(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 107291677 A
(43)申请公布日 2017.10.24

(21)申请号 201710576555.4

(22)申请日 2017.07.14

(71)申请人 北京神州泰岳软件股份有限公司
地址 100089 北京市海淀区万泉庄路28号万柳新贵大厦A座601室
申请人 中科鼎富(北京)科技发展有限公司

(72)发明人 徐龙 王文军 房平会

(74)专利代理机构 北京弘权知识产权代理事务所(普通合伙) 11363
代理人 逯长明 许伟群

(51)Int.Cl.
G06F 17/22(2006.01)

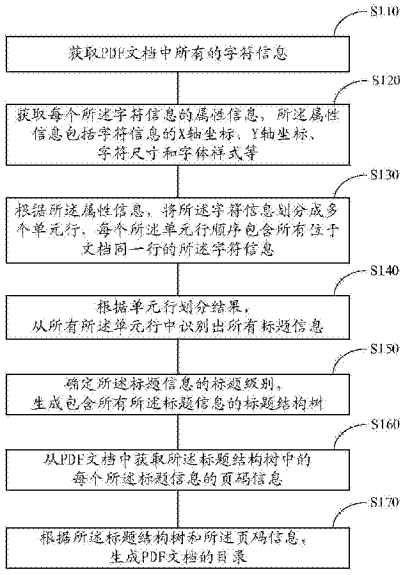
权利要求书3页 说明书12页 附图5页

(54)发明名称

一种PDF文档标题结构树生成方法、装置、终端及系统

(57)摘要

本发明实施例提供了一种PDF文档标题结构树生成方法、装置、终端及系统，为了解决从PDF文档中提取文档的标题结构的问题，首先，获取PDF文档中所有的字符信息；然后，获取每个所述字符信息的属性信息，所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等；其次，根据所述属性信息，将所述字符信息划分成多个单元行，每个所述单元行顺序包含所有位于文档同一行的所述字符信息；再次，根据单元行划分结果，从所有所述单元行中识别出所有标题信息；最后，确定所述标题信息的标题级别，生成包含所有所述标题信息的标题结构树，解决了现有技术中无法从PDF文档中提取文档标题结构的问题。



CN 107291677 A

1.一种PDF文档标题结构树生成方法，其特征在于，所述方法包括：

获取PDF文档中所有的字符信息；

获取每个所述字符信息的属性信息，所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等；

根据所述属性信息，将所述字符信息划分成多个单元行，每个所述单元行顺序包含所有位于文档同一行的所述字符信息；

根据单元行划分结果，从所有所述单元行中识别出所有标题信息；

确定所述标题信息的标题级别，生成包含所有所述标题信息的标题结构树。

2.根据权利要求1所述的方法，其特征在于，所述获取PDF文档中所有的字符信息的步骤，包括：

对PDF文档进行文档内容解析；

根据解析结果，获取PDF文档中所有的所述字符信息。

3.根据权利要求1所述的方法，其特征在于，所述获取每个所述字符信息的属性信息，所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等的步骤，包括：

在文档页面建立二维坐标系，所述二维坐标系包括沿页面宽度方向的X轴和沿页面高度方向的Y轴；

根据所述二维坐标系获取所述字符信息的X轴坐标、Y轴坐标、字符尺寸，以及，从字体库中匹配所述字符信息的字体样式，从而获取每个所述字符信息的所述属性信息。

4.根据权利要求1所述的方法，其特征在于，根据所述属性信息，将所述字符信息划分成多个单元行，每个所述单元行顺序包含所有位于文档同一行的所述字符信息的步骤，包括：

对每个页面的所述字符信息，按照Y轴坐标的大小进行一次排序；

根据所述一次排序的结果，对Y轴坐标相同的所述字符信息，按照X轴坐标的大小进行二次排序；

根据所述二次排序的结果，将Y轴坐标数值相同的所述字符信息，划分为所述单元行。

5.根据权利要求1所述的方法，其特征在于，所述根据单元行划分结果，从所有所述单元行中识别出所有标题信息的步骤，包括：

根据所述字符信息的字符尺寸，判断所述单元行中是否包含字符尺寸最小的字符信息；

如果否，则判断所述单元行是否以序号开头和/或使用加粗字体；

如果是，则判断所述单元行在序号处以外，是否还包含标点符号；

如果否，则判断所述单元行相邻的前一个所述单元行和后一个所述单元行中的字符信息是否均占满整行；

如果否，则判断所述单元行的所述字符信息的起始X轴坐标和终止X轴坐标是否在预设坐标范围内；

如果是，则将所述单元行识别为标题信息。

6.根据权利要求1所述的方法，其特征在于，所述确定所述标题信息的标题级别，生成包含所有所述标题信息的标题结构树的步骤，包括：

将无序号且字符尺寸最大的所述标题信息，确定为一级标题，所述一级标题为级别最

高的标题；

在所述标题信息中，确定有序号的所述标题信息的标题级别；

将除所述一级标题以外，无序号的所述标题信息确定为最低级别的标题；

根据所述标题信息在文档中的位置和所述标题信息的级别，确定所述标题信息的父子关系；

根据所述父子关系，生成所述标题结构树。

7.根据权利要求1所述的方法，其特征在于，所述确定所述标题信息的标题级别，生成包含所有所述标题信息的标题结构树的步骤之后，还包括：

从PDF文档中获取所述标题结构树中的每个所述标题信息的页码信息；

根据所述标题结构树和所述页码信息，生成PDF文档的目录。

8.一种PDF文档标题结构树生成装置，其特征在于，所述装置包括：

第一获取单元，用于获取PDF文档中所有的字符信息；

第二获取单元，用于获取每个所述字符信息的属性信息，每个所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等；

第一生成单元，用于根据所述属性信息，将所述字符信息划分成多个单元行，每个所述单元行顺序包含所有位于文档同一行的所述字符信息；

识别单元，用于根据单元行划分结果，从所有所述单元行中识别出所有标题信息；

第二生成单元，用于确定所述标题信息的标题级别，生成包含所有所述标题信息的标题结构树。

9.一种PDF文档标题结构树生成终端，其特征在于，所述终端包括：存储器和处理器；

所述存储器用于存储处理器可执行的程序；

所述处理器被配置为：

获取PDF文档中所有的字符信息；

获取每个所述字符信息的属性信息，每个所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等；

根据所述属性信息，将所述字符信息划分成多个单元行，每个所述单元行顺序包含所有位于文档同一行的所述字符信息；

根据单元行划分结果，从所有所述单元行中识别出所有标题信息；

确定所述标题信息的标题级别，生成包含所有所述标题信息的标题结构树。

10.一种PDF文档标题结构树生成系统，其特征在于，所述系统包括：服务器和用户终端；

所述服务器包括接收模块、处理模块和发送模块；

所述接收模块，用于从所述用户终端接收PDF文档；

所述处理模块，用于获取PDF文档中所有的字符信息；

以及，用于获取每个所述字符信息的属性信息，每个所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等；

以及，用于根据所述属性信息，将所述字符信息划分成多个单元行，每个所述单元行顺序包含所有位于文档同一行的所述字符信息；

以及，用于根据单元行划分结果，从所有所述单元行中识别出所有标题信息；

以及，用于确定所述标题信息的标题级别，生成包含所有所述标题信息的标题结构树；

所述发送模块，用于将所述标题结构树发送至所述用户终端；

所述用户终端，用于向所述服务器发送PDF文档，以及，用于从所述服务器接收所述PDF文档的所述标题结构树。

一种PDF文档标题结构树生成方法、装置、终端及系统

技术领域

[0001] 本发明涉及文字信息处理领域，尤其涉及一种PDF文档标题结构树生成方法、装置、终端及系统。

背景技术

[0002] 便携式文档格式(英语:Portable Document Format,简称PDF)是电子设备中常用的呈现文档的文件格式，每个PDF文档包含固定布局的平面文档的完整描述，包括文本、字形、图形及其他需要显示的信息。PDF文档的内容经常是一篇文章、一本书籍等，因此，在PDF文档中，文档的内容按照文章的结构、书籍的章节等具有不同的层级，每个层级的内容在开头处通常具有与内容层级对应的标题。

