

证书号第6923499号



发明专利证书

发明名称：集成式文档编辑器

发明人：伊莱·泽维

专利号：ZL 2018 8 0071870.4

专利申请日：2018年09月18日

专利权人：伊莱·泽维

地址：美国加利福尼亚州

授权公告日：2024年04月19日

授权公告号：CN 111492338 B

国家知识产权局依照中华人民共和国专利法进行审查，决定授予专利权，颁发发明专利证书并在专利登记簿上予以登记。专利权自授权公告之日起生效。专利权期限为二十年，自申请日起算。

专利证书记载专利权登记时的法律状况。专利权的转移、质押、无效、终止、恢复和专利权人的姓名或名称、国籍、地址变更等事项记载在专利登记簿上。



局长
申长雨

申长雨





证书号 第6923499号

专利权人应当依照专利法及其实施细则规定缴纳年费。本专利的年费应当在每年09月18日前缴纳。未按照规定缴纳年费的，专利权自应当缴纳年费期满之日起终止。

申请日时本专利记载的申请人、发明人信息如下：

申请人：

伊莱·泽维

发明人：

伊莱·泽维



(12) 发明专利

(10) 授权公告号 CN 111492338 B

(45) 授权公告日 2024. 04. 19

(21) 申请号 201880071870.4
 (22) 申请日 2018.09.18
 (65) 同一申请的已公布的文献号
 申请公布号 CN 111492338 A
 (43) 申请公布日 2020.08.04
 (85) PCT国际申请进入国家阶段日
 2020.05.06
 (86) PCT国际申请的申请数据
 PCT/US2018/051400 2018.09.18
 (87) PCT国际申请的公布数据
 W02019/055952 EN 2019.03.21
 (73) 专利权人 伊莱·泽维
 地址 美国加利福尼亚州
 (72) 发明人 伊莱·泽维
 (74) 专利代理机构 北京品源专利代理有限公司
 11332
 专利代理师 谭营营 王天鹏

(51) Int.Cl.
 G06F 3/0484 (2006.01)
 G06F 3/0488 (2006.01)
 G06F 3/0482 (2006.01)
 (56) 对比文件
 CN 101986249 A, 2011.03.16
 CN 102067130 A, 2011.05.18
 CN 102455862 A, 2012.05.16
 CN 103294657 A, 2013.09.11
 CN 103733172 A, 2014.04.16
 CN 105373309 A, 2016.03.02
 US 2004174399 A1, 2004.09.09
 US 2007070064 A1, 2007.03.29
 US 2015286395 A1, 2015.10.08
 US 2016011726 A1, 2016.01.14

审查员 李小婉

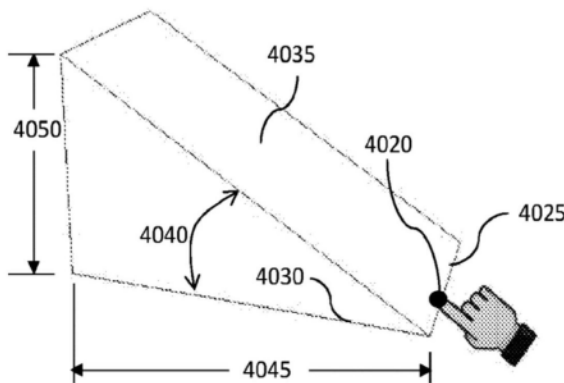
权利要求书4页 说明书27页 附图55页

(54) 发明名称

集成式文档编辑器

(57) 摘要

一种计算装置,包括:存储器,用于存储关于图形对象的信息,所述信息包括所述图形对象的位置;显示器,用于显示所述信息的至少一部分的表示;表面,用于检测对改变的指示;以及一个或多个处理单元。响应于检测到指示,一个或多个处理单元被配置为自动地改变:至少一个图形对象的几何形状,其中:所述至少一个图形对象的至少一个位置与至少一个其它图形对象的相同。所述改变包括所述至少一个图形对象的至少一个位置的改变,以及所述至少一个其它图形对象的至少一个位置不被改变;以及所述表示的至少一部分,其中,所述显示器被配置为显示改变后的表示的至少一部分。



1. 一种计算装置,包括:
存储器,用于存储关于图形对象的信息,所述信息包括所述图形对象的位置,
显示器,用于显示所述信息的至少一部分的表示,
表面,用于检测对改变的指示,以及
一个或多个处理单元;
响应于检测到所述指示,所述一个或多个处理单元被配置为自动地改变:
至少一个图形对象的几何形状,其中:
所述至少一个图形对象的至少一个位置与至少一个其它图形对象的相同,
所述改变包括所述至少一个图形对象的至少一个位置的改变,以及
所述至少一个其它图形对象的至少一个位置不被改变;以及
所述表示的至少一部分,
其中,所述显示器被配置为显示改变后的表示的至少一部分。
2. 根据权利要求1所述的计算装置,其中,在所述显示器上的表示的至少一部分是二维或三维的。
3. 根据权利要求1所述的计算装置,其中,所述关于图形对象的信息是矢量图形或CAD绘图数据。
4. 根据权利要求1所述的计算装置,其中,所述指示包括触摸手势、敲击手势、位置改变的指示或所述位置改变的方向的指示。
5. 根据权利要求1所述的计算装置,其中,所述一个或多个处理单元还被配置为识别命令。
6. 根据权利要求1所述的计算装置,其中,所述一个或多个处理单元被配置为:响应于如所述显示器上指示的,所述指示的至少一部分接近所述至少一个图形对象的一个或多个表面的至少一个位置或在所述至少一个图形对象的一个或多个表面的线、圆弧、拐角、表面或边缘内,而自动改变所述几何形状。
7. 根据权利要求5所述的计算装置,其中,所述一个或多个处理单元还被配置为:响应于检测到指示所显示部分的手势而自动识别所述至少一个图形对象的表示的所显示部分。
8. 根据权利要求7所述的计算装置,还被配置为致使所述显示器在改变所述几何形状之前或同时放大所显示部分,并且在已经改变了所述几何形状之后缩小所显示部分。
9. 根据权利要求1所述的计算装置,其中,所述至少一个其它图形对象的几何形状不被改变。
10. 根据权利要求1所述的计算装置,其中,所述一个或多个处理单元被配置为:响应于如所述显示器上指示的,检测到所述指示的至少一部分接近所述至少一个图形对象的几何特征或在其内,而自动识别指示所述几何形状的改变的所述至少一个图形对象的至少一个参数。
11. 根据权利要求1所述的计算装置,其中,所述显示器和与所述显示器一体的所述表面是触摸屏。
12. 根据权利要求5所述的计算装置,其中,所述命令包括:
用于改变所述至少一个图形对象的线的长度或角度中的至少一个的命令,并且其中,
所述一个或多个处理单元被配置为:响应于如所述显示器上指示的,检测到所述指示中的

位置的改变接近所述线或所述线的位置,而自动改变所述长度或角度;或

用于改变所述至少一个图形对象的表面的宽度、高度或角度中的至少一个的命令,并且其中,所述一个或多个处理单元被配置为:响应于如所述显示器上指示的,检测到所述指示接近所述表面的至少一个位置或边缘或者在所述表面内,而自动改变所述表面的宽度、高度或角度中的至少一个。

13. 根据权利要求5所述的计算装置,其中,所述命令包括:

用于将所述至少一个图形对象的直线改变为圆弧的命令,并且其中,所述一个或多个处理单元被配置为:响应于如所述显示器上指示的,检测到所述指示接近所述直线或在其内,而自动将所述直线改变为圆弧;

用于将所述至少一个图形对象的平坦表面改变为弧形表面的命令,并且其中,所述一个或多个处理单元被配置为:响应于如所述显示器上指示的,检测到所述指示接近所述平坦表面的至少一个位置或在所述平坦表面内,而自动将所述平坦表面改变为弧形表面;

用于改变所述至少一个图形对象的圆弧的半径的命令,并且其中,所述一个或多个处理单元被配置为:响应于如所述显示器上指示的,检测到所述指示内的位置的改变接近所述圆弧或在所述圆弧内,而自动改变所述圆弧的半径;或

用于改变所述至少一个图形对象的弧形表面的半径的命令,并且其中,所述一个或多个处理单元被配置为:如所述显示器上指示的,响应于检测到所述指示接近所述弧形表面或在所述弧形表面内,而自动改变所述弧形表面的半径。

14. 根据权利要求5所述的计算装置,其中,所述命令包括:

用于将圆角或圆弧添加到所述至少一个图形对象的拐角的命令,并且其中,所述一个或多个处理单元被配置为:响应于如所述显示器上指示的,检测到所述指示接近所述拐角或在所述拐角内,而自动将所述圆角或所述圆弧添加到所述拐角;或者

用于改变所述至少一个图形对象的圆形角的圆角或圆弧的半径和至少一个位置的命令,并且其中,所述一个或多个处理单元被配置为:响应于如所述显示器上指示的,检测到所述指示内的位置的改变接近所述圆角的至少一个位置或在所述圆角内,而自动改变所述圆形角的半径和至少一个位置。

15. 根据权利要求5所述的计算装置,其中,所述命令包括:

用于将倒角添加到所述至少一个图形对象的拐角的命令,并且其中,所述一个或多个处理单元被配置为响应于检测到指示接近所述拐角或在所述拐角内而自动添加所述倒角;或

用于改变经倒角的拐角的宽度、高度或角度中的至少一个的命令,并且其中,所述一个或多个处理单元被配置为:响应于如所述显示器上指示的,检测到所述指示内的位置的改变接近经倒角的拐角的至少一个位置或在经倒角的拐角内,而自动改变所述宽度、高度和角度中的至少一个。

16. 根据权利要求5所述的计算装置,其中,所述命令包括用于修剪所述至少一个图形对象的一部分的命令,并且其中,所述一个或多个处理单元被配置为:响应于所述显示器上指示的,检测到所述指示接近所述至少一个图形对象或在所述至少一个图形对象内而自动修剪所述至少一个图形对象的所述一部分。

17. 根据权利要求5所述的计算装置,其中,所述命令包括用于将分段添加到所述至少

一个图形对象的分段线对象的分段中或在所述至少一个图形对象的分段线对象的分段中进行几何改变的命令,并且其中,所述一个或多个处理单元被配置为:响应于如所述显示器上指示的检测到所述指示接近所述分段线对象的至少一个位置或在所述分段线对象内,而自动将所述分段添加到所述分段中或在所述分段中进行几何改变。

18. 一种计算方法,包括:

存储关于图形对象的信息,所述信息包括所述图形对象的位置;

在显示器上显示所述信息的至少一部分的表示;

检测对改变的指示;其中响应于检测到所述指示:

自动改变至少一个图形对象的几何形状,其中:

所述至少一个图形对象的至少一个位置与至少一个其它图形对象相同,

所述改变包括对所述至少一个图形对象的至少一个位置的改变,以及

所述至少一个其它图形对象的至少一个位置不被改变;

