性和知名度都非常高。

**4.3 功能运行**

# 文件批处理

## （1）文件选择

PDFPatcher内使用“打开文件”按钮可向文件列表添加需处理的文件，也可以通过资源管理器直接拖动一批文件进入列表中，随后，使用鼠标左键拖动选中的项目可以用来调整文件批处理的顺序，也可以选中“按数值和字母排序”将文件名中的数字考虑为数值，选中“按字母顺序排序则根据文件名的字母顺序排序”。如果想要删除文件项目，可直接单击删除按钮删除选中的项目。

## （2）修改文件属性

点击文件列表中文件项目的“标题”、“作者”、“主题”和“关键词”列，列表项将进入编辑状态。编辑项目的文本，可修改输出文件的属性。注意这里只会修改输出文件的属性而不会修改源文件属性。如果PDF文件打开为乱码，说明当前编码模式不正确，可以选择“刷新文档属性”来更改编码模式。支持UTF-16 Big Endian, UTF-16 Little Endian, UTF-8, BG18030, BIG5。

## （3）独立修改文件

使用“独立补丁”模式可以分别为每个PDF添加处理方式。在“配置处理文件选项”总可以指定阅读器界面，去除赋值和打印限制，禁止书签，改变阅读比例等功能。

# 高级补丁功能

高级补丁功能用于将PDF文档的信息导出成一个可编辑的文档。该文档包含PDF文档属性、书签、页面链接、页面尺寸等信息，称为“信息文件”。

补丁操作分两步进行：

第一步，[导出信息文件](#_导出信息文件)：将PDF文档中的信息导出成[可编辑修改的XML信息文件](#_信息文件参考)。

第二步，[导入信息文件](#_导入信息文件)：使用文本编辑器修改上一步骤导出XML文件，然后在处理PDF文件处指定信息文件，点击“生成PDF文件”按钮，将源PDF文件和信息文件合并，生成一个具有修改后的设置的新PDF文件。

## 导出信息文件

选择“处理PDF文档”的“独立补丁”处理方式。想文件列表中添加需要导出信息的PDF文件，将原始PDF文件的信息导出到“PDF信息文件”的对应位置。导出信息文件可以指定文档属性，文档书签，页面内的链接，解析书签和页面链接的命名位置，阅读器设置，尺寸度量单位。还可以选择以XML表示导出PDF页面的内容以方便PDF开发人员。

## 导入信息文件

若导出了信息文件，可以使用文本编辑器或者XML编辑器打开修改，然后将修改后的信息文件合并到原始PDF文件，生成一个新的PDF文件。如果不需要将信息文件的全部内容导入PDF文件，可以在“导入选项”处指定导入内容。设置类似于上文导出信息文件的设置。

# PDF文档选项

PDFPatcher可以指定页面尺寸，包括页面宽高，对齐方式，页面留白，自动设置页面尺寸等。主要用于显示PDF文件，增强用户使用体验。

通过改变阅读方式，可以改变PDF阅读器显示的信息内容和翻页方式。设置文档的阅读方向为“从右到左”可以用于适应一些古籍的阅读。设置书签状态可以用于指定是否打开或者关闭所有书签。

PDFPatcher还提供PDF文件压缩清理功能，用于提高PDF文档的压缩率，并可以清理一些不需要的内容。可以清除页面所有文本，压缩索引表和书签，优化压缩黑白图片。优化压缩黑白图片使用JBIG2算法压缩PDF文档内的黑白图片以求减少图片所占用的文件空间，如果压缩之后无法减少文件大小，就放弃压缩。

PDFPatcher还可以在文档属性选项卡总指定文档属性。一般场合下不需要使用此选项卡设置文档属性，在文档列表中直接输入需要设置的文档属性即可。

嵌入字体功能可以用于将PDF文档的字体更换为系统安装的其他字体。对于一些没有嵌入字体的PDF文档，还可以实现嵌入字体的功能。

# PDF合并

PDF合并文件功能可实现多种方式制作PDF文件的功能，例如制作扫描书籍的电子书、将整个目录的图片和PDF文件合并为一个PDF文件等。合并功能还可在已有的PDF文件中抽取指定范围的页面，导出到生成的PDF文件，即具有合并或拆分PDF文件。在界面正中有一个列表框，称为“文件列表”。在文件列表，可混合指定图片和PDF文件，将图片插入到PDF文件的页面中。双击列表中的源PDF文件，可以指定该文档中参与合并操作的页码范围。页码范围默认为第一页到最后一页，通过修改页码范围，可实现拆分、重排PDF文件页面的功能。