[0003] 由于，文档的标题通常与文档的内容相对应，所以，文档的标题结构通常能够体现文档的内容结构，因此，在一些文档数据管理系统中，通常通过展现标题结构或提供标题内容检索的方式，为用户提供文档结构预览或文档内容检索。在现有技术中，通常通过识别PDF文档目录的方式获取到文档的标题结构，然而，有些文档并不包含目录，就无法通过识别PDF文档目录的方式获取到文档的标题结构。

[0004] 因此，对于PDF文档，尤其是对于不包含目录的PDF文档，如何PDF文档中提取文档的标题结构成为本领域技术人员亟待解决的问题。

发明内容

[0005] 本发明提供了一种PDF文档标题结构树生成方法、装置、终端及系统，以解决现有技术中存在的问题。

[0006] 第一方面，本发明实施例提供了一种PDF文档标题结构树生成方法，所述方法包括:获取PDF文档中所有的字符信息;获取每个所述字符信息的属性信息，所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等;根据所述属性信息，将所述字符信息划分成多个单元行，每个所述单元行顺序包含所有位于文档同一行的所述字符信息;根据单元行划分结果，从所有所述单元行中识别出所有标题信息;确定所述标题信息的标题级别，生成包含所有所述标题信息的标题结构树。

[0007] 第二方面，本发明实施例提供了一种PDF文档标题结构树生成装置，所述装置包括:第一获取单元，用于获取PDF文档中所有的字符信息;第二获取单元，用于获取每个所述字符信息的属性信息，每个所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等;第一生成单元，用于根据所述属性信息，将所述字符信息划分成多个单元行，每个所述单元行顺序包含所有位于文档同一行的所述字符信息;识别单元，用于根据单元行划分结果，从所有所述单元行中识别出所有标题信息;第二生成单元，用于确定所述标题信息的标题级别，生成包含所有所述标题信息的标题结构树。

[0008] 第三方面，本发明实施例提供了一种PDF文档标题结构树生成终端，所述终端包括:存储器和处理器;所述存储器用于存储处理器可执行的程序;所述处理器被配置为:获

取PDF文档中所有的字符信息;获取每个所述字符信息的属性信息,每个所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等;根据所述属性信息,将所述字符信息划分成多个单元行,每个所述单元行顺序包含所有位于文档同一行的所述字符信息;根据单元行划分结果,从所有所述单元行中识别出所有标题信息;确定所述标题信息的标题级别,生成包含所有所述标题信息的标题结构树。

[0009] 第四方面,本发明实施例提供了一种PDF文档标题结构树生成系统,所述系统包括:服务器和用户终端;所述服务器包括接收模块、处理模块和发送模块;所述接收模块,用于从所述用户终端接收PDF文档;所述处理模块,用于获取PDF文档中所有的字符信息;以及,用于获取每个所述字符信息的属性信息,每个所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等;以及,用于根据所述属性信息,将所述字符信息划分成多个单元行,每个所述单元行顺序包含所有位于文档同一行的所述字符信息;以及,用于根据单元行划分结果,从所有所述单元行中识别出所有标题信息;以及,用于确定所述标题信息的标题级别,生成包含所有所述标题信息的标题结构树;所述发送模块,用于将所述标题结构树发送至所述用户终端;所述用户终端,用于向所述服务器发送PDF文档,以及,用于从所述服务器接收所述PDF文档的所述标题结构树。

[0010] 本发明实施例提供的技术方案,为了解决从PDF文档中提取文档的标题结构的问题,首先,获取PDF文档中所有的字符信息,所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等;然后,获取每个所述字符信息的属性信息;其次,根据所述属性信息,将所述字符信息划分成多个单元行,每个所述单元行顺序包含所有位于文档同一行的所述字符信息;再次,根据单元行划分结果,从所有所述单元行中识别出所有标题信息;最后,确定所述标题信息的标题级别,生成包含所有所述标题信息的标题结构树,从而实现了从非标准的PDF文档中获得文档的标题结构,解决了现有技术中无法从PDF文档中提取文档标题结构的问题。

附图说明

[0011] 为了更清楚地说明本发明的技术方案,下面将对实施例中所需要使用的附图作简单地介绍,显而易见地,对于本领域普通技术人员而言,在不付出创造性劳动性的前提下,还可以根据这些附图获得其他的附图。

[0012] 图1为本发明实施例提供的一种PDF文档标题结构树生成方法的流程图;

[0013] 图2为本发明实施例提供的一种PDF文档标题结构树生成方法步骤S110的流程图;

[0014] 图3为本发明实施例提供的一种PDF文档标题结构树生成方法步骤S120的流程图;

[0015] 图4为本发明实施例提供的一种PDF文档标题结构树生成方法步骤S130的流程图;

[0016] 图5为本发明实施例提供的一种PDF文档标题结构树生成方法步骤S140的流程图;

[0017] 图6为本发明实施例提供的一种PDF文档标题结构树生成方法步骤S150的流程图;

[0018] 图7为本发明实施例生成的一种PDF文档标题结构树的示意图;

[0019] 图8为本发明实施例提供的一种PDF文档标题结构树生成装置的框图;

[0020] 图9为本发明实施例提供的一种PDF文档标题结构树生成终端的结构框图;

[0021] 图10为本发明实施例提供的一种PDF文档标题结构树生成系统的结构框图。

具体实施方式

[0022] 为了使本技术领域的人员更好地理解本发明中的技术方案，下面将结合本发明实施例中的附图，对本发明实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例仅仅是本发明一部分实施例，而不是全部的实施例。基于本发明中的实施例，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例，都应当属于本发明保护的范围。

[0023] 实施例一

[0024] 本发明实施例提供了一种PDF文档标题结构树生成方法。图1为本发明实施例提供的一种PDF文档标题结构树生成方法的流程图，如图1所示，所述方法可以包括以下步骤：

[0025] 在步骤S110中，获取PDF文档中所有的字符信息。

[0026] 图2为本发明实施例提供的一种PDF文档标题结构树生成方法步骤S110的流程图，参见图2，本实施例的步骤S110包括以下步骤：

[0027] 在步骤S111中，对PDF文档进行文档解析。

[0028] 本实施例中，使用文字识别技术，分别对PDF文档的每个页面进行文档内容解析。

[0029] 示例地，本实施例使用Apache PDFbox工具库对PDF文档进行文档内容解析，Apache PDFBox工具库是一个用来处理PDF文档的Java工具库，Apache PDFbox工具库能够从PDF文档中解析到PDF文档每个页面中所包含字符的UNICODE编码和图片信息。

[0030] 在步骤S112中，根据解析结果，获取PDF文档中所有的所述字符信息。

[0031] 本实施例中，根据步骤S11中使用Apache PDFbox工具库对PDF文档解析到的字符的UNICODE编码，从PDF文档中获取到与UNICODE编码对应的字符信息。

[0032] 在步骤S120中，获取每个所述字符信息的属性信息，所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等。

[0033] 由于在PDF文档中，标题中字符的字符位置、字符尺寸、以及字体样式与PDE文档正文不同，PDF文档中每个字符信息的属性信息也不同，因此，属性信息能够用作从PDF文档中识别出标题信息。

[0034] 图3为本发明实施例提供的一种PDF文档标题结构树生成方法步骤S120的流程图，参见图3，本实施例的步骤S120包括以下步骤：

[0035] 在步骤S121中，在文档页面建立二维坐标系，所述二维坐标系包括沿页面宽度方向的X轴和沿页面高度方向的Y轴。

[0036] 由于在步骤S110中，已经通过Apache PDFbox工具库获取到了PDF文档中所有的字符信息；并且，在PDF文档中，每个字符信息的位置是固定不变的；因此，本步骤可以通过在文档页面建立二维坐标系的方式，实现对文档页面中的所有字符信息的位置进行参数化表示；从而，根据参数化表示结果，确定字符信息的属性信息。