自动改变所述表示的至少一部分;以及

显示改变后的表示的至少一部分。

19. 根据权利要求18所述的计算方法,其中,在所述显示器上的所述表示的至少一部分是二维的或三维的。

20. 根据权利要求18所述的计算方法,其中,所述关于图形对象的信息是矢量图形或CAD绘图数据。

21. 根据权利要求18所述的计算方法,其中,所述指示包括触摸手势、敲击手势、位置改变的指示或所述位置改变的方向的指示。

22. 根据权利要求18所述的计算方法,还包括识别命令。

23. 根据权利要求18所述的计算方法,还包括:响应于所述指示的至少一部分接近所述至少一个图形对象的一个或多个表面的至少一个位置或在所述至少一个图形对象的一个或多个表面的线、圆弧、拐角、表面或边缘内,而自动改变所述几何形状。

24. 根据权利要求22所述的计算方法,还包括:响应于检测到指示所显示部分的手势而自动识别所述至少一个图形对象的表示的所显示部分。

25. 根据权利要求24所述的计算方法,还包括:在改变所述几何形状之前或同时放大所显示部分,并且在已经改变了所述几何形状后缩小所显示部分。

26. 根据权利要求18所述的方法,其中,所述至少一个其它图形对象的几何形状不被改变。

27. 根据权利要求18所述的计算方法,还包括:响应于检测到所述指示的至少一部分接近所述至少一个图形对象的几何特征或在所述至少一个图形对象的几何特征内,而自动识别指示所述几何形状的改变的所述至少一个图形对象的至少一个参数。

28. 根据权利要求22所述的计算方法,其中,所述命令包括:

用于改变所述至少一个图形对象的线的长度或角度中的至少一个的命令,并且其中响应于检测到指示内的位置的改变接近所述线或所述线的位置,而自动改变所述长度或角度;或

用于改变所述至少一个图形对象的表面的宽度、高度或角度中的至少一个的命令,并且其中响应于检测到所述指示接近所述表面的至少一个位置或边缘或者在所述表面内,而

自动改变所述表面的宽度、高度或角度中的至少一个。

29. 根据权利要求22所述的计算方法,其中,所述命令包括:

用于将所述至少一个图形对象的直线改变为圆弧的命令,并且其中响应于检测到所述指示接近所述直线或在所述直线内,而自动将所述直线改变为圆弧,

用于将所述至少一个图形对象的平坦表面改变为弧形表面的命令,并且其中响应于检测到所述指示接近所述平坦表面的至少一个位置或在所述平坦表面内,而自动将所述平坦表面改变为弧形表面;

用于改变所述至少一个图形对象的圆弧的半径的命令,并且其中响应于检测到指示内的位置的改变接近所述圆弧或在所述圆弧内,而自动改变所述圆弧的半径;或

用于改变所述至少一个图形对象的弧形表面的半径的命令,并且其中响应于检测到所述指示接近所述弧形表面或在所述弧形表面内,而自动改变所述弧形表面的半径。

30. 根据权利要求22所述的计算方法,其中,所述命令包括:

用于将圆角或圆弧添加到所述至少一个图形对象的拐角的命令,并且其中响应于检测到所述指示接近所述拐角或在所述拐角内而自动添加所述圆角或所述圆弧;或

用于改变所述至少一个图形对象的圆形角的圆角或圆弧的半径和至少一个位置的命令,并且其中响应于检测到所述指示接近所述圆角的至少一个位置或在所述圆角内而自动改变所述圆形角的半径和至少一个位置。

31. 根据权利要求22所述的计算方法,其中,所述命令包括:

用于将倒角添加到所述至少一个图形对象的拐角的命令,并且其中,响应于检测到所述指示接近所述拐角的至少一个位置或在所述拐角内而自动添加所述倒角;或

用于改变经倒角的拐角的宽度、高度或角度中的至少一个的命令,并且其中响应于检测到所述指示内的位置的改变接近经倒角的拐角的至少一个位置或在经倒角的拐角内,而自动改变所述宽度、高度和角度中的至少一个。

32. 根据权利要求22所述的计算方法,其中,所述命令包括用于修剪所述至少一个图形对象的一部分的命令,并且其中响应于检测到所述指示接近所述至少一个图形对象或在所述至少一个图形对象内而自动修剪所述至少一个图形对象的所述一部分。

33. 根据权利要求22所述的计算方法,其中,所述命令包括用于将分段添加到所述至少一个图形对象的分段线对象的分段中或在所述至少一个图形对象的分段线对象的分段中进行几何改变的命令,并且其中响应于检测到所述指示接近所述分段线对象的至少一个位置或在所述分段线对象内,而自动添加所述分段或在所述分段中进行几何改变。

集成式文档编辑器

[0001] 相关申请

[0002] 本申请要求于2017年9月15日提交的美国临时专利申请62/559,269的权益,其内容通过引用并入本文。

背景技术

[0003] 所公开的实施例涉及文档创建和编辑。更具体地,所公开的实施例涉及信息条目的识别与文档创建的集成。手写数据输入计算机程序是已知的。最广泛的使用是在个人数字助理装置中。由于各种原因,手写输入到使用键盘的装置并不广泛。例如,字符的转录和识别相对较慢,并且对于字符或命令输入还没有被广泛接受的标准。

发明内容

[0004] 根据所公开的实施例,提供了用于通过使用作为显示器一部分的数字化识别器(诸如数字化垫板、触摸屏或其他位置输入接收机构)将手写信息、特别是校正信息并入先前创建的可修改文本或图形文档(例如文本数据、图像数据或命令提示)中的方法和系统。在数据输入模式下,数据单元借由书写笔或类似的划线工具插入并接受放置在指定位置处、将书写笔的x-y位置与文档中的实际位置相关联、或通过模拟键盘击键(或通过运行代码/程序)访问文档存储器中的位置。在识别模式下,使用可选的嵌入编辑或其他命令将输入的数据识别为清晰的文本,并将其转换为机器可读格式。否则,数据被识别为图形(对于容纳图形的应用程序)并被接受到相关联的图像帧中。可以同时识别文本或图形表单的数据组合。在一个特定的实施例中,在最初调用数据输入模式之后,书写工具的位置存在误差窗口,使得工具的实际放置不是关键的,这是因为数据的输入通过书写笔的初始x-y位置与文档中的实际位置相关。此外,根据笔在文档中的位置,存在允许的误差(例如,相对于周围的数据)。在命令输入模式中,可以输入从各种应用程序共有的基本集中选择的手写符号,并且可以执行对应的命令。在特定实施例中,应用了不依赖于应用程序并且可以是用户直观的手写符号和/或命令的基本集合。此手写命令集允许进行修订和创建文档,而无需事先了解特定应用程序的命令。

[0005] 在特定实施例中,诸如与文字处理器一起使用时,当用户在文档中的指定位置处调用注释模式并且然后可以经由输入装置将手写信息输入到本机注释字段中时,可以实施所公开的实施例,然后将其转换为文本或图像或转换为要执行的命令数据,手写识别器同时或在完成手写信息单元输入后操作。然后,被识别为文本的信息会自动或根据单独的命令转换为密码并导入到文本主体中。然后,被识别为图形的信息会自动或根据单独的命令转换为图像数据(诸如本机图形格式或JPEG图像),并在指定点处导入到文本主体中。可以执行解释为命令的信息,诸如编辑命令,其控制文档中文本的添加、删除或移动,以及字体类型或大小改变或颜色改变。在进一步的特定实施例中,所公开的实施例可以被合并为用于文字处理器程序的插件模块,并且可以作为系统的一部分被调用,诸如使用宏或通过追踪修订(Track Changes)特征被调用。

[0006] 在替代实施例中,用户可以在调用识别模式之前手动指示输入的性质,无论输入是文本、图形还是命令,通过提供由程序提示用于设置首选符号并学习用户的手写模式的逐步协议,可以进一步改进识别。

[0007] 在所公开的实施例的至少一个方面中,一种计算装置,包括:存储器;以及触摸屏,包括:显示介质,用于显示存储在该存储器中的至少一个图形对象的表示,该图形对象具有存储在该存储器中的至少一个参数;表面,用于确定对该至少一个参数的改变的指示,其中,响应于指示该改变,该计算装置被配置为自动改变该存储器中的至少一个参数,并自动改变该存储器中的一个或多个图形对象的表示;并且其中,该显示介质被配置为显示具有改变后的参数的一个或多个图形对象的改变后的表示。

[0008] 在公开的实施例的另一方面,一种方法包括:在计算装置的显示介质上显示矢量图形的表示,该矢量图形包括多个图形对象,每个图形对象具有存储在该存储器中的至少一个位置、以及一个或多个参数,其中每个参数能由一个或多个函数改变;检测对该多个图形对象中的至少一个图形对象的一个或多个参数中的至少一个参数的改变的指示;其中响应于检测到该指示:自动改变该至少一个参数;基于改变后的至少一个参数自动改变该矢量图形中的几何特征;基于改变后的几何特征自动改变该矢量图形的表示;并且在该显示介质上显示改变后的矢量图形的表示。