## 设置书签文本和书签样式

合并生成PDF文件时可指定导航书签，在阅读器中点击书签将跳转到与文件对应的页面。

书签文本在合并文件列表的“书签文本”列设置。将文件添加到合并文件列表后，点击“书签文本”列的单元格。该单元格将进入编辑状态，在其中输入文本，即可为文件设置对应的书签项。

在书签文本不为空的情况下，选中文件项，再点击工具栏上的“B”和“I”按钮，可分别为书签项设置粗体和斜体样式；点击工具栏上调色板按钮旁边的倒三角形小按钮，可设置书签项的文本颜色。

如果书签文本内容为空，则该项目不生成书签。如果该项有子项，且子项文本不为空，则子项仍生成对应的书签。

## 合并文档选项

合并文档选项用于指定合并功能生成的PDF文档的选项。点击工具栏上的“合并文档选项”项目，或点击界面上的“设置合并PDF文档选项”链接，将打开“合并文档选项”对话框。

选项对话框以选项卡形式划分选项。界面分“页面布局”、“阅读方式”、“书签”、“文档杂项”和“页码标签”等选项卡。“阅读方式”、“文档杂项”和“页码标签”的内容和功能与PDF文档选项基本一致。

## 书签合并

添加项目时根据文件名自动生成书签文本：选中此项时，为每个源文件（图片或源PDF文件）对应的页面添加一个书签项，书签项的文本为源文件名（删除扩展名部分）。

忽略文件名的前导数字：在选中“添加项目时根据文件名自动生成书签文本”复选框时，此选项有效。选中此项时，书签项的文本为源文件名删除前面的数字后的部分。如删除数字后文件名为空，则不为该页添加书签。例如，用于生成PDF的文件列表有4个图像文件，其文件名分别为：“0001封面.jpg”、“0002目录.tif”、“0003.tif”和“0004第一章.tif”，则生成的书签有三项，名称为“封面”（跳转到第1页）、“目录”（跳转到第2页）和“第一章”（跳转到第4页）。

子文件夹排在文件前面：选中此选项时，在添加文件夹到文件列表时，文件夹包含的子文件将排在文件前面。不选中此选项时，按照文件的字母顺序排序文件。

文件名分别按数值和文本排序：选中此选项时，在添加文件夹到文件列表时，文件夹包含的文件将分别按数值和文本排序，例如以下序列是按数值和文本排序1.pdf、2.pdf和10.pdf（排序时将“10”视为数值，排在“2”后面），而1.pdf、10.pdf和2.pdf是按文本排序（排序时将“10”视为文本，排在“2”前面）。

## 自动生成书签

此功能可用于自动生成PDF文档的书签。其原理是：PDF文档的文本有一定的格式规律，因此可按照页面文本、尺寸、所用字体等规则，筛选出符合规则的文字（往往是大字标题），从而利用这些文本生成PDF书签。

选择PDF文件之后，就可以指定需要识别标题作为书签的PDF文件。可以通过调整选项来选择性导出不同信息。点击“导出信息文件”按钮。程序将分析识别原始PDF文件中指定页码范围页面内的文版，抽取尺寸大于“标题文本尺寸”的文本组成成为书签，也可以保存为XML信息文件。如果一次分析的书签不太理想，可以重新调整筛选选项，如调整标题文本尺寸以包含字体更小（或更大）的标题文本、指定过滤文本筛选不需要的文本、在高级筛选处理选项中添加筛选器剔除特定的字体等。

## 编辑书签文件

在软件文件编辑器中可以对书签文件进行编辑。选择“编辑书签文件”功能。点击“打开”按钮，指定需要编辑的书签文件或者PDF文件。加载文件之后，可对侧边树形列表的书签进行修改。之后可以将书签保存为信息文件。使用上文高级补丁可以将书签文件添加到导出文件中。