[0037] 示例地，以A4大小篇幅的PDF文档为例，其页面尺寸为宽210mm×长297mm，按照打印质量的分辨率标准，设定文档页面的dpi＝300，得到A4大小篇幅的PDF文档页面像素大小为2479×3508；然后，以文档页面左上角第一个像素点作为原点坐标(0,0)，以水平方向为X轴，X轴正方向为水平向右方向，以竖直方向为Y轴，Y轴正方向为竖直向下方向，建立二维坐标系，以每个像素点的宽度作为单位刻度值，即在二维坐标系中，每个像素点的宽度为1，从

而，实现对文档页面中的所有字符信息的位置进行参数化表示。

[0038] 在步骤S122中，根据所述二维坐标系获取所述字符信息的X轴坐标、Y轴坐标、字符尺寸，以及，从字体库中匹配所述字符信息的字体样式，从而获取每个所述字符信息的所述属性信息。

[0039] 示例地，以步骤S121中建立的二维坐标系为例，对每个字符信息，在二维坐标系中，沿坐标轴正方向，设定每个字符信息的水平方向起始坐标点为x1，水平方向终止坐标点为x2，竖直方向起始坐标点为y1，竖直方向终止坐标点为y2，由此得出：

[0040] X轴坐标：本实施例以每个字符的x1值作为字符的X轴坐标；

[0041] Y轴坐标：本实施例以每个字符的y1值作为字符的Y轴坐标；

[0042] 字符坐标：本实施例中以(x1，y1)作为字符信息的字符坐标

[0043] 字符高度：本实施例以每个字符y2-y1的值作为字符的字符高度；

[0044] 字符宽度：本实施例以每个字符x2-x1的值作为字符的字符宽度；

[0045] 字符尺寸：本实施例以每个字符的(x2-x1，y2-y1)作为字符的字符尺寸。

[0046] 此外，本实施例中，通过识别字符信息的字迹所覆盖的坐标信息，能够得到字符信息的字体特征数据，根据字体特征数据在字体库中进行字体样式匹配，能够得到字符信息的字体样式，本实施例中的字体样式包括：字体名称、加粗字体、倾斜字体和划线字体等。

[0047] 示例地，对于示例字符：

(200)

[0048] (256) 例 (352)

(296)

[0049] 上述示例字符“例”，括号中的数值为该字符的坐标点x1、x2、y1和y2的值，括号的方位与坐标点在坐标轴中的位置相对应，其中x1＝256、x2＝352、y1＝200、y2＝296，由此得出该字符的X轴坐标为256、Y轴坐标为200、字符坐标为(256，200)、字符宽度为96、字符高度为96、字符尺寸为(96，96)。

[0050] 在步骤S130中，根据所述属性信息，将所述字符信息划分成多个单元行，每个所述单元行顺序包含所有位于文档同一行的所述字符信息。

[0051] 由于，在普遍使用的文档排版方式中，文档的标题采用独占一行的方式，因此，本步骤将字符信息划分成多个单元行，每个单元行包括文档的一行内容，能够把PDF文档中的标题信息以单元行的形式划分出来。

[0052] 图4为本发明实施例提供的一种PDF文档标题结构树生成方法步骤S130的流程图，参见图4，本实施例的步骤S130包括以下步骤：

[0053] 在步骤S131中，对每个页面的所述字符信息，按照Y轴坐标的大小进行一次排序。

[0054] 由于在PDF文档中，每行内容以水平排列的方式呈现，因此，位于同一行的所有字符信息的Y轴坐标是相同的，且在本实施例提供的二维坐标系中，每一行内容的Y轴坐标值都会比上一行的Y轴坐标值大，Y轴坐标值的大小能够体现出文档各行内容的位置关系。

[0055] 示例地，某PDF文档的一个页面中包含以下内容：

[0056] 济南的冬天

[0057] 对于一个在北平住惯的人，像我，冬天要是不刮风……

[0058] ……

[0059] 上述文档页面中字符信息的字符坐标为：

[0060] 济(m1,n1)南(m2,n1)的(m3,n1)冬(m4,n1)天(m5,n1)

[0061] 对(k1,n2)于(k2,n2)一(k3,n2)个(k4,n2)在(k5,n2)北(k6,n2)平(k7,n2)住(k8,n2)惯(k9,n2)的(k10,n2)人(k11,n2)，(k12,n2)像(k13,n2)我(k14,n2)，(k15,n2)冬(k16,n2)天(k17,n2)要(k18,n2)是(k19,n2)不(k20,n2)刮(k21,n2)风(k22,n2)……

[0062] ……

[0063] 从上述示例文档中字符信息的字符坐标可以看出，在该PDF文档页面中，许多字符信息的Y轴坐标值相同，说明这些字符信息位于文档的同一行，本步骤中，将PDF文档页面中的字符信息按照Y轴坐标值由小到大排序，Y轴坐标值相同的字符信息在排序中位于同一个序列。

[0064] 在步骤S132中，根据所述一次排序的结果，对Y轴坐标相同的所述字符信息，按照X轴坐标的大小进行二次排序。

[0065] 本实施例中，由于二维坐标系的原点位于文档页面左上角，X轴正方向为从左到右，因此，对Y轴坐标相同的字符信息，按照坐标值从小到达的顺序进行排序。

[0066] 示例地，通过本实施例步骤S131和步骤S132对上述文档进行一次排序和二次排序之后，得到的二次排序结果为：

[0067] 济(m1,n1)南(m2,n1)的(m3,n1)冬(m4,n1)天(m5,n1)对(k1,n2)于(k2,n2)一(k3,n2)个(k4,n2)在(k5,n2)北(k6,n2)平(k7,n2)住(k8,n2)惯(k9,n2)的(k10,n2)人(k11,n2)，(k12,n2)像(k13,n2)我(k14,n2)，(k15,n2)冬(k16,n2)天(k17,n2)要(k18,n2)是(k19,n2)不(k20,n2)刮(k21,n2)风(k22,n2)……

[0068] 在步骤S133中，根据所述二次排序的结果，将Y轴坐标数值相同的所述字符信息，划分为所述单元行。

[0069] 示例地，针对上述二次排序结果，本步骤中划分单元行的结果为：

[0070] 单元行1：济南的冬天

[0071] 单元行2：对于一个在北平住惯的人，像我，冬天要是不刮风……

[0072] 单元行3：……

[0073] ……：……

[0074] 单元行N：……

[0075] 步骤S130中，通过对PDF文档的字符信息划分单元行，实现了对PDF文档的字符信息以单元行为识别单位进行整体识别，便于以单元行为识别单位从PDF文档中识别出标题信息。

[0076] 在步骤S140中，根据单元行划分结果，从所有所述单元行中识别出所有标题信息。

[0077] 本实施例中，分别对每个单元行识别标题信息，在对所有单元行进行识别之后，就能够得到PDF文档的全部标题信息。

[0078] 图5为本发明实施例提供的一种PDF文档标题结构树生成方法步骤S140的流程图，参见图5，本实施例的步骤S140包括以下步骤：

[0079] 在步骤S141中，根据所述字符信息的字符尺寸，判断所述单元行中是否包含字符尺寸最小的字符信息。

[0080] 在PDF文档中，标题的字体大小通常设置的比正文的字体大，即使标题字体的字号

和正文字体的字号相同，由于标题采用粗体字，其在二维坐标系中的大小也要大于正文的常规字体的大小，所以，正文的字体在整个文档中最小，因此，本步骤判断单元行中是否包含字符尺寸最小的字符信息，如果是，则说明该单元行中包含文档正文的字符信息，该单元行中的字符信息不是标题信息，如果否，则说明该单元行中的字符信息可能是标题信息。

[0081] 示例地，本实施例中通过字符尺寸(x2-x1,y2-y1)的值来判断单元行中是否包含字符尺寸最小的字符信息，如果单元行中出现字符的x2-x1和y2-y1均为最小值，则说明该单元行中包含字符尺寸最小的字符信息。