[0009] 通过参考以下结合附图的详细描述,将更好地理解所公开的实施例的这些和其他特征,这些附图应被认为是说明性的而非限制性的。

附图说明

[0010] 图1是示出了根据所公开的实施例的一个实施例的基本功能块和数据流的示意性框图。

[0011] 图2是中断处理程序的流程图,该中断处理程序响应于书写笔在书写表面上敲击(tap)来读取手写信息。

[0012] 图3是用于读取手写信息的轮询技术的流程图。

[0013] 图4是根据所公开的实施例的表示性实施例的操作流程图,其中在结束(conclude)所有手写信息之后,将手写信息合并到文档中。

[0014] 图5是根据所公开的实施例的表示性实施例的操作流程图,其中手写信息在输入期间被同时合并到文档中。

[0015] 图6是根据所公开的实施例的可用于在过程中的各个步骤期间显示手写信息的选项的图示示例。

[0016] 图7是手写符号/命令的样本及其相关联的含义的图示。

[0017] 图8是为图7中示出的前3个符号操作中的每个提供通用例程的列表。

[0018] 图9是从RHI存储器中处理和定义的识别功能元件接收到的数据的数据流的图示。

[0019] 图10是适于存储与一个手写命令相关联的数据的RHI存储器的存储器块格式的示例。

[0020] 图11是根据第一实施例的示出了模拟键盘击键的图1和图38的嵌入式元件的数据流的示例。

[0021] 图12是表示根据第一实施例的使用模拟键盘击键的技术的图4和图5的子例程D的

流程图。

[0022] 图13是根据第二实施例的示出了程序运行的图1和图38的嵌入式元件的数据流的示例。

[0023] 图14是表示根据第二实施例的示出了程序运行的图4和图5的子例程D的流程图。

[0024] 图15至图20是图12中针对图7中示出的前三个符号操作并根据图8中示出的通用例程引用的子例程H的流程图。

[0025] 图21是根据第一实施例的使用模拟键盘击键的技术的图4和图5中引用的用于结束Microsoft®Word类型文档的修订的嵌入的子例程L的流程图。

[0026] 图22是用于结束MS Word类型文档的修订的图21的子例程L的替代的流程图。

[0027] 图23是用于从RHI存储器复制识别出的图像并经由剪贴板将其放置在文档存储器中的图12中引用的子例程I的样本流程图。

[0028] 图24是用于将图像从RHI存储器复制到剪贴板中的图23和图37中引用的子例程N的代码的样本。

[0029] 图25是图26至图32和图37的流程图中引用的内置宏的转换后的Visual Basic代码的样本。

[0030] 图26至图32是图14中针对图7中示出的前三个符号操作并根据图8中针对MS Word示出的通用例程引用的子例程J的流程图。

[0031] 图33是用于根据第二实施例使用程序的运行来结束MS Word的修订的嵌入的Visual Basic中针对图4和图5中引用的子例程M的代码的样本。

[0032] 图34是MS Word注释模式下有用的内置宏的转换后的Visual Basic代码的样本。

[0033] 图35提供了将记录的宏转换为Visual Basic代码的示例,其模拟了MS Word的某些键盘键。

[0034] 图36是用于检查表中是否存在要被模拟为键盘击键的手写字符并因此可以被模拟的过程(并且如果存在,则用于执行模拟该击键的相关代码行)的流程图。

[0035] 图37是图14中的子例程K的示例的流程图,该子例程K用于从RHI存储器复制识别出的图像并将其经由剪贴板放置在文档存储器中。

[0036] 图38是图1示出的替代的示意性框图,其示出了根据本公开实施例的另一实施例的使用触摸屏的基本功能块和数据流。

[0037] 图39是使用无线垫板制作的集成式编辑文档的示意图。

[0038] 图40A-图40D示出了用户与触摸屏进行交互以插入线的示例。

[0039] 图41A-图41C示出了使用命令删除对象的示例。

[0040] 图42A-图42D示出了用户与触摸屏交互以改变线长的示例。

[0041] 图43A-图43D示出了用户与触摸屏交互以改变线角的示例。

[0042] 图44A-图44D示出了用户与触摸屏进行交互以将半径应用于线或改变圆弧的半径的示例。

[0043] 图45A-图45C示出了用户与触摸屏进行交互以使一条线与另一条线平行的示例。

[0044] 图46A-图46D示出了用户与触摸屏进行交互以向对象添加圆角或圆弧的示例。

[0045] 图47A-图47D示出了用户与触摸屏交互以添加倒角的示例。

[0046] 图48A-图48F示出了使用命令修剪对象的示例。

- [0047] 图49A-图49D示出了用户与触摸屏交互以移动弧形对象的示例。
- [0048] 图50A-图50D示出了使用“不捕捉”命令的示例。
- [0049] 图51A-图51D示出了使用“不捕捉”命令的另一示例。
- [0050] 图52A-图52D示出了使用命令修剪对象的另一示例。
- [0051] 图53是具有图标的用户界面的示例。
- [0052] 图54A-图54B示出了在触摸屏上与立方体的矢量图形的三维表示进行交互之前和之后的示例。
- [0053] 图54C-图54D示出了在触摸屏上与球体的矢量图形的三维表示进行交互之前和之后的示例。
- [0054] 图54E-图54F示出了在触摸屏上与斜坡的矢量图形的三维表示进行交互之前和之后的示例。
- [0055] 图55A-图55B示出了用于文本编辑、选择模式的用户界面菜单的示例。
- [0056] 图56示出了在命令模式下标记文本的手势示例。
- [0057] 图57示出了在命令模式下标记文本的手势的另一示例。
- [0058] 图58A-图58B示出了在绘制手势以标记文本时自动缩放文本的示例。

具体实施方式

[0059] 参考图1,其是根据第一实施例的集成式文档编辑器10的示意性框图,其示出了根据该第一实施例的基本功能块和数据流。使用数字化垫板12,其书写区域(例如,在8-1/2”×11”纸张的边距内)容纳与编辑页面的x-y位置相对应的标准大小的纸。垫板12接收来自书写笔10的数据(例如,利用标准笔借由压力磁力地或机械地)。来自数字化垫板12的数据被数据接收器14读取为位图和/或矢量数据,并且然后对应于或参考数据接收存储器16中的适当的x-y位置来存储。可选地,该信息可以实时地显示在显示器25的屏幕上,以向书写者提供实时反馈。

[0060] 可替换地,并且如图38中示出的,集成了其接收和显示机构的触摸屏11(或作为显示器的一部分的其他位置输入接收机构)接收来自书写笔10的数据,由此在触摸屏上显示原始文档,如同它将显示在数字化垫板12上的打印页面上一样,并且笔10的书写发生在触摸屏上与它将被书写在打印页面上的相同位置处。在这种情况下,图1的显示器25、垫板12和数据接收器14被替换为图38的元件11、触摸屏和相关联的电子器件,并且以下参考图1讨论元件16、18、20、22和24。在触摸屏显示替代方案下,消除了书写纸。

[0061] 当打印页面与数字化垫板12一起使用时,可能需要调整位置的配准,使得打印页面上的位置与存储在数据接收存储器16中的数据中的正确x-y位置相关。

[0062] 书写笔10的位置(在触摸屏11上或数字化垫板12上)与文档存储器22中的实际x-y位置之间的相关性不需要完全精确,这是因为笔10的位置是参考现有的机器代码数据。换句话说,在书写点周围存在误差窗口,其可以在不丢失有用信息的情况下被允许,这是因为假定新的手写信息(例如,修订)必须始终对应于笔的特定位置,例如,靠近文本、绘图或图像。这与将光标放置在文档中的插入点处并从命令模式改变为数据输入模式类似,但并不总是相同。例如,书写点可以在两行文本之间,但相比另一行文本而言更接近一行文本。可以根据笔敲击点和敲击点周围的数据连续计算该误差窗口。在关于旨在其中插入新数据的

确切位置出现模糊性的情况下(例如,当书写点与文档存储器22中的多个可能位置重叠时),触摸屏11(或垫板12)可以生成信号,诸如蜂鸣声,请求用户在更接近需要插入手写信息的点敲击。如果仍然未解决模糊性(使用数字化垫板12时),可能会要求用户遵循调整程序。

[0063] 如果需要,可以进行调整,使得将数字化垫板12上的书写区域设置为对应于特定的活动窗口(例如,在多窗口屏幕中)或窗口的一部分(即,当窗口的活动部分覆盖部分屏幕时,例如记帐程序QuickBooks的账单或发票),使得有效地利用了数字化垫板12的书写区域。在文档是表单(例如订单表单)的情况下,纸质文档可以预先设置为表单的特定格式,使得可以在表单的特定字段(其对应于文档存储器22中的这些字段)处输入手写信息。另外,在不需要存档手写纸质文档的操作中,数字化垫板12上的手写信息在被集成到文档存储器22中之后可以被删除。可替代地,可以使用允许多次删除(清除手写信息)的多用途介质,尽管触摸屏替代方案将优于该替代方案。

[0064] 识别功能元件18从数据接收存储器16读取信息,并将识别结果或识别出的手写元素写入识别出的手写信息(RHI)存储器20中。识别出的手写信息元素(RHI元素)诸如字符、单词和符号存储在RHI存储器20中。RHI元素在RHI存储器20中的位置与其在数据接收存储器16和文档存储器22中的位置相关。在符号被识别并解释为命令之后,它们可以以例如JPEG格式被存储为图像或图标(或者可以将它们模拟为好像它们是键盘键。此技术将在下文进行讨论。),这是因为这些符号是为了直观起见。它们对于审查和解释文档中的修订很有用。另外,在最终合并之前识别出的手写信息(例如,用于审查的修订)可以手写(原样或作为修订后的机器代码手写以提高可读性)或以标准文本显示。

[0065] 嵌入式标准和功能元件24从RHI存储器20读取信息并将其嵌入到文档存储器22中。文档存储器22中的信息显示在显示器25上,显示器25例如是计算机监视器或触摸屏的显示器。嵌入式功能基于修订的阶段和选出的用户标准/首选项来确定要显示什么以及将什么嵌入到文档存储器22中。

[0066] 将识别出的信息嵌入到文档存储器22中既可以同时应用,或者在所有手写信息输入结束之后(诸如在修订之后)应用。在有或没有用户参与的情况下,手写信息的合并可以同时发生。用户每次可以指示手写命令及其相关联的文本和/或图像已结束,并且然后可以一次一个将其合并到文档存储器22中。(下面将讨论在没有用户参与的情况下同时合并手写信息。)文档存储器22包含例如以下文件之一:1)文字处理文件,诸如MS Word文件或Word Perfect文件;2)电子表格,诸如Excel文件;3)表单,诸如销售订单、会计软件(例如QuickBooks)中的账单或发票;4)表格或数据库;5)桌面发布文件,诸如QuarkXPress或PageMaker文件;或6)演示文件,诸如MS PowerPoint文件。

[0067] 应当注意,文档可以是任何类型的电子文件、文字处理文档、电子表格、网页、表单、电子邮件、数据库、表格、模板、图表、图形、图像、对象或这些类型的文档的任何部分,诸如文本块或数据单元。另外,文档存储器22、数据接收存储器16和RHI存储器20可以是任何类型的存储器或存储器装置或存储器装置的一部分,例如,任何类型的RAM、磁盘、CD-ROM、DVD-ROM、光盘或任何其他类型的存储。还应注意,本领域技术人员将认识到,本文中讨论的元件/组件(例如,在图1、图38、图9、图11、图13中)诸如RHI元件可以在电子或计算机硬件和/或软件的任何组合中实施。例如,所公开的实施例可以在通用计算机或其他类型的计

算/通信装置(诸如手持计算机、个人数字助理(PDA)、手机等)上操作的软件中实施。可替代地,通用计算机可以与诸如专用集成电路(ASIC)或某些其他电子组件之类的专用硬件接口以实施所公开的实施例。因此,可以理解,可以使用形成程序并通过例如中央处理单元执行为指令/数据的一个或多个软件模块的各种代码来执行所公开的实施例,或者可以使用专门配置并专用于执行所公开的实施例的硬件模块来执行所公开的实施例。可替代地,可以使用软件和硬件模块的组合来执行所公开的实施例。

[0068] 识别功能元件18包含以下识别方法中的一种或多种:

[0069] 1-字符识别,例如,其可以在用户用大写字母清楚地拼写每个字符以努力最小化识别错误的情况下使用,

[0070] 2-整体研究,其中对单词的整体表示进行全局识别,并且不尝试独立识别字符。(整体方法的主要优点是避免了分词。它们的主要缺点是它们与单词描述的固定词典有关:由于这些方法不依赖字母,因此单词是借由特征直接描述的。向词典添加新单词典型地需要人工训练或从ASCII单词自动生成单词描述。)