通过工具栏的左右箭头按钮，可以调整书签的层次，修改书签文件的组织结构，也可以直接拖动书签到指定位置。

书签编辑器的另一大特色是可以批量搜索并替换书签文本。点击“搜索及替换书签文本”按钮，将弹出如下图所示的对话框，使用该对话框可完成搜索替换书签文本任务。默认搜索不区分英文大小写，如需区分，可选择“区分大小写”复选框。

搜索模式有三种。第一种是“普通”模式，用来检索书签文本是否包含搜索文本。第二种是“正则表达式”模式，搜索文本被视为一个正则表达式，当书签文本符合该表达式时，就算是匹配。第三种是“XPath”模式。在该模式下，书签编辑器的书签被视为一个XML文档，每个书签是一个“书签”元素，书签的属性（如目标页码、动作等）则被视为XML“书签”元素的属性。搜索文本被视为一个XML路径表达式，当书签元素匹配该表达式时，视为匹配。书签XML文档的结构可参考信息文件的[文档书签](#_文档书签)一节。

# 识别图像文本

识别图像文本功能可将PDF文档内图片的文字识别出来，变成可编辑的文本文件，在编辑文件后，可将该文本写入PDF文档成为透明的文本层，使PDF文档的图像文本可被检索工具索引。用户可以通过“文字识别语言”，“旋转矫正”，“拉伸矫正“，”检测页面方向“，“纠正倾斜页面”等功能调整PDF图片样式，提高识别准确率。

另外，识别图像文本还可以将扫描PDF文档目录页识别成建议书签文件，用于快速制作PDF书签。识别后的输出文本文件可以直接使用书签编辑器打开。使用书签编辑器的“更改目标页码”功能可以修复目录页码与实际页码的对应关系。

# 文件提取

该功能可以将PDF文档的图片无损导出为图片文件，或者可以提取选中的页面导出为新的PDF文件。可以尝试合并相同页面的图片：尝试将同一页内具有相同水平坐标、相同宽度和相同格式的图片合并为一副图片。提供这个功能的原因是有些PDF制作工具会将一副图片分割成多片，逐片写入PDF文件。这样，提取出来的图片将是零散的。使用此功能会将这些零散的图片重新整合为一个图片。JPEG图片无损合并为PNG图片：选择“尝试合并相同页面的图片”时，如合并前的图片为JPEG图片，则将这批图片无损合并为PNG图片。

# 文档结构探查

使用“文档结构探查器”可以以树形列表展示PDF文档内部c恩赐结构。列表包含各个节点名称，可以展开具有子节点的节点，文档的根节点是文档编录节点。程序将文档的各个页面转换成单独的页面节点，独立显示在Pages节点下。列表下方是备注窗口，讲解所选系欸党的含义以及取值范围。

如果选中文档的某个数据流绩点，可以将内容节点导出成二进制文件或者文本文件供外部程序分析。

# 软件任务

为普通用户提供相应的PDF管理，编辑，剪切，导出等功能，为PDF开发者提供PDF文件结构分析功能。

设备要求：支持安装Windows7至Windows10系统的设备

环境要求：要求计算机安装.NET Framework 2.0或更高版本，图片文字识别以及自动导出书签功能需要机器上安装了微软Office2003或者Office2007或更新版本的Document Imaging组件识别引擎。

**5 验收建议**

**5.1 优化方向**

1. 对于修改文档和制作PDF文件功能，考虑增加自动化和批量操作功能，让用户可以一次性处理多个文件，提高效率。

2. 自动生成PDF书签可以帮助用户更快速地浏览和查找PDF文件的内容，考虑优化书签的生成算法，提高准确性和速度。

3. 对于无损导出图片功能，考虑增加对不同图片格式的支持，并优化导出算法，提高图片质量和导出速度。

4. 对于大型PDF文件，考虑增加分页和分段功能，提高文件的可读性和管理性。

5. 考虑增加对PDF文件的合并和拆分功能，让用户可以更方便地管理和处理PDF文件。

6. 考虑增加对PDF文件的注释和批注功能，让用户可以更方便地进行文档标注和交流。

在PDFPatcher中，还可以添加以下功能：

1. 加密和解密功能：可以为PDF文档添加密码保护，防止未经授权的人员查看或修改文档。实现思路是使用加密算法对PDF文档进行加密，只有输入正确的密码才能解密。

2. 数字签名功能：可以为PDF文档添加数字签名，确保文档的完整性和真实性。实现思路是使用数字证书对PDF文档进行签名，验证签名的过程中，会检查文档是否被篡改过。