[0082] 在步骤S142中，如果否，则判断所述单元行是否以序号开头和/或使用加粗字体；

[0083] 在PDF文档中，为了使标题能够体现出文档的结构，标题会以序号开头，并使用加粗字体与正文加以区分，因此，本步骤在步骤S141的判断结果为否的情况下，判断单元行是否为以序号开头和/或使用加粗字体，如果判断结果为是，说明该单元行中的字符信息可能是标题信息，进入下一个判断步骤，如果判断结果为否，说明该单元行中的字符信息不是标题信息；需要注意的是，本步骤判断单元行是否使用加粗字体，必须以整个单元行为判断单位，当整个单元行中的所有字符信息均为加粗字体时，才能认为该单元行使用加粗字体，如果该单元行仅有部分字符信息使用加粗字体，则不可以认为该单元行使用加粗字体。

[0084] 在步骤S143中，如果是，则判断所述单元行在除序号处以外，是否还包含标点符号。

[0085] 根据文档标题撰写规范，文档的标题不应包含标点符号，因此，在步骤S142判断结果为是的情况下，本步骤通过判断单元行在除序号处以外，是否还包含标点符号的方式，进一步确定单元行中的字符信息是标题信息的可能性，排除单元行中的序号是文档正文中出现的引用标题内容的情况、以及排除加粗字体是文档正文中为强调某些内容而使用加粗字体的情况；本步骤对单元行的判断过程为：如果判断结果为否，则说明该单元行中的字符信息可能是标题信息，进入下一个判断步骤；如果判断结果为是，则说明该单元行中的字符信息不是标题信息。

[0086] 示例地，某单元行的内容为：

[0087] 古老的济南，城里那么狭窄，城外又那么宽敞

[0088] 本实施例的步骤S142和S143对上述示例内容的判断过程为：在步骤S142中，判断所述单元行是否以序号开头和/或使用加粗字体的结果为是，因此在步骤S143中，判断所述单元行在序号处以外，是否还包含标点符号，由于该单元行包含逗号，但并不包含序号，因此，步骤S143的判断结果为是，说明该单元行中的字符信息不是标题信息。

[0089] 在步骤S144中，如果否，则判断所述单元行相邻的前一个所述单元行和后一个所述单元行中的字符信息是否均占满整行。

[0090] 在PDF文档中，标题的前一行是上文的段尾末行或者上一级父标题，标题的后一行是下文的段首起始行或者下一级子标题。由于段尾末行字数的原因，段尾末行通常不会出现占满整行的现象；由于标题的字数有限，标题也不会占满整行，此外，由于段落内容在段首处要空出两个文字字符的位置，因此，段首起始行也不会出现字符信息占满整行的现象。总之，无论标题的前一行是上文的段尾末行，还是上一级父标题，以及，无论标题的后一行是下文的段首起始行，还是下一级子标题，标题所在单元行相邻的前一个单元行和后一个单元行均不会同时出现字符信息占满整行的现象。因此，如果本步骤的判断结果为否，则说

明该单元行中的字符信息可能是标题信息，进入下一个判断步骤；如果判断结果为是，则说明该单元行中的字符信息不是标题信息。

[0091] 需要补充说明的是，当单元行中的字符信息是文档正文某段内容中的一行内容时，通常会出现单元行相邻的前一个单元行和后一个单元行中的字符信息是否均占满整行的现象。

[0092] 在步骤S145中，如果否，则判断所述单元行的所述字符信息的起始X轴坐标和终止X轴坐标是否在预设坐标范围内。

[0093] 在PDF文档中，每一级标题都以固定的格式出现在文档中段首的指定位置，但是，在一些文档中，一些文档内容以注释或者批注的形式出现在文档的其他位置，例如文档正文的左侧或右侧，为了防止这些内容被误识别成标题，本步骤在对文档页面建立的二维坐标系的基础上，根据文档正文在坐标系中的坐标范围，设定限定标题位置坐标的预设坐标范围(Xmin,Xmax)，通过判断单元行中字符信息的起始X轴坐标和终止X轴坐标是否在预设坐标范围内的方式，确定单元行中的字符信息是否是标题信息。

[0094] 需要补充说明的是，在文档页面中，有时还包括页眉、页脚和页码，页眉、页脚和页码通常位于文档页面的顶部或底部，并且，页眉、页脚和页码中不涉及文档标题和正文信息，因此，为了排除单元行中的页眉、页脚和页码，步骤S145还可以是：如果否，则判断所述单元行的所述字符信息的起始X轴坐标和终止X轴坐标是否在第一预设坐标范围内，以及所述单元行的所述字符信息的起始Y轴坐标和终止Y轴坐标是否在第二预设坐标范围内。其中，第一预设坐标范围为(Xmin,Xmax)，第二预设坐标范围为(Ymin,Ymax)，(Xmin,Xmax)和(Ymin,Ymax)共同限定了文档页面中的一块矩形区域，该矩形区域内只包含文档标题和正文。

[0095] 在步骤S146中，如果是，则将所述单元行识别为标题信息。

[0096] 本实施例的步骤S140，实现了从单元行中识别出PDF文档全部的标题信息，根据识别出的PDF文档全部的标题信息，可以生成PDF文档的标题结构树。

[0097] 需要补充说明的是，本实施例在步骤S140中示出的步骤S141-步骤S145的顺序仅作为一种示例性的顺序，不是唯一顺序，在步骤S140中，步骤S141-S145的顺序可以任一排列，这是由于在步骤S141-步骤S145中，每一个步骤均为一个判断条件，当单元行满足步骤S141-步骤S145的所有判断条件时，步骤S146就会把该单元行识别为标题，与判断顺序无关。但是，本申请实施例中示出的步骤S141-步骤S145的顺序是本方法步骤S140的最优方案，能够减少步骤S140的工作量，提高步骤S140的识别速度和识别的准确性。具体为，在一篇文档中，正文的字符量远远大于标题的字符量，且正文的字符尺寸最小，因此，步骤S141根据字符信息的字符尺寸，判断单元行中是否包含字符尺寸最小的字符信息，可以直接过滤掉大量包含正文的单元行，缩小后续步骤的单元行处理量，然后，步骤S142根据标题通常使用序号和加粗字体的普遍特性，判断单元行是否以序号开头和/或使用加粗字体，属于对标题单元行普遍特性的判断；接下来，步骤S143和步骤S144判断单元行在序号处以外，是否还包含标点符号，以及判断单元行相邻的前一个单元行和后一个单元行中的字符信息是否均占满整行，属于对加粗字体单元行不属于标题时的极端情况的判断；最后，步骤S145是对单元行中字符信息位置的判断；因此本实施例中的步骤S141-S145先后从：缩小范围—普遍特性—极端情况—字符信息位置的四个方面，逐渐收敛地从所有单元行中识别出所有标题

信息，能够减少步骤S140的工作量，提高步骤S140的识别速度和识别的准确性。在步骤S150中，确定所述标题信息的标题级别，生成包含所有所述标题信息的标题结构树。

[0098] 图6为本发明实施例提供的一种PDF文档标题结构树生成方法步骤S150的流程图，参见图6，本实施例的步骤S150包括以下步骤：

[0099] 在步骤S151中，将无序号且字符尺寸最大的所述标题信息，确定为一级标题，所述一级标题为级别最高的标题。

[0100] 文档的一级标题为文档的题目，在一篇文档中，题目不带序号且字体为整篇文档中的最大字体，因此，本实施例在步骤S151中，将无序号且字符尺寸最大的所述标题信息，确定为一级标题。字符尺寸最大的确定条件为：当字符尺寸(x2-x1,y2-y1)中的值均为最大值时，确定字符尺寸最大。

[0101] 在步骤S152中，在所述标题信息中，确定有序号的所述标题信息的标题级别。

[0102] 在一篇文档中，标题的序号位于文档标题前，以数字和“.”的形式呈现，并根据文档的内容结构以数值递增和层级递增的方式显示。

[0103] 文档的标题的序号格式和标题级别的对应关系如下：

[0104]

序号格式	标题级别
1	二级标题
1.1	三级标题

[0105]

1.1.1	四级标题
1.1.1.1	五级标题
……	……

[0106] 因此，根据文档的标题的序号格式和标题级别的对应关系，能够确定有序号的标题信息的标题级别。在所有的标题级别中，相邻两个级别的标题具有父子关系，例如：三级标题是二级标题的子标题，四级标题是五级标题的父标题。