[0071] 3-分析策略,其处理与增加的抽象级别相对应的几个级别的表示。(单词不是作为一个整体来考虑的,而是作为较小尺寸单位的序列,其必须与字符容易地关联,以便进行独立于特定词汇的识别。)

[0072] 可以通过整体研究或通过分析策略来识别单词或符号的字符串,诸如结合图7描述的并在下文讨论的那些,尽管可能优选字符识别。被识别为字符、单词或符号的单元例如以ASCII格式存储在RHI存储器20中。作为图形的单元作为图形(例如作为JPEG文件)存储到RHI存储器中。如果该应用程序容纳图形,并且可选地,如果由用户批准为图形并作为图形存储到RHI存储器20中,则将不能被识别为字符、单词或符号的单元解释为图像。应当注意,在不能容纳图形的应用程序(例如Excel)中,不能被识别为字符、单词或符号的单元可能不能解释为图形;在这种情况下,可能需要用户参与。

[0073] 为了改善识别功能,可以由识别元件18从文档存储器22读取数据,以验证识别出的手写信息与原始文档中的数据不冲突,并尽可能地解析/最小化保留模糊性的识别出的信息。用户还可以通过批准/不批准显示器25上示出的识别出的手写信息(例如,修订)来解决模糊性。此外,可以采用自适应算法(超出本公开的范围)。据此,用户参与最初可能相对重要,但是随着自适应算法学习特定的手写模式并将其存储为历史模式,随着识别变得更加鲁棒,未来的模糊性应被最小化。

[0074] 图2至图5是根据示例性实施例的操作的流程图,并且下面对其进行简要解释。出于权利要求支持的目的,将所有附图中的文本在此明确地并入该书面描述中。图2示出了每当书写笔10敲击和/或离开垫板12(或触摸屏11)的书写表面时读取数字化垫板12(或触摸屏11)的输出的程序。此后,将数据存储在数据接收存储器16中(步骤E)。识别元件和数据接收器(或触摸屏)两者都访问数据接收存储器。因此,在一个元件的读/写周期期间,应禁用另一元件的访问。

[0075] 可选地,如图3中示出的,程序每隔几毫秒检查一次,以查看是否有新数据要从数字化垫板12(或触摸屏11)读取。如果是这样,则从数字化识别器接收数据并将其存储在数据接收存储器16中(E)。该过程继续,直到用户指示出修订已结束为止或直到超时为止。

[0076] 手写信息的嵌入可以根据图4解释的程序一次执行,或者根据图5解释的程序同时

执行。

[0077] 识别元件18一次识别一个单元,例如一个字符、一个单词、图形或符号,并使它们可用于RHI处理器和存储器20(C)。此后,将参考图9讨论该处理器的功能及其将识别出的单元存储到RHI存储器中的方式。未立即识别出的单元将在最后作为图形处理,或者用户可以通过其他手段手动指示,诸如选择表或键盘输入(F)。可替代地,如果用户指示何时开始图形写入以及何时结束图形写入,则将图形解释为图形。一旦手写信息结束,它就被分组到存储器块中,由此每个存储器块都包含与一个手写命令(例如修订)有关的全部(如图4所示)或可能部分(如图5所示)识别出的信息。然后,嵌入式功能(D)将识别出的手写信息(例如,修订)以“供审查”模式嵌入。一旦用户批准/不批准修订,根据用户设置的首选项(A)将其以最终模式(L)嵌入。在以下示出的示例中,MS Word中的修订以追踪修订模式一次全部嵌入。而且,在以下示出的示例中,例如,当数字化垫板12与系统的其余部分分离时,根据图4的MS Word中的修订可以是有用的,由此在经由USB或其他IEEE或ANSI标准端口结束修订后,可以将来自数字化垫板内部存储器的手写信息下载到数据接收存储器16中。

[0078] 图4是各个步骤的流程图,由此,一旦结束“全部”手写信息,就将“全部”识别出的手写信息(诸如修订)嵌入文档存储器22中。首先,设置文档类型(例如Microsoft®Word或QuarkXPress),具有软件版本和用户首选项(例如是否当修订可用时合并该修订,或者在用户批准/不批准时一次合并一个修订),以及用户针对各种命令(诸如插入文本、删除文本和移动文本)优选的各种符号(A)。手写信息从数据接收存储器16读取,并存储在识别元件18的存储器中(B)。从接收存储器16读取的信息被标记/标志为已读取,或者在被识别元件18读取并存储在其存储器中之后被擦除;这将确保识别元件18仅读取新数据。

[0079] 图5是各个步骤的流程图,由此同时执行(例如,随着修订的进行)将识别出的手写信息(例如,修订)嵌入文档存储器22中。步骤1-3与图4中的流程图的步骤(上面讨论的)相同。一旦识别出诸如字符、符号或单词的单元,就由RHI处理器20对其进行处理并将其存储在RHI存储器中。处理器(图9中引用的GMB功能30)将其识别为可以立即嵌入或不嵌入的单元。检查它是否可以嵌入(步骤4.3);如果可以的话(步骤5),则将其嵌入(D),并且然后(步骤6)将其删除或标记/更新为嵌入的(G)。如果其不能被嵌入(步骤4.1),则从数字化垫板12(或从触摸屏111)读取更多信息。只要手写信息即将出现,就会重复并继续步骤4-6的过程。一旦嵌入所有数据(通过End命令或简单的超时指示出),将以与图4的流程图讨论的相同方式处理(F)无法识别出的单元。最后,一旦用户批准/不批准修订,它们根据用户选择的首选项以最终模式嵌入(L)。

[0080] 图6是可供用户在MS Word的各个步骤中显示手写信息各种选项和首选项的示例。在“供审查(ForReview)”模式下,修订显示为“供审查”,以待批准“最终”合并。例如,修订可以以“追踪修订(Track Changes)”模式被嵌入,并且一旦被批准/不被批准(如在“接受/拒绝改变”中),它们就被作为“最终”被嵌入到文档存储器22中。可替代地,符号也可以被显示在显示器25上。符号被选择性地选择为直观的,并且因此对于快速审查修订是有用的。出于相同的原因,文本修订可以按原样以手写显示,或者按修订后的机器代码手写显示,以提高可读性;在“最终”模式下,所有符号都将被擦除,并且修订被合并为文档的组成部分。

[0081] 图7中示出了手写命令/符号及其相对于其相关联的数据的解释的基本集合的示

例,以在各种类型的文档中进行修订。

[0082] 对于读/写操作,需要直接访问文档存储器22中的特定位置。对于售后应用程序而言,将来自RHI存储器20的识别出的手写信息嵌入到文档存储器22中(例如,用于合并修订)可能是不可能的(或被限制)。下面讨论的每个实施例提供了替代的“后门”解决方案来克服该障碍。

[0083] 实施例一:模拟键盘输入:

[0084] RHI存储器20中的命令信息用于在文档存储器22中的指定位置中插入或修订数据,诸如文本或图像,其中,执行机构模拟键盘击键,并且在可用时与运行的预先记录和/或分配给击键序列(即快捷键)的内置宏结合操作。诸如文本的数据可以从RHI存储器20复制到剪贴板,并且然后粘贴到文档存储器22中的指定位置,或者其可以被模拟为键盘击键。该实施例将在下文讨论。

[0085] 实施例二:运行程序:

[0086] 在诸如Microsoft®Word、Excel和WordPerfect的应用程序中,在诸如VB脚本和Visual Basic等编程能力可用的情况下,将存储在RHI存储器20中的命令及其相关联的数据转换为将其按预期嵌入到文档存储器22中的程序。在该实施例中,操作系统剪贴板可以用作数据(例如,文本和图像)的缓冲器。此后还将讨论该实施例。

[0087] 如实施例一和实施例二中所讨论的,与手写命令相关联的信息是文本或图形(图像),尽管它可以是文本和图形的组合。在任一实施例中,剪贴板都可以用作缓冲器。

[0088] 对于RHI存储器中的复制操作:

[0089] 当从RHI存储器20中的存储器块中指示出的特定位置复制文本或图像的单元以插入文档存储器22中的指定位置时。

[0090] 对于文档存储器中的剪切/粘贴和粘贴操作:

[0091] 用于在文档存储器22内移动文本或图像,并且用于粘贴从RHI存储器20复制的文本或图像。

[0092] 实施例一的主要优势是,在具有或不具有编程能力的大量应用程序中,仅依赖控制键执行命令以及在内置或预先记录的宏可用时的有用性。当模拟控制键(诸如向上箭头)或同时组合键(诸如Ctrl-C)时,执行命令。

[0093] 除非被转换成实际的低级编程代码(例如,Visual Basic Code),否则宏不能在实施例二中运行。相反,在实施例一中以应用程序固有的控制语言(记录的和/或内置的)运行宏可以简单地通过模拟其分配的一个或多个快捷键来实现。如果使用Visual Basic编辑器创建包括不能记录为宏的Visual Basic指令的代码,则例如在MS Word中,实施例二可能优于实施例一。

[0094] 可替代地,实施例二可以与实施例一结合使用,由此,例如,代替将文本从RHI存储器20移动到剪贴板,并且然后将其放置在文档存储器22中的指定位置中,将文本模拟为键盘击键。如果需要,可以在实施例二中通过为每个键编写代码来模拟键盘键,该代码在执行时模拟击键。可替代地,可以针对诸如QuarkXPress之类的不具有编程能力的应用程序来实施实施例一,并且可以针对具有编程能力的一些应用程序来实施实施例二。在这种情况下,具有编程能力的某些应用程序仍可以在实施例一中或者在实施例一和实施例二两者中实施。

[0095] 可替代地,数据接收存储器16中的x-y位置(以及文档存储器22中的指定位置)可以基于以下内容在打印输出上或在显示器25上并且(如果需要的话)在触摸屏11上进行识别:1)识别/鉴别书写笔周围的唯一文本和/或图像表示,以及2)搜索笔周围的识别出/鉴别出的数据并将其与原始文档中的数据进行匹配,该原始文档中的数据可以被转换为与手写信息格式相同的位图和/或矢量格式,其被存储在数据接收存储器16中。然后,手写信息及其在文档存储器22中对应索引的x-y位置被发送到远程平台,以进行识别、嵌入和显示。

[0096] 具有在笔中内置的附接电路的微型相机读取书写笔周围的数据表示形式和手写信息。在开始手写之前,经由无线连接(例如,蓝牙)或经由物理连接(例如,USB端口)将表示文档存储器22中的原始数据的数据下载到笔内部存储器中。

[0097] 手写信息结束后(经由物理或无线链接),将手写信息及其识别出的x-y位置下载到远程平台的数据接收存储器16中,或者在识别手写信息的x-y位置时,经由无线链路将其发送到远程平台。然后,将手写信息一次(即,根据图4示出的流程图)或同时(即,根据图5中示出的流程图)全部嵌入文档存储器22中。