3. OCR识别功能：可以对扫描文档进行OCR识别，将文档中的图片转换为可编辑的文本。实现思路是使用OCR技术对文档中的图片进行识别，将识别结果转换为文本。

4. 表单编辑功能：可以对PDF文档中的表单进行编辑和填写，提高表单处理效率。实现思路是使用表单编辑器对PDF文档中的表单进行编辑和填写。

**5.2 验收列表**

1. 修改文档功能：

- 验证能否添加、删除和编辑文本和图像

- 验证能否添加、删除和编辑注释和标记

- 验证能否添加、删除和编辑表格和表格内容

- 验证能否添加、删除和编辑链接和样式

2. 自动生成书签功能：

- 验证书签是否自动创建和命名

- 验证书签是否准确地反映文档的内容和结构

- 验证书签是否可自定义和编辑

- 验证书签是否可导出和导入

3. 文字识别功能：

- 验证是否能够正确识别文档中的文字

- 验证是否支持多种语言文字识别

- 验证是否能够保留文档中的原始格式和排版

- 验证是否能够识别手写文字或扫描文档中的文字

4. 无损导出图片功能：

- 验证导出的图片是否保留原始的分辨率和质量

- 验证导出的图片是否支持多种格式（如JPEG、PNG、TIFF等）

- 验证导出的图片是否能够自定义分辨率和尺寸

- 验证导出的图片是否能够批量处理

5. 提取指定页面功能：

- 验证是否能够准确地提取指定页面

- 验证是否能够批量提取多个页面

- 验证提取的页面是否能够保留原始的格式和排版

- 验证提取的页面是否能够导出为多种格式（如PDF、Word、Excel等）

6. 分析文档结构功能：

- 验证能否自动识别文档的标题、正文、图表、表格等结构元素

- 验证能否自动识别文档的章节、段落、页眉、页脚等结构信息

- 验证能否自动生成文档结构树或目录

- 验证能否手动编辑和更新文档结构信息

7. 导出信息文件功能：

- 验证能否将文档结构信息导出为多种格式（如XML、HTML、CSV等）

- 验证导出的信息文件是否能够准确地反映文档的结构信息

- 验证导出的信息文件是否能够自定义和编辑

- 验证导出的信息文件是否能够批量处理和导入

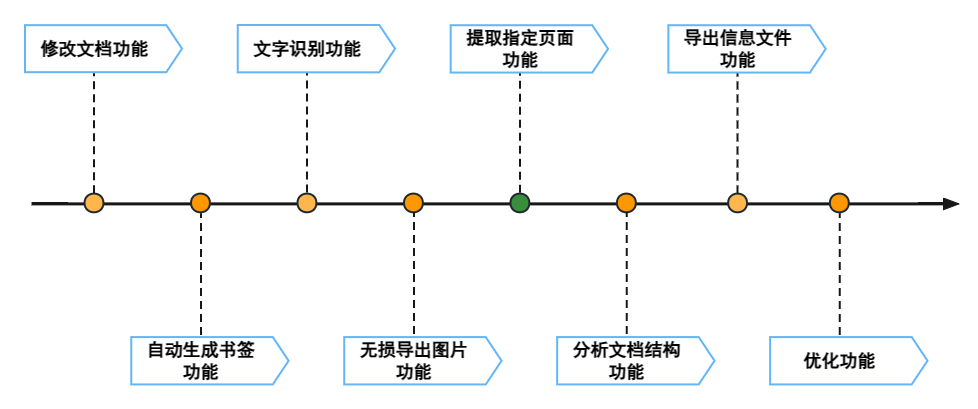


图 验收列表