[0107] 需要说明的是，由于一级标题无序号，有序号的标题信息最高为二级标题。

[0108] 在步骤S153中，将除所述一级标题以外，无序号的所述标题信息确定为最低级别的标题；

[0109] 在一篇文档中，无序号的标题除了一级标题之外，还有在文档的标题中处于最低级别的叶子标题，叶子标题在文档中的作用包括：对文档内容某个要点的概括、对文档内容的分步概括等，这些标题所概括的内容不足以构成文档的章节，因此不带有序号。本实施例在步骤S153中，将除所述一级标题以外，所有不带序号的标题信息确定为最低级别的标题，即确定为叶子标题，叶子标题在标题结构树中位于标题结构树的末端。

[0110] 需要说明的是，有些标题虽然不带有能够表示文档章节的序号，但是会包含标号，例如：①、(1)等，这些标号不是以数字和“.”的形式呈现，因此，本实施例不会将标号识别成序号，这类标题属于无序号的标题，所以，本实施例中，这类带有标号的标题信息会被确定为叶子标题。

[0111] 在步骤S154中，根据所述标题信息在文档中的位置和所述标题信息的级别，确定所述标题信息的父子关系。

[0112] 本实施例中，首先，按照标题信息在文档中位置的先后顺序，对提取到的所有标题

信息排序。

[0113] 示例地，本实施例中对所有标题信息的排序结果为：

[0114] 文档标题

[0115] 1 标题信息1

[0116] 1.1 标题信息2

[0117] 1.1.1 标题信息3

[0118] 1.1.2 标题信息4

[0119] 1.2 标题信息5

[0120] 2 标题信息6

[0121] 2.1 标题信息7

[0122] 根据排序结果，按照以下规则确定标题信息的父子关系：

[0123] 1、相同级别的标题信息按照文档中的位置先后，为并列关系；

[0124] 2、出现在当前标题信息之前，与当前标题信息位置最接近的上一级标题信息，是当前标题信息的父级标题。

[0125] 示例地，根据本实施例对所有标题信息的排序结果，1.1.1和1.1.2为并列关系，且1.1为1.1.1和1.1.2的父级标题，以此类推，1.1和1.2为并列关系，1为1.1和1.2的父级标题。

[0126] 根据以上规则，得到的标题信息的父子关系为(以不同的缩进量表示)：

[0127] 文档标题

[0128] 1 标题信息1

[0129] 1.1 标题信息2

[0130] 1.1.1 标题信息3

[0131] 1.1.2 标题信息4

[0132] 1.2 标题信息5

[0133] 2 标题信息6

[0134] 2.1 标题信息7

[0135] 在步骤S155中，根据所述父子关系，生成所述标题结构树。

[0136] 本实施例生成的标题结构树如图7所示。

[0137] 本实施例在步骤S150之后，还可以包括步骤S160和步骤S170。

[0138] 在步骤S160中，从PDF文档中获取所述标题结构树中的每个所述标题信息的页码信息。

[0139] 本实施例中，页码信息可以通过对PDF文档页眉或页脚进行文字识别后获得。

[0140] 在步骤S170中，根据所述标题结构树和所述页码信息，生成PDF文档的目录。

[0141] 本发明实施例提供的技术方案，为了解决从PDF文档中提取文档的标题结构的问题，首先，获取PDF文档中所有的字符信息，所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等；然后，获取每个所述字符信息的属性信息；其次，根据所述属性信息，将所述字符信息划分成多个单元行，每个所述单元行顺序包含所有位于文档同一行的所述字符信息；再次，根据单元行划分结果，从所有所述单元行中识别出所有标题信息；最后，确定所述标题信息的标题级别，生成包含所有所述标题信息的标题结构树，从而实现

了从非标准的PDF文档中获得文档的标题结构，解决了现有技术中无法从PDF文档中提取文档标题结构的问题。

[0142] 实施例二

[0143] 本发明实施例提供了一种PDF文档标题结构树生成装置。图8为本发明实施例提供的一种PDF文档标题结构树生成装置的框图，如图8所示，所述装置包括：

[0144] 第一获取单元210，用于获取PDF文档中所有的字符信息。

[0145] 第二获取单元220，用于获取每个所述字符信息的属性信息，每个所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等。

[0146] 第一生成单元230，用于根据所述属性信息，将所述字符信息划分成多个单元行，每个所述单元行顺序包含所有位于文档同一行的所述字符信息。

[0147] 识别单元240，用于根据单元行划分结果，从所有所述单元行中识别出所有标题信息。

[0148] 第二生成单元250，用于确定所述标题信息的标题级别，生成包含所有所述标题信息的标题结构树。

[0149] 本发明实施例提供的技术方案，为了解决从PDF文档中提取文档的标题结构的问题，首先，获取PDF文档中所有的字符信息，所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等；然后，获取每个所述字符信息的属性信息；其次，根据所述属性信息，将所述字符信息划分成多个单元行，每个所述单元行顺序包含所有位于文档同一行的所述字符信息；再次，根据单元行划分结果，从所有所述单元行中识别出所有标题信息；最后，确定所述标题信息的标题级别，生成包含所有所述标题信息的标题结构树，从而实现了从非标准的PDF文档中获得文档的标题结构，解决了现有技术中无法从PDF文档中提取文档标题结构的问题。

[0150] 实施例三

[0151] 本发明实施例提供了一种PDF文档标题结构树生成终端。图9为本发明实施例提供的一种PDF文档标题结构树生成终端的结构框图，如图9所示，所述终端包括：存储器310和处理器320；

[0152] 所述存储器310用于存储处理器320可执行的程序；

[0153] 所述处理器320被配置为：

[0154] 获取PDF文档中所有的字符信息；

[0155] 获取每个所述字符信息的属性信息，每个所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等；

[0156] 根据所述属性信息，将所述字符信息划分成多个单元行，每个所述单元行顺序包含所有位于文档同一行的所述字符信息；

[0157] 根据单元行划分结果，从所有所述单元行中识别出所有标题信息；

[0158] 确定所述标题信息的标题级别，生成包含所有所述标题信息的标题结构树。

[0159] 本发明实施例提供的技术方案，为了解决从PDF文档中提取文档的标题结构的问题，首先，获取PDF文档中所有的字符信息，所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等；然后，获取每个所述字符信息的属性信息；其次，根据所述属性信息，将所述字符信息划分成多个单元行，每个所述单元行顺序包含所有位于文档同一行

的所述字符信息；再次，根据单元行划分结果，从所有所述单元行中识别出所有标题信息；最后，确定所述标题信息的标题级别，生成包含所有所述标题信息的标题结构树，从而实现了从非标准的PDF文档中获得文档的标题结构，解决了现有技术中无法从PDF文档中提取文档标题结构的问题。

[0160] 实施例四

[0161] 本发明实施例提供了一种PDF文档标题结构树生成系统。图10为本发明实施例提供的一种PDF文档标题结构树生成系统的结构框图，如图10所示，所述系统包括：服务器410和用户终端420；

[0162] 所述服务器410包括接收模块411、处理模块412和发送模块413；

[0163] 所述接收模块411，用于从所述用户终端420接收PDF文档；

[0164] 所述处理模块412，用于获取PDF文档中所有的字符信息；

[0165] 以及，用于获取每个所述字符信息的属性信息，每个所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等；