[0098] 如果需要,显示器25可在整个显示器上或显示器的所选位置处包括预设图案(例如,雕刻或丝网印刷),使得当由笔的相机读取时,可以确定显示器25上的确切x-y位置。显示器25上的预设图案可用于解决模糊性,例如当文档存储器22中的位置周围的相同信息在文档中多次存在时。

[0099] 此外,在触摸屏11的所选位置上的笔的敲击可用于确定文档存储器中的x-y位置(例如,当用户在显示在触摸屏上的表单内进行是-否类型选择时)。例如,这可以在可以接受来自笔或用作鼠标和书写工具的任何其他定点装置的输入的平板电脑上执行。

[0100] 可替代地(或除了触摸屏之外),书写笔可以向具有热感测或光学感测的屏幕发射聚焦的激光/IR束,并且感测到的束的位置可以用于识别屏幕上的x-y位置。在这种情况下,不需要使用具有内置微型相机的笔。当使用具有热感测/光学感测的触摸屏或显示器(或普通显示器上预设图案时)来检测屏幕上的x-y位置时,可以基于以下来确定文档存储器22中的指定的x-y位置:1)检测到的笔10在屏幕上的x-y位置,以及2)在显示的数据和文档存储器22中的数据之间相关的参数(例如,应用程序名称、屏幕上的光标位置和缩放百分比)。

[0101] 可替代地,可以模拟鼠标以基于数据接收存储器16中指示出的X-Y位置将插入点放置在文档存储器22中的指定位置处。然后,可以根据实施例一或实施例二将来自RHI存储器20的信息嵌入到文档存储器22中。此外,一旦插入点位于文档存储器22中的指定位置处,也可以通过模拟鼠标指针点击操作来实现对文档存储器22内的文本或图像的选择。

[0102] 注释插入特征的使用:

[0103] **Microsoft®Word**的注释特征(或其他程序应用中的类似注释插入功能)可以由用户采用,或自动与上面讨论的任何一种方法结合使用,并且然后可以将来自RHI存储器20的手写信息嵌入文档存储器22的指定注释字段中。此方法将在下文中进一步讨论。

[0104] 追踪修订特征的使用:

[0105] 在将信息嵌入文档存储器22之前,识别文档类型并且设置用户首选项(A)。用户可以选择在追踪修订特征中显示修订。**Microsoft®Word**的追踪修订模式(或其他应用程序中的类似特征)可以由用户调用,或者结合实施例一和实施例二中的一个或两个自动调用,并且然后可以将来自RHI存储器20的手写信息嵌入文档存储器22中。在将所有修订合并到

文件存储器22中之后,它们可以被整个文档接受,或者它们可以根据用户命令一次被接受/拒绝一个。可替代地,可以在进行修订时接受/拒绝它们。

[0106] 插入机制也可以是模拟追踪修订特征的插件。可替代地,可以在调用注释特征之后调用追踪修订特征,使得注释字段中的修订显示为修订,即“供审查”。这对于由多方审查/修订的大型文档尤其有用。

[0107] 在另一个实施例中,原始文档被读取并转换成具有已知可访问格式(例如,用于文本的ASCII和用于图形的JPEG)的文档,并被存储到中间存储器位置。所有读/写操作都直接在其上执行。一旦完成修订,或者在传输到另一个平台之前,可以将其转换回原始格式并存储到文档存储器22中。

[0108] 如所讨论的,修订被书写在放置在数字化垫板12上的纸质文档上,由此该纸质文档包含/类似于存储在文档存储器22中的机器代码信息,并且该纸质文档上的x-y位置对应于文档存储器22中的x-y位置。在可替代的实施例中,可以在空白纸(或另一文档)上进行修订,由此,手写信息例如是用于在电子表格的单元格中写入或修订值/数字的命令(或命令的集合),或用于更新数据库特定位置中的新信息;这可能很有用,例如在审查文档(或文档的集合)后需要用于更新电子表格、表格或数据库的动作的情况下。在该实施例中,接收存储器16中的x-y位置无关紧要。

[0109] RHI处理器和存储器块

[0110] 在参考流程图更详细地讨论将信息嵌入文档存储器22中的方式之前,有必要定义如何将识别出的数据存储存储在存储器中以及如何将其与文档存储器22中的位置相关联。如前面解释的,将识别出的信息嵌入到文档存储器22中既可以同时应用,或者在所有手写信息都结束之后应用。图4中引用的嵌入式功能(D)一次从RHI存储器20中的存储器块读取一个数据,其对应于一个手写命令及其相关联的文本数据或图像数据。图5中引用的嵌入式函数(D)从存储器块读取数据并同时嵌入识别出的单元。

[0111] 存储器块:图10中示出了如何在存储器块32中定义手写命令及其相关联的文本或图像的示例。例如,如果添加了额外的命令,即除了在命令字段中指定的命令之外,则可以扩展此格式。定义识别出的单元的x-y位置的参数(即图10中的插入点1和插入点2)取决于应用程序而变化。例如,可以使用参数页数(Page#)、行数(Line#)和列数(Column#)来定义MS Word中文本或图像的x-y位置/插入点(如图10中示出的)。在应用程序Excel中,x-y位置可以转换为电子表格中的单元格位置,即Sheet#、Row#和Column#。因此,需要定义x-y插入点1和x-y插入点2的不同格式,以适应各种应用程序。

[0112] 图9是识别出的单元的数据流图。这些将在下面讨论。

[0113] FIFO(先进先出)协议:一旦识别出一个单元,就将其存储在队列中,等待元件20的处理器处理,并且更具体地说,等待GMB功能30的处理。“New Recog”标志(当单元可用时由识别元件18设置为“一”)向RU接收器29指示出识别出的单元(即,队列中的下一个)是可用的。在识别出的单元被读取并存储在图9的存储器元件26和28中(例如,如图4和图5中示出的子例程的步骤3.2中)之后,将“New Recog”标志重置回“零”。作为响应,识别元件18:1)使下一个识别出的单元可被RU接收器29读取,并且2)将“New Recog”标志设置回“一”,以向RU接收器29指示下一单元准备就绪。只要识别出的单元即将出现,这个过程就继续。该协议确保识别元件18与识别出的单元被读取并存储在RHI存储器中(即,在图9的存储器元件26和

28中)的速度同步。例如,当同时处理手写信息时,在将先前的存储器块嵌入文档存储器22中之前,可能存在多于一个的存储器块可用。

[0114] 以类似的方式,也可以在元件24和22之间以及图1和图38的元件16和18之间、以及图1的元件14和12之间采用这种FIFO技术,以确保独立过程良好地同步,无论一个元件可用数据的速度如何、以及另一个元件读取和处理数据的速度如何。

[0115] 可选地,“New Recog”标志可以在h/w中(诸如在IC内)实施,例如,通过在识别出的单元可用时将行设置为“高”,并且在单元被读取并存储后将行设置为“低”,即确认收到。

[0116] 过程1:作为单元(诸如字符、符号或单词)被识别:1)它被存储在识别出的单元(RU)存储器28中,并且2)如在数据接收存储器16中指示出的,其在RU存储器28中的位置及其x-y位置被存储在XY-RU位置中,以在RU表26中寻址。只要手写单元被识别并且即将出现,该过程就会继续。

[0117] 过程2:与过程1并行,分组成存储器块(GMB)功能30鉴别每个识别出的单元(诸如字符、单词或手写命令(符号或单词)),并将它们存储在存储器块32的适当位置。在诸如“移动文本”、“增大字体大小”或“改变颜色”的操作中,在将整个手写命令嵌入到文档存储器22之前,必须结束该手写命令。在诸如“删除文本”或“插入新文本”的操作中,一旦识别出命令,就可以开始删除或嵌入文本,并且然后当用户继续在数字化垫板12(或在触摸屏11上)上书写时,删除(或插入文本)操作可以同时继续。

[0118] 在该最后一种情况下,一旦将一个或多个识别出的单元合并到文档存储器22中(或从中删除),它就从RHI存储器22中删除,即从图9的存储器元件26、28和32中删除。如果不期望删除,则可以将嵌入式单元标志为“合并的/嵌入的”或移动到另一个存储器位置(如图5中的流程图的步骤6.2中示出的)。这应确保持存储器块中的信息与新的未合并信息持续地保持一致。

[0119] 过程3:由于将一个或多个单元分组到存储器块中,所以1)识别出的单元的标识(无论其是否可以立即合并),以及2)可以合并到RHI存储器中的单元的位置被持续地更新。

[0120] 1.由于将单元分组到存储器块中,因此将标志(即“标识标志”)设置为“一”,以指示何时可以嵌入一个或多个单元。应注意的是,为每个存储器块定义了该标志,并且可以为同一存储器块设置多于一次(例如,当用户通过一行文本敲击时)。该标志在图5的步骤4.1-4.3中检查,并在嵌入一个或多个识别出的单元后(即在图5中子例程的步骤6.1中,并且在初始化时)重置为“零”。应注意,当与存储器块相关联的所有识别出的单元一次全部嵌入时,上面讨论的“标识”标志是无关紧要的;在这种情况下,并且在手写信息结束、被识别出、被分组并存储在RHI存储器的正确位置之后,图9的GMB功能30会将图4的步骤6.1中的“所有单元”标志设置为“一”,以指示出可以嵌入所有单元。

[0121] 2.由于单元被分组到存储器块中,因此,每当引入新的存储器块时(即,当引入尚未准备好嵌入的一个或多个识别出的单元时;当“标识”标志为零时),并且每当将存储器块嵌入文档存储器22中时,都会更新用于存储器块的指针,即“下一个存储器块指针”31,使得指针将总是指向已准备好(当其准备好时)嵌入的存储器块的位置。该指针向(图12的)子例程Embedd1和(图14的)Embedd2指示具有已准备好嵌入的一个或多个识别出的单元的相关存储器块的确切位置(如在这些子例程的步骤1.2中)。

[0122] 更新“下一个存储器块指针”31的场景的示例是:当与改变字体大小有关的手写输

入开始时,然后与改变颜色有关的另一手写输入已经开始(请注意,这两个命令无法合并,直到其结束后为止),并且然后用于删除文本的另一个手写输入已经开始(请注意,只要GMB功能识别出该命令,该命令就会被嵌入)。

[0123] “存储器块数”33中的值指示出要嵌入的存储器块的数量。此元素由GMB功能30设置,并在图12和图14中示出的子例程的步骤1.1中使用。当手写信息在其结束后立即全部嵌入时,即当从图4中示出的子例程调用图12和图14的子例程时(即,从图5的子例程调用它们时,其不相关;然后将其值设置为“一”,这是因为在该实施例中,一次嵌入一个存储器块),此计数器是相关的。

[0124] 实施例一

[0125] 图11是示出根据实施例一的基本功能块和数据流的示意性框图。这些和所有其他附图的文字在很大程度上是不言而喻的,并且在此无需重复。但是,其文本可能是本文档中使用的权利要求语言的基础。

[0126] 图12是根据实施例一的图4和图5中引用的嵌入式子例程D的流程图示例。以下是需要注意的。

[0127] 1. 当图5中示出的例程调用此子例程时(即,同时嵌入手写信息时):1) 将存储器块计数器(在步骤1.1中)设置为1,并且2) 将存储器块指针设置为当前要嵌入的存储器块所在的位置;该值在图9的存储器块指针元素(31)中定义。

[0128] 2. 当该子例程由图4中示出的子例程调用时(即,当所有手写信息结束之后嵌入所有手写信息时):1) 将存储器块指针设置为要嵌入的第一个存储器块的位置,并且2) 将存储器块计数器设置为图9的存储器块元素(33)的#中的值。

[0129] 在操作中,一次从RHI存储器20中取出一个存储器块32(G),并按以下方式处理:

[0130] 与文本修订有关的存储器块(H):

[0131] 命令以与经由键盘执行操作相同的序列被转换为击键(35),并且然后以序列存储在击键存储器34中。模拟键盘元件36使用该数据来模拟键盘,使得应用程序读取从键盘接收到的数据(尽管此元件可能包括经由键盘不可用的附加键,诸如图7中示出的符号,例如用于在MS Word文档中插入新文本)。剪贴板38可以处理文本的插入,或者文本可以被模拟为键盘击键。查找表40确定预先记录的宏和内置宏的一个或多个适当的控制键和击键序列,这些宏在被模拟时执行所期命令。这些键盘键依赖于应用程序,并且是参数的函数,诸如应用程序名称、软件版本和平台。某些控制键(诸如,箭头键)在大量应用程序中执行相同的命令;然而,该假设被排除在图11中的设计之外,即,通过在图11的元素40中包含查找表命令击键。尽管在图15-20的流程图中,假定以下控制键(在包含的应用程序中)执行相同的命令:“向上翻页”、“向下翻页”、“向上箭头”、“向下箭头”、“向右箭头”和“向左箭头”(用于在文档内移动插入点)、“Shift+向右箭头”(用于选择文本)和“删除”(用于删除所选文本)。元素40可以包括用于大量应用程序的查找表,尽管它可以包括用于一个或任何所期数量的应用程序的表。

[0132] 与新图像有关的存储器块(I):

[0133] 首先将图像(图形)从RHI存储器20复制(更具体地,基于存储器块32中的信息)到剪贴板38中。其指定位置经由击键的序列(例如,经由箭头键)位于文档存储器22中。它被存储(即,通过击键序列Cntr-V从剪贴板38粘贴)到文档存储器22中。如果该命令涉及另一种

操作,诸如“减小图像大小”或“移动图像”,则该图像首先在文档存储器22中被识别并被选择。然后,通过适当的击键序列应用该操作。

[0134] 图15至图20(图12中引用的子例程H的流程图)示出了针对MS Word和其他应用程序结合图8并且在图8中讨论的前三个基本文本修订的执行。这些流程图是不言而喻的,并且因此在此不做进一步描述,而是并入该文本中。

[0135] 参考图15的流程图中示出的函数StartOfDocEmbl,应注意以下几点:

[0136] 1. 该函数由函数SetPointerembl调用,如图16中示出的。

[0137] 2. 尽管在许多应用程序中,快捷键组合“Cntrl+Home”会将插入点带到文档(包括MS Word)的开头,但是编写此例程是为了利用箭头键执行相同的操作。

[0138] 3. 在该子例程中的文档存储器22中指定的x-y位置是基于Page#、Line#和Column#定义的;当x-y定义不同时,需要其他子例程。

[0139] 一旦嵌入所有修订,它们就将根据图21中示出的流程图或图22中示出的流程图以最终模式合并。在此实施方式示例中,追踪修订特征用于“接受所有改变”,其嵌入所有修订作为文档的组成部分。

[0140] 如上面讨论的,击键序列的基本集合可用于执行命令的基本集合,以便在大量应用程序中创建和修订文档。例如,箭头键可用于跳至文档中的指定位置。当这些键与Shift键结合使用时,可以选择所期文本/图形对象。此外,剪贴板操作,即典型的组合击键序列Cntrl-X(用于剪切)、Cntrl-C(用于复制)和Cntrl-V(用于粘贴),可用于许多应用中的基本编辑/修订操作。应该注意的是,尽管可用的键盘控制键的数量相对较少,但是在这方面,在OEM级别处的应用程序设计不受限制。(例如参见图1-图5)。应当注意,相同的键组合可以执行不同的命令。例如,在QuarkXPress中删除项目是通过击键Cntrl-K实现的,其中MS Word中的击键Cntrl-K会打开超链接。因此,通过访问图11的查找表命令击键命令控制键40,ConvertText1函数H为存储在RHI存储器中的命令数据确定键盘击键序列。

[0141] 宏的使用:

[0142] 在诸如Microsoft® Word、Excel和Word Perfect的应用程序中,使用宏增强了手写命令的执行。这是因为可以简单地记录可以执行所期操作的击键序列并将其分配给快捷键。一旦模拟一个或多个分配的快捷键,就执行记录的宏。以下是Microsoft® Word的一些有用的内置宏。为了简化,将它们基于用于嵌入手写信息(D)的操作进行分组。

[0143] 将插入点带到文档中的特定位置:

[0144] CharRight, CharLeft, LineUp, LineDown, StartOfDocument, StartOfLine, EndOfDocument, EndOfLine, EditGoto, GotoNextPage, GotoNextSection, GotoPreviousPage, GotoPreviousSelection, GoBack

[0145] 选择:

[0146] CharRightExtent, CharLeftExtend, LineDownExtend, LineUpExtend, ExtendSelection, EditFind, EditReplace

[0147] 对所选单词/图形的操作:

[0148] EditClear, EditCopy, EditCut, EditPaste, CopyText, FontColors, FontSizeSelect, GrowFont, ShrinkFont, GrowFontOnePoint, ShrinkFontOnePoint, AllCaps, SmallCaps, Bold, Italic, Underline, UnderlineCoor, UnderlineStyle,

WordUnderline,ChangeCase,DoubleStrikethrough,Font,FontColor,FontSizeSelect

[0149] 显示修订:

[0150] Hidden,Magnifier,Highlight,DocAccent,CommaAccent,DottedUnderline,DoubleUnderline,DoubleStrikethrough,HtmlSourceRefresh,InsertFieldChar (用于封装显示符号),ViewMasterDocument,ViewPage,ViewZoom,ViewZoom100,ViewZoom200,ViewZoom75

[0151] 图像:

[0152] InsertFrame,InsertObject,InsertPicture,EditCopyPicture,EditCopyAsPicture,EditObject,InsertDrawing,InsertFram,InsertHorizontlLine

[0153] 文件操作:

[0154] FileOpen,FileNew,FileNewDefault,DocClose,FileSave,SaveTemplate

[0155] 如果没有为宏分配快捷键,则其可以通过以下程序进行分配:

[0156] 在工具菜单上点击,并且选择自定义,导致出现自定义表单。在键盘按钮上点击会产生自定义键盘对话框。在类别框中列出所有菜单,并且在命令框中列出所有其相关联的命令。通过在命令框中选择所期内置宏并按下所期快捷键,可以简单地将快捷键分配给特定的宏。

[0157] 宏的组合可以记录为新的宏;每当模拟分配给它的击键序列时,新宏就会运行。以相同的方式,结合击键(例如,箭头键)的宏可以被记录为新的宏。应当指出,可能不允许将某些序列记录为宏。

[0158] 宏的使用以及对宏的键序列的分配也可以在其他字处理器(诸如WordPerfect)中完成。

[0159] 在具有内置编程能力的应用程序(诸如Microsoft®Word)中模拟键盘键36,可以通过运行相当于按下该键盘键的代码来实现。参考图35和图36,呈现了该操作的细节。其文本通过引用并入本文。否则,模拟键盘是可以与Windows或其他计算机操作系统结合执行的功能。

[0160] 实施例二

[0161] 图13是示出根据实施例二的基本功能块和数据流的示意性框图。图14是根据实施例二的图4和图5中引用的嵌入式函数D的流程图示例。从RHI存储器20(G)中取出存储器块并进行处理。这些附图的文本通过引用并入本文。

[0162] 图14应注意以下几点:

[0163] 1. 当图5中示出的例程调用此子例程时(即,当同时嵌入手写信息时):1) 将存储器块计数器(在下面的步骤1.1中)设置为1,并且2) 将存储器块指针设置为当前要嵌入的存储器块所在的位置;该值在图9的存储器块指针元素(31)中定义。

[0164] 2. 当该子例程被图4中示出的子例程调用时(即,当所有手写信息都结束之后嵌入所有手写信息时):1) 存储器块指针被设置为要嵌入的第一个存储器块的位置,并且2) 将存储器块计数器设置为图9的存储器块元素(33)的#中的值。

[0165] 程序的集合一次执行一次在图9的存储器块32中定义的命令。图26至图32是图14中引用的子例程J的流程图,其中的文本通过引用方式并入本文。所描绘的程序执行图8中针对MS Word讨论的前三个基本文本修订。这些子例程是不言而喻的,并且此处不进一步解

释,但该文本通过引用并入。

[0166] 图33是Visual Basic中的代码,它以最终模式嵌入信息,即“接受追踪修订的所有改变”,其将所有修订嵌入为文档的组成部分。

[0167] 需要将图26至图32的流程图中引用的每个宏转换为可执行代码,诸如VB脚本或Visual Basic代码。如果不确定使用哪种方法或属性,则宏记录器典型地可以将记录的操作转换为代码。这些宏到Visual Basic的转换后的代码在图25中示出。

[0168] 剪贴板38可以处理文本到文档存储器22中的插入,或者文本可以被模拟为键盘击键。(详见图35-图36)。如在实施例一中,诸如将图像从RHI存储器20复制到文档存储器22的图像操作(K)执行如下:首先将图像从RHI存储器20复制到剪贴板3f8。它的指定位置位于文档存储器22中。然后经由剪贴板38将其粘贴到文档存储器22中。

[0169] 程序选择和执行元件42对程序的选择是命令、应用程序、软件版本和平台等的函数。因此,ConvertText2 J通过访问查找命令程序表44为存储在RHI存储器20中的命令数据选择特定程序。程序也可以由事件发起,例如,在打开或关闭文件时,或通过键输入,例如,通过按下Tab键将插入点带到电子表格的特定单元格时。

[0170] 在Microsoft®Word中,Visual Basic编辑器可用于创建非常灵活、功能强大的宏,这些宏包括无法从键盘记录的Visual Basic指令。Visual Basic编辑器提供了额外的帮助,诸如关于对象和属性或其行为方面的参考信息。