[0166] 以及，用于根据所述属性信息，将所述字符信息划分成多个单元行，每个所述单元行顺序包含所有位于文档同一行的所述字符信息；

[0167] 以及，用于根据单元行划分结果，从所有所述单元行中识别出所有标题信息；

[0168] 以及，用于确定所述标题信息的标题级别，生成包含所有所述标题信息的标题结构树；

[0169] 所述发送模块413，用于将所述标题结构树发送至所述用户终端420；

[0170] 所述用户终端420，用于向所述服务器410发送PDF文档，以及，用于从所述服务器接收所述PDF文档的所述标题结构树。

[0171] 本实施例中，用户终端420可以是个人计算机，移动电话、平板设备、以及其他具有信息数据传输功能的设备。

[0172] 本发明实施例提供的技术方案，为了解决从PDF文档中提取文档的标题结构的问题，首先，获取PDF文档中所有的字符信息，所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等；然后，获取每个所述字符信息的属性信息；其次，根据所述属性信息，将所述字符信息划分成多个单元行，每个所述单元行顺序包含所有位于文档同一行的所述字符信息；再次，根据单元行划分结果，从所有所述单元行中识别出所有标题信息；最后，确定所述标题信息的标题级别，生成包含所有所述标题信息的标题结构树，从而实现了从非标准的PDF文档中获得文档的标题结构，解决了现有技术中无法从PDF文档中提取文档标题结构的问题。

[0173] 本发明可用于众多通用或专用的计算系统环境或配置中，例如：个人计算机、服务器计算机、手持设备或便携式设备、平板型设备、多处理器系统、基于微处理器的系统、置顶盒、可编程的消费电子设备、网络PC、小型计算机、大型计算机、包括以上任何系统或设备的分布式计算环境等等。

[0174] 本发明可以在由计算机执行的计算机可执行指令的一般上下文中描述，例如程序模块。一般地，程序模块包括执行特定任务或实现特定抽象数据类型的例程、程序、对象、组件、数据结构等等。也可以在分布式计算环境中实践本发明，在这些分布式计算环境中，由通过通信网络而被连接的远程处理设备来执行任务。在分布式计算环境中，程序模块可以

位于包括存储设备在内的本地和远程计算机存储介质中。

[0175] 需要说明的是，在本文中，诸如“第一”和“第二”等之类的关系术语仅仅用来将一个实体或者操作与另一个实体或操作区分开来，而不一定要求或者暗示这些实体或操作之间存在任何这种实际的关系或者顺序。而且，术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含，从而使得包括一系列要素的过程、方法、物品或者设备不仅包括那些要素，而且还包括没有明确列出的其他要素，或者是还包括为这种过程、方法、物品或者设备所固有的要素。

[0176] 本领域技术人员在考虑说明书及实践这里公开的发明后，将容易想到本发明的其它实施方案。本发明旨在涵盖本发明的任何变型、用途或者适应性变化，这些变型、用途或者适应性变化遵循本发明的一般性原理并包括本发明未公开的本技术领域中的公知常识或惯用技术手段。说明书和实施例仅被视为示例性的，本发明的真正范围和精神由下面的权利要求指出。

[0177] 应当理解的是，本发明并不局限于上面已经描述并在附图中示出的精确结构，并且可以在不脱离其范围进行各种修改和改变。本发明的范围仅由所附的权利要求来限制。

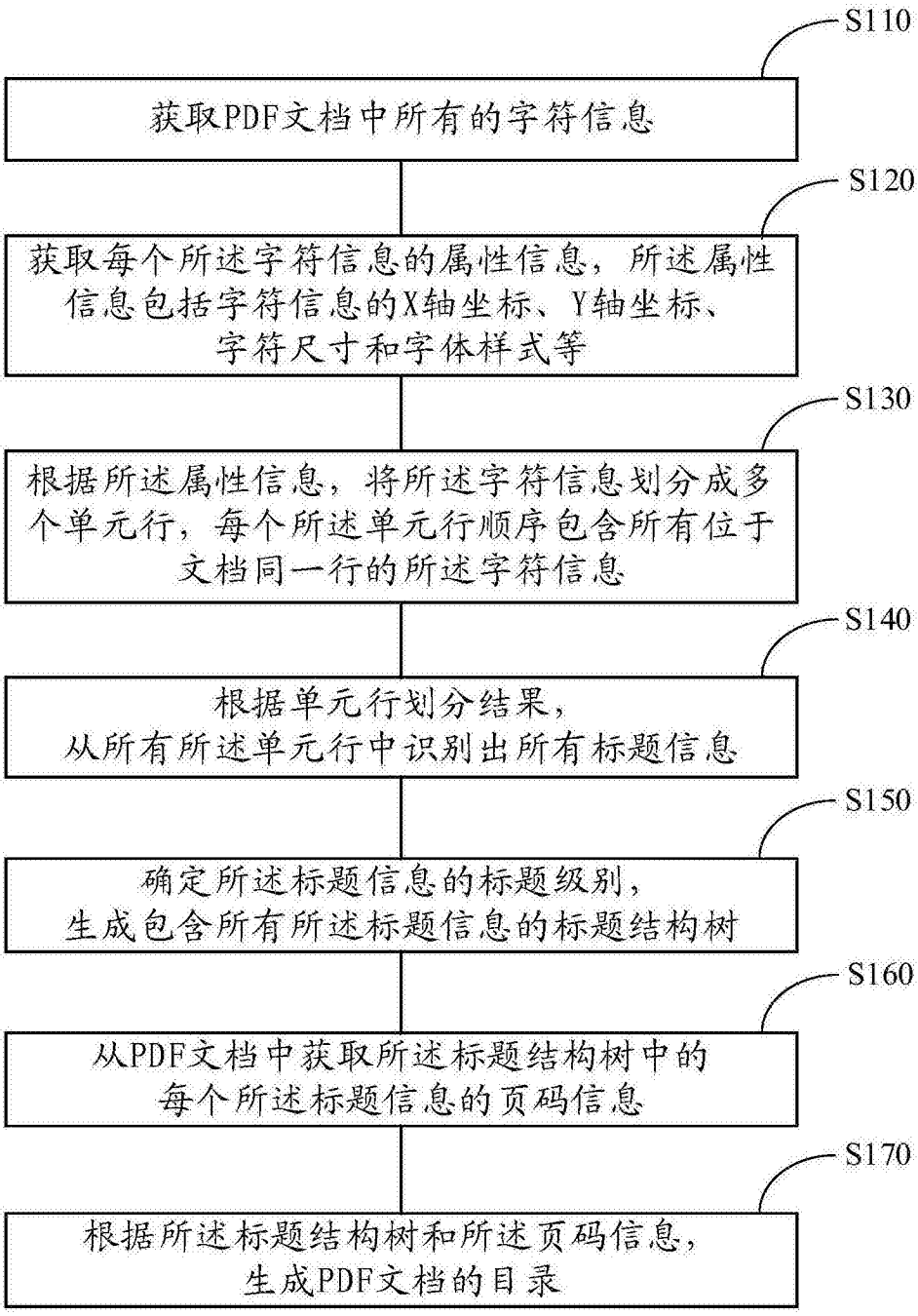

S110
获取PDF文档中所有的字符信息
S120
获取每个所述字符信息的属性信息，所述属性信息包括字符信息的X轴坐标、Y轴坐标、字符尺寸和字体样式等
S130
根据所述属性信息，将所述字符信息划分成多个单元行，每个所述单元行顺序包含所有位于文档同一行的所述字符信息
S140
根据单元行划分结果，从所有所述单元行中识别出所有标题信息
S150
确定所述标题信息的标题级别，生成包含所有所述标题信息的标题结构树
S160
从PDF文档中获取所述标题结构树中的每个所述标题信息的页码信息
S170
根据所述标题结构树和所述页码信息，生成PDF文档的目录