[0171] 使用注释特征作为插入机制

[0172] 在以下情况中,通过注释特征将手写修订合并到文档中可能是有益的:修订主要是将新文本插入指定的位置,或者当需要对文档中各个指定位置的多个修订进行索引以简化将来对修订的访问;这对于由多方审查的大型文档尤其有用。每个注释都可以进一步加载到子文档中,该子文档由主文档中的注释#(或标志)引用。注释模式也可以与追踪修订模式一起使用。

[0173] 对于实施例一:可以通过模拟击键序列Alt+Cntrl+M来实现插入注释。具有此序列的已记录宏的Visual Basic转换后的代码为“Selection.Comments.Add Range:=Selection.Range”,这可以用于在实施例2中实现相同的结果。

[0174] 一旦在注释模式中,就可以将RHI存储器20中的修订作为注释合并到文档存储器22中。如果文本包括修订,则可以在将文本插入注释窗格之前调用追踪修订模式。

[0175] 在MS Word的注释模式下使用的有用的内置宏:

[0176] GotoCommentScope;突出显示与注释参考标记相关联的文本

[0177] GotoNextComment;跳转到活动文档中的下一个注释

[0178] GotoPreviousComment;跳转到活动文档中的上一个注释

[0179] InsertAnnotation;插入注释

[0180] DeleteAnnotation;删除注释

[0181] ViewAnnotation;显示或隐藏注释窗格

[0182] 上述宏可以在实施例一中通过模拟其快捷键来使用,或者在实施例二中利用其在Visual Basic中的转换后的代码来使用。图34提供了这些宏中的每个的转换后的Visual Basic代码。

[0183] 电子表格、表单和表格

[0184] 将手写信息嵌入电子表格的单元格或表单或表格中的字段可以是新信息,或者它可以用于修改现有数据(例如,删除、在单元格之间移动数据或在字段中添加新数据)。无论哪种方式,在将手写信息嵌入文档存储器22中之后,它都可以致使应用程序(例如,Excel)改变文档存储器22内的参数,例如,当单元格中的嵌入式信息是电子表格中的公式的参数时(当嵌入时会改变公式的输出),或者当它是销售订单中的项目的价格时(当嵌入时会改变销售订单的小计);如果需要的话,这些新参数可以由嵌入式功能24读取并显示在显示器25上,以向用户提供有用的信息,诸如新的小计、拼写检查输出、项目的库存状态(例如,当提交销售订单时)。

[0185] 如所讨论的,用于字处理类型文档的文档存储器22中的x-y位置可以例如由page#、line#和字符数(character#)定义(参见图10,插入点1和插入点2的x-y位置)。类似地,例如,可以基于文档内单元格/字段的位置(例如,电子表格的column#、Row#和Page#)来定义用于表单、表格或电子表格的文档存储器22中的x-y位置。可替代地,可以基于来自给定已知位置的Tab和/或箭头键的数量来定义它。例如,可以基于来自表单中第一个字段(即“客户;工作”)的Tab数量来定义会计应用程序QuickBooks中销售订单中的字段。

[0186] 嵌入式功能可以读取x-y信息(参见图12和14中引用的流程图中的步骤2),并且然后根据实施例一(参见图15-图16中引用的示例流程图)或根据实施例二(参见图26中引用的MS Word的示例流程图)将插入点带到所期位置。然后,可以嵌入手写信息。例如,对于QuickBooks中的销售订单,模拟键盘键组合“Cntrl+J”会将插入点带到第一个字段,客户;工作;然后,模拟三个Tab键会将插入点带到“日期”字段,或者模拟八个Tab键会将插入点带到第一个“项目代码”字段。

[0187] 软件应用程序QuickBooks没有宏或编程能力。QuickBooks中的表单(例如,销售订单、账单或采购订单)和列表(例如,会计科目表和客户;工作列表)可以经由工具栏经由下拉菜单或经由快捷键来调用。因此,实施例一可以用来模拟键盘击键,以调用特定表单或特定列表。例如,可以通过模拟键盘键组合“Cntrl+N”来调用新发票,并且可以通过模拟键盘键组合“Cntrl+A”来调用会计科目表列表。可以通过模拟以下键盘击键来实现调用未定义相关联的快捷键的销售订单:

[0188] 1. “Alt+C”;从与“顾客”有关的工具栏菜单带来下拉菜单

[0189] 2. “Alt+0”;调用新的销售订单表单

[0190] 一旦调用表单,就可以将插入点带到指定的x-y位置,并且然后可以嵌入识别出的手写信息(即一个或多个命令和相关联的文本)。

[0191] 就用户而言,他可以将信息(例如,用于发布账单)写在预设表单(例如,与数字化垫板12或触摸屏11结合)上。用户在图4和图5中示出的步骤1“文档类型和首选项设置”(A)中选择诸如条目类型(表单或命令)、输入命令的顺序以及表单设置之类的参数。

[0192] 例如,以下序列手写命令将在2005年2月3日在OfficeMax处张贴采购办公用品的账单,总金额为\$45。如果已经在QuickBooks中设置了供应商OfficeMax,则可以省略作为与采购相关联的帐户的参数“办公用品”。可以从文档存储器22中读取信息,并且嵌入式功能24可以基于该信息来确定该账户先前是否已经设置,并且在显示器25上报告结果。例如,这可以通过尝试从“帐户”字段中剪切信息(即经由剪贴板)来实现,假设已经设置了帐户。可将剪贴板中的数据与预期结果进行比较,并在此基础上生成显示的输出。

[0193] 账单

[0194] 2005年2月3日

[0195] OfficeMax

[0196] \$45

[0197] 办公用品

[0198] 在诸如Excel之类的应用程序中,实施例一和实施例二中的一个或两个都可用于将插入点带到所期位置并嵌入识别出的手写信息。

[0199] 应用程序示例

[0200] 无线垫板

[0201] 无线垫板可用于将集成式文档传输到计算机,并可选地接收与传输的信息有关的信息。例如,可以在以下情况下使用它:

[0202] 1-在医生办公室填写表单

[0203] 2-填写运送包裹的空运提单

[0204] 3-在DMV处提交驾驶执照申请

[0205] 4-在汽车租赁公司或零售商店为客户服务

[0206] 5-在犯罪现场或事故现场做笔记

[0207] 6-异地下订单,例如在常规会议上。

[0208] 可以在数字化垫板12的顶部或使用触摸屏11等将手写信息插入到预先设计的文档(诸如订单表单、应用程序、表格或发票)中的指定位置。预先设计的表单存储在远程或附近的计算机中。手写信息可以经由无线链路同时发送到接收计算机。接收计算机将识别手写信息、对其进行解释、并将其以机器代码的形式存储到预先设计的文档中。可选地,接收计算机将准备响应并将其发送回发送垫板(或触摸屏),例如以帮助用户。

[0209] 例如,在会议上以订单的形式在垫板12上填写的信息可以在书写信息时发送到驻留在附近或远程服务器计算机中的记帐程序或数据库。继而,程序可以检查项目的状态,诸如成本、价格和库存状态,并实时发送信息以帮助接单。当接单指示订单已完成时,销售订单或发票可以在远程服务器计算机中过帐。

[0210] 图39是结合无线垫板的使用而示出的集成式编辑文档系统的示意图。无线垫板包括数字化垫板12、显示器25、数据接收器48、处理电路60、传输电路I50和接收电路II 58。数字化垫板接收来自书写笔10的触觉位置输入。传输电路I 50经由数据接收器48从数字化垫板12获取数据,并将其提供给远程处理单元的接收电路I 52。接收电路II 58经由远程电路的传输电路II 56从显示处理54捕获信息,并将其提供给显示器25的处理电路60。接收存储器I 52与数据接收存储器16通信,如前面解释的,数据接收存储器16与识别模块18交互,识别模块18继而与如前面解释的RHI处理器和存储器20以及文档存储器22交互。嵌入式标准和功能元件24与元件20和22交互,以修改主题电子文档,并将输出传送给显示处理单元54。

[0211] 远程通讯

[0212] 在不同位置处的两个或多个参与方之间的通信中,可以将手写信息合并到文档中,可以将信息识别并转换为机器可读的文本和图像,并作为“供审查”合并到文档中。如结合图6所讨论的(作为MS Word类型的文档的示例性实施例),可以以多种方式显示“供审查”信息。然后可以将“供审查”文档发送给一个或多个接收方(例如,经由电子邮件)。接收方可

以经由数字化触摸板12、经由触摸屏11或经由无线垫板批准部分或全部修订和/或进一步手写修订(如发送方所做的那样)。然后可以再次发送该文档“供审查”。该过程可能会继续进行,直到所有修订都合并/结束为止。

[0213] 经由传真进行修订

[0214] 页面上的手写信息(具有或不具有机器打印的信息)可以经由传真发送,并且增强为多功能装置(打印机/传真机、字符识别扫描仪)的接收传真机可以将文档转换为机器可读的文本/图像用于指定的应用程序(例如Microsoft®Word)。可以基于页面上标记的指定修订区域来区分修订与原始信息,并相应地进行转换(例如,通过在修订处划线或划圆圈)。然后可以将其发送(例如,经由电子邮件)“供审查”(如上面讨论的,在“远程通信”下)。

[0215] 使用手机的集成式文档编辑器

[0216] 可以在数字化垫板12上输入手写信息,由此数字化垫板12上的位置对应于手机显示器上的位置。可替代地,可以在用作数字化垫板以及显示器的触摸屏上输入手写信息(即,类似于图38中引用的触摸屏11)。手写信息可以是新信息,或者现有存储信息的修订(例如,电话号码、联系人姓名、待办事项、日历事件、图像照片等)。手写信息可以由识别元件18识别、由RHI元件20处理、并且然后嵌入到文档存储器22中(例如,在特定联系信息的特定存储位置中)。例如,可以通过直接访问文档存储器中的位置(例如,特定的联系人姓名)来实现手写信息的嵌入;但是,通过其嵌入识别出的手写信息的方法可以由手机制造商在OEM级别处确定。

[0217] 集成式文档编辑器在手写信息认证中的使用

[0218] 可以预先设置唯一的表示,诸如签名、印章、指纹或任何其他绘制图案,并将其作为词汇的一部分的单元或作为新字符馈送到识别元件18中。当手写信息被识别为将被放置在例如数字化垫板12(图1)或触摸屏11(图38)的特定预期x-y位置中的这些预设单元之一时,验证或部分验证将通过。如果识别出的单元与预设的预期单元之间不匹配,则验证将失败。这对于认证文档(例如,电子邮件、投票或表单)以确保文档的作者/发送者是预期的发送者很有用。其他示例用于验证和访问银行信息或信用报告。唯一的预设模式可以是以下两者之一或两者:1)存储在属于用户的特定平台中,和/或2)存储在远程数据库位置中。应当注意,不必在文档中公开唯一的预设模式(例如签名)。例如,当签名的验证通过时,嵌入式功能24将例如将单词“OK”嵌入文档的签名行/字段中。