图1

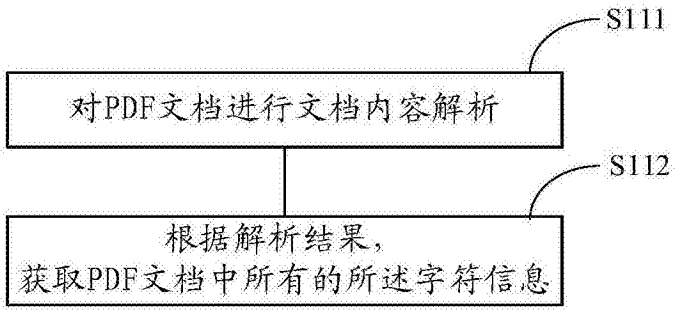

S111
对PDF文档进行文档内容解析
S112
根据解析结果，
获取PDF文档中所有的所述字符信息

图2

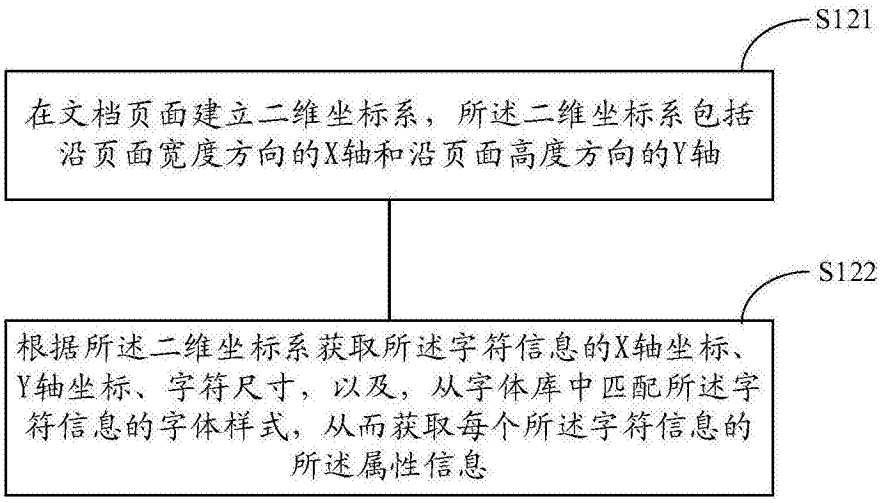

S121
在文档页面建立二维坐标系，所述二维坐标系包括沿页面宽度方向的X轴和沿页面高度方向的Y轴
S122
根据所述二维坐标系获取所述字符信息的X轴坐标、Y轴坐标、字符尺寸，以及，从字体库中匹配所述字符信息的字体样式，从而获取每个所述字符信息的所述属性信息

图3

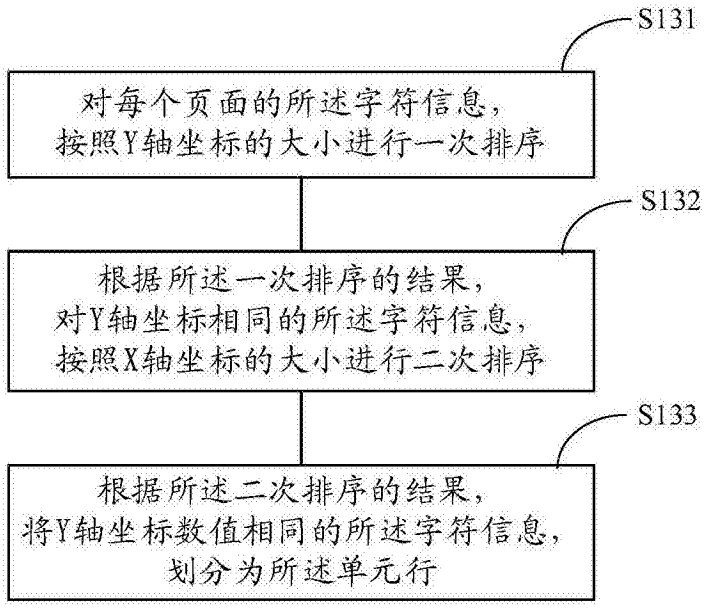

S131
对每个页面的所述字符信息，
按照Y轴坐标的大小进行一次排序
S132
根据所述一次排序的结果，
对Y轴坐标相同的所述字符信息，
按照X轴坐标的大小进行二次排序
S133
根据所述二次排序的结果，
将Y轴坐标数值相同的所述字符信息，
划分为所述单元行

图4

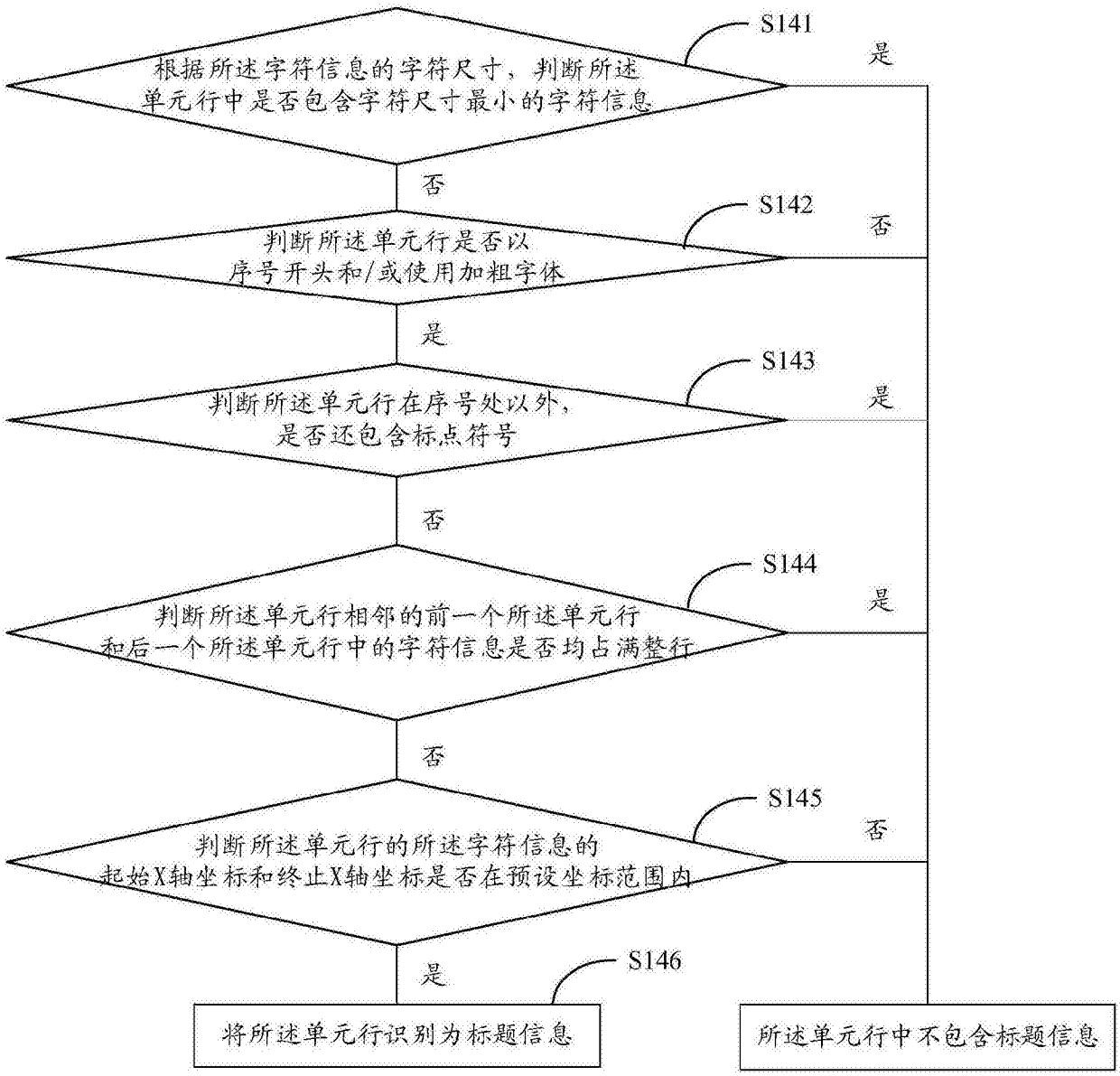

S141
根据所述字符信息的字符尺寸，判断所述单元行中是否包含字符尺寸最小的字符信息
是
否
S142
判断所述单元行是否以序号开头和/或使用加粗字体
否
是
S143
判断所述单元行在序号处以外，是否还包含标点符号
是
否
S144
判断所述单元行相邻的前一个所述单元行和后一个所述单元行中的字符信息是否均占满整行
是
否
S145
判断所述单元行的所述字符信息的起始X轴坐标和终止X轴坐标是否在预设坐标范围内
否
是
S146
将所述单元行识别为标题信息
所述单元行中不包含标题信息

图5

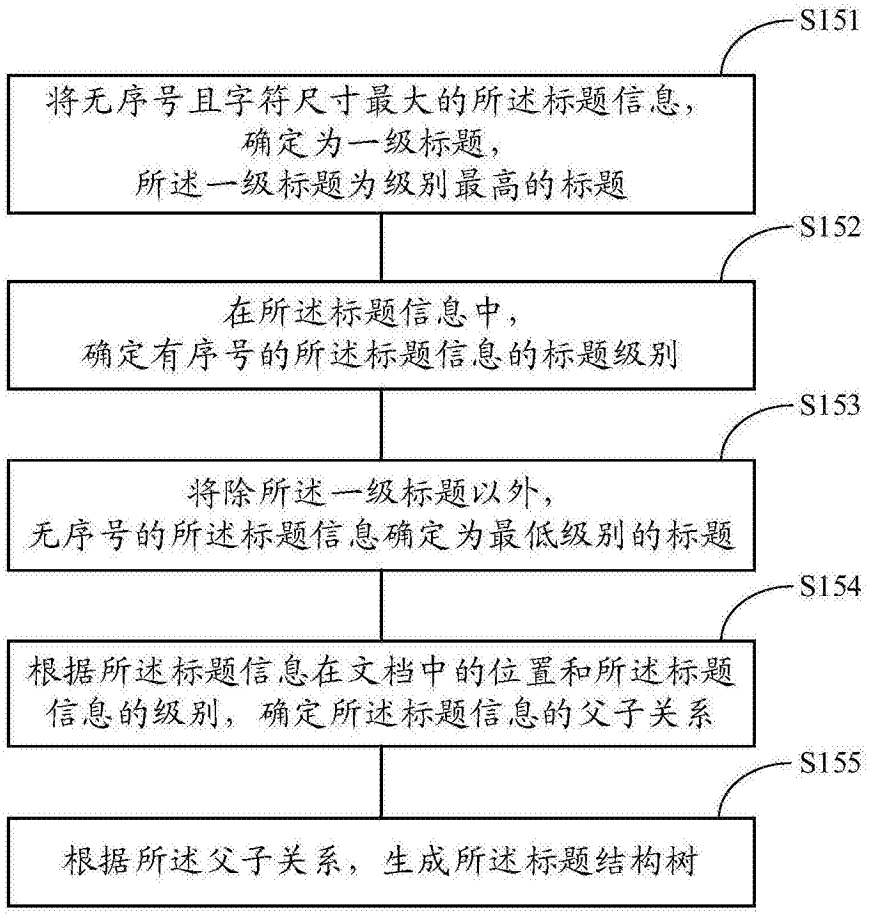

S151
将无序号且字符尺寸最大的所述标题信息，
确定为一级标题，
所述一级标题为级别最高的标题
S152
在所述标题信息中，
确定有序号的所述标题信息的标题级别
S153
将除所述一级标题以外，
无序号的所述标题信息确定为最低级别的标题
S154
根据所述标题信息在文档中的位置和所述标题信息的级别，确定所述标题信息的父子关系
S155
根据所述父子关系，生成所述标题结构树

图6

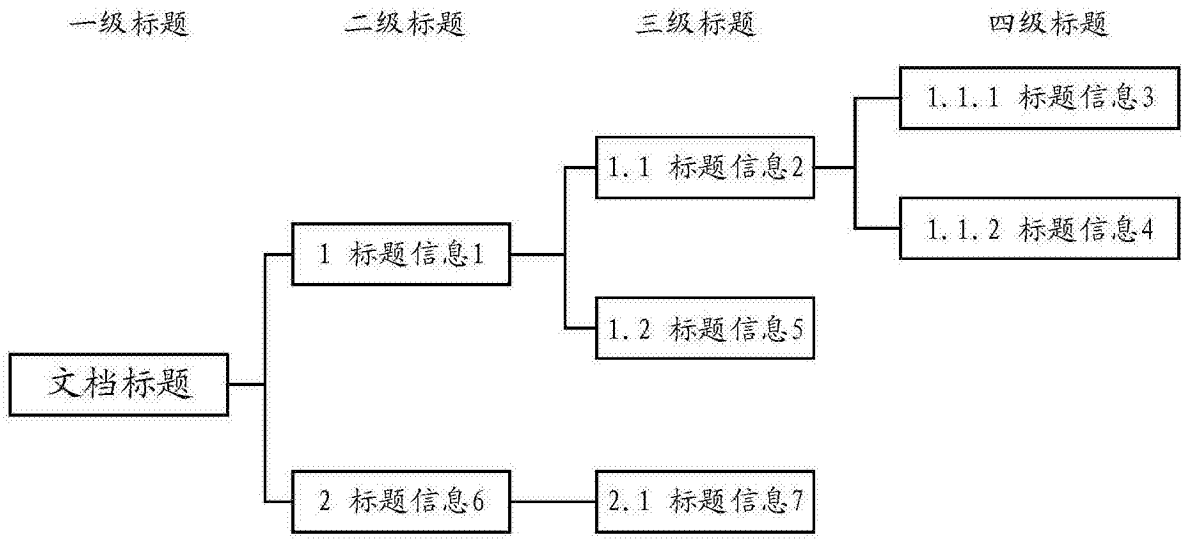

一级标题
二级标题
三级标题
四级标题
文档标题
1 标题信息1
2 标题信息6
1.1 标题信息2
1.2 标题信息5
2.1 标题信息7
1.1.1 标题信息3
1.1.2 标题信息4

图7

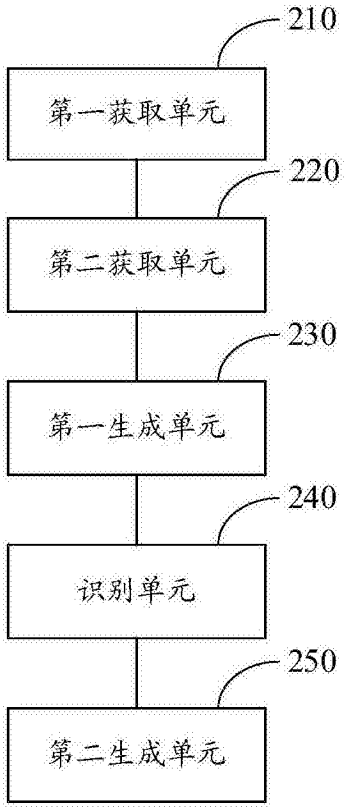

210
第一获取单元
220
第二获取单元
230
第一生成单元
240
识别单元
250
第二生成单元

图8

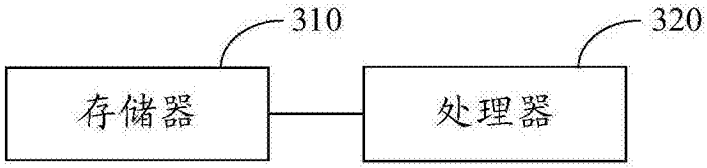

310
存储器
320
处理器

图9

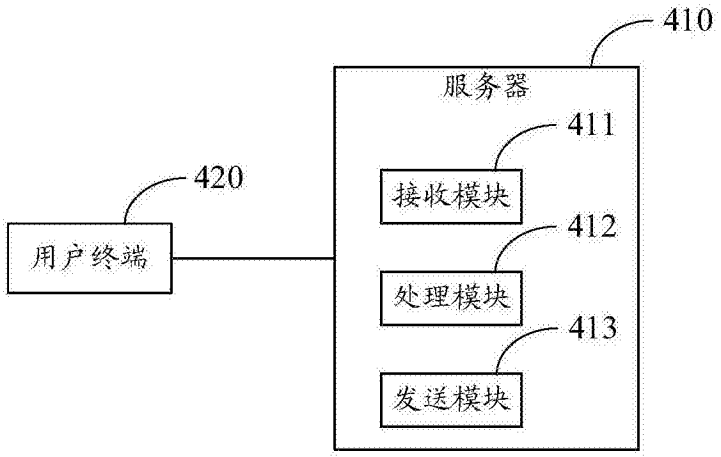

410
服务器
411
接收模块
412
处理模块
413
发送模块
420
用户终端

图10