[0219] 在美国专利申请号15/391,710(其是美国专利号9,582,095的延续)和美国专利申请号13/955,288中讨论了讨论自动计算文档位置的计算装置和方法,在该文档位置处自动应用通过用户输入在计算装置的触摸屏上传送的用户命令。

[0220] 所公开的实施例还涉及与一个或多个图形对象的所显示的表示的简化的用户交互。简化的用户交互可以利用计算装置的触摸屏,并且可以包括使用手势来指示图形对象的一个或多个参数中的一个或多个所期改变。参数可以包括存储在计算装置的存储器中或由计算装置的函数计算出的线长、线角或圆弧半径、大小、表面积或图形对象的任何其他参数中的一个或多个。这些一个或多个参数的改变是由计算装置的函数基于触摸屏上的用户交互来计算的,并且这些计算出的改变可以被计算装置的其他函数用来计算其他图形对象中的改变。

[0221] 如上所述,文档可以是任何类型的电子文件、文字处理文档、电子表格、网页、表

单、电子邮件、数据库、表格、模板、图表、图形、图像、对象或这些类型的文档的任何部分,诸如文本块或数据单位。应当理解,文档或文件可以用于任何适当的应用中,包括但不限于计算机辅助设计、游戏和教育材料。

[0222] 所公开的实施例的目的是允许用户在简短的交互式屏幕教程之后在移动中或现场快速编辑计算机辅助设计(CAD)绘图;不需要诸如操作CAD绘图应用程序(例如AutoCAD®软件)所需的技能/专业知识。另外,所公开的实施例可以通过提供更简单和更快的用户交互来节省显著的时间,同时避免了与专业人士的修订迭代。典型的用户可能包括但不限于建筑商和承包商、建筑师、室内设计师、专利律师、发明家和制造工厂经理。

[0223] 公开的实施例的另一个目的是允许用户使用为编辑CAD绘图提供的相同手势集来编辑各种常用文档格式(诸如doc和docx格式)的图形文档。应该注意的是,CAD绘图应用程序中常用的一些命令(例如AutoCAD®软件、诸如用于将半径应用于线或添加倒角的命令)在文字处理应用程序中或桌面发布应用程序中不可用。

[0224] 公开的实施例的进一步的目的是允许用户使用相同的手势,基于在计算装置的触摸屏上的用户交互,以各种文档格式来创建CAD绘图和图形文档,包括诸如DXF格式和doc和docx格式的CAD绘图格式。

[0225] 所公开实施例的又一个目的是允许用户与触摸屏上的图形对象的三维表示进行交互,以指示一个或多个图形对象的一个或多个参数的所期改变,这继而将导致计算装置的功能以自动影响指示出的改变。

[0226] 通过参考附图集合(图40A-图58B),将更好地理解本文公开的实施例的这些、其他实施例和其他特征,这些附图集合应被视为说明性示例而不是限制性的。可以将图40A-图52D、图54A-54F和图56-58A视为应用程序教程的一部分,以使用户熟悉这些图中讨论的手势的使用。

[0227] 虽然参考与图形对象的二维表示的用户交互来描述图41A至图52D所公开的实施例,但是应当理解,所公开的实施例也可以参考与图形对象的三维表示的用户交互来实施。

[0228] 第一,用户通过绘制字母或选择表示所期命令的图标来选择命令(例如,在图42A-图42D中讨论的改变线长的命令)。第二,计算装置识别命令。然后,响应于用户与触摸屏上的图形对象的所显示的表示的交互以指示一个或多个参数(诸如,线长)的所期改变,计算装置自动致使指示出的参数引起所期改变,并且当适用时,也会自动影响图形对象位置的改变,并且因此进一步地影响存储图形的存储器中的其他图形对象。

[0229] 图形对象的参数的所期改变,即其值(和/或其形状,当图形对象的形状为参数时,诸如从直线对象改变为分段直线对象,或从一种形状逐渐改变为另一种形状,诸如从圆形/球体改变为椭圆形,并且反之亦然)的增大或减小,可以通过沿着在触摸屏上绘制的手势的定位位置的改变来指示(例如,如图42A-图42B中示出的),并且在此期间当用户继续绘制手势时,计算装置逐渐且自动地应用所期改变。从用户的角度看,在绘制手势的同时,参数的值似乎也在改变。

[0230] 存储在装置存储器中的主题绘图或其一部分(在此定义为“图形矢量”)可以作为二维表示(在此定义为“矢量图像”)显示在触摸屏上,用户可以与之交互,以便传达图形对象的一个或多个参数(诸如线长、线角或圆弧半径)的所期改变。如上面讨论的,计算装置自动在图形对象中自动引起这些所期改变,并且在适用时,也在其位置中以及进一步地在图

形矢量内的其他图形对象的参数和位置中自动引起这些所期改变,这些改变可能是由用户指示出的图形对象的改变引起的。图形矢量可以可替代地在触摸屏上表示为三维矢量图像,以便允许用户在图形对象的实际三维表示中查看/审查图形对象的参数改变的影响,而不是在查看二维表示时尝试可视化效果。

[0231] 此外,用户可以与触摸屏上的三维矢量图像进行交互,以指示一个或多个图形对象的一个或多个参数的所期改变,例如,通过指向/触摸或敲击三维表示的几何特征,诸如在表面上或拐角处,这将导致计算装置自动改变图形矢量的一个或多个图形对象的一个或多个参数。这样的用户与几何特征的交互可以例如沿着表面长度、宽度或高度、沿着两个连接表面的边缘(例如,沿着连接顶表面和侧表面之一的边缘)、在斜角/修边拐角的内部或内部的一个或多个表面内、倾斜表面(例如,斜坡)、或者在弧形拐角内部或外部的弧形表面内。

[0232] 通过首先使用存储(并在xyz坐标轴系统中定义)在装置存储器(在本文中称为“位置”)中的一个或多个点/位置,并将它们与用户可以与之交互以传送图形对象的所期改变的矢量图像的几何特征相关,可以实现用户与触摸屏上的三维矢量图像的几何特征的交互与存储在装置存储器中的矢量图形的大小和/或几何形状之间的相关性。本文的位置定义为使得该位置的改变、或从该位置延伸/分支的线(直线、弧形线或分段线)的存储或计算出的参数(诸如长度、半径或角度)的改变(在本文中定义为“变量”)可以用作一个或多个函数中的变量(或作为变量之一),其能够计算由于该变量的改变而导致矢量图形的大小和/或几何形状的改变。可以在感兴趣的区域内定义用户交互,该感兴趣的区域是触摸屏上的几何特征的区域,用户可以在该区域内进行手势/交互;该区域例如可以是立方体的整个表面、或者是不包括靠近中心的区域的整个立方体表面。另外,响应于检测到手指在预定/预期方向(或在预定/预期方向之一)上或在该区域内的预定/预期触摸和/或敲击,计算装置自动确定/识别相关变量并自动执行其一个或多个相关联的函数以自动影响用户传达的一个或多个所期改变。

[0233] 例如,矩形或立方体的边缘/拐角的位置是可以用作函数(或其中一个函数)中的变量的位置,该函数能够计算由于该变量改变而引起的矩形或立方体的几何形状的改变。类似地,立方体的两个边缘/拐角之间(即,两个位置之间)的线的长度或立方体的两个连接表面之间的角度可以用作变量。或者,可以将圆或球体的中心点用作从该处延伸圆或球体的半径的“位置”;在该示例中,半径可以是函数的变量,当用户与球体交互(例如,触摸)时,该函数能够计算圆的周长和表面积或球体的周长、表面和体积。类似地,当用户与对称矢量图像进行交互时,从具有对称几何形状(诸如立方体或管状体)的矢量图形的中心点延伸的线的长度、或从中心点延伸线末端处的位置可用作能够计算对称矢量图形大小的改变或其几何形状的改变的函数(或函数之一)的变量(或变量之一)。或者,在其显示的多个表面之一(诸如,圆锥的基底的表面)中具有对称性的三维矢量图形中,可以定义两个位置,第一位置位于基底处的表面的中心点处,并且第二位置是从该位置延伸到圆锥顶部的线的边缘;在此示例中,变量可以是第一位置以及从第一个位置延伸到圆锥顶部的线的长度,当用户与表示圆锥的矢量图像进行交互时,可以在能够计算圆锥大小和几何形状的改变的一个或多个函数中使用这些变量。或者,在触摸屏上表示为三维矢量图像的复杂或非对称图形矢量(用户可以与之交互以传达图形矢量的改变)可以将其划分为装置存储器中的多个部分

图形矢量(在触摸屏上表示为一个矢量图像),每个部分图形矢量均由能够计算其大小和几何形状的改变的一个或多个函数表示,由此,图形矢量的大小和几何形状可以由计算装置基于部分图形矢量的总和来计算。

[0234] 在一个实施例中,响应于用户“推动”(即,实际上是触摸)或敲击图形矢量的所显示的表示的几何特征处(即,在矢量图像处),计算装置自动增大或减小图形矢量或图形特征上表示的一个或多个参数的大小。例如,在立方体的拐角或坡道表面的所显示的表示处触摸或敲击会导致计算装置分别自动减小或增大立方体(图54A-图54B)或斜坡的下降角度/倾斜角度的大小。

[0235] 类似地,响应于在球体的所显示的表示处的任何地方触摸或敲击,计算装置分别自动减小或增大球体的半径,这继而分别减小或增大球体的周长、表面积和体积。或者,当用户继续挤压/保持矢量图像的几何特征时,响应于继续“挤压”(即保持/触摸)表示图形矢量中的特征的矢量图像的几何特征,诸如管状体或立方体顶部的侧边缘,计算装置自动将该图形矢量的一个或多个外边缘逐渐聚集在一起。类似地,当用户继续敲击或触摸矢量图像的几何特征时,响应于用户敲击或保持/触摸几何特征的顶表面,计算装置分别自动且逐渐地使几何特征的外边缘向外或向内移动。或者,响应于在顶表面的中心点处或附近的触摸(注意,此处的关注区域靠近中心,在先前示例中从关注区域中排除了该中心),计算装置自动创建具有以该中心点为中心的半径的条纹(或其他预定形状),并且继续触摸或敲击(在触摸屏上的任意位置)将分别致使计算装置自动且逐渐减小或增大条纹的半径。

[0236] 在另一个实施例中,首先响应于用户指示所期命令,计算装置识别该命令。然后,用户可以在矢量图像的显示的几何特征上打手势以指示矢量图形中的所期改变。例如,当用户分别在圆角/圆弧表面处(或触摸屏上的任何位置)继续触摸或敲击时,响应于在拐角的表面的所显示的表示处继续“推动”(即触摸)或敲击,在用户指示出用于添加圆角(在内角的表面处)或圆弧(在外角的表面处)的命令以及计算装置识别出的命令后,计算装置会自动对拐角进行倒圆(如果拐角尚未倒圆),并且然后导致圆角/圆弧的半径的值(以及相邻线对象的位置)增大或减小。或者,在计算装置识别出用于改变线长的命令之后(例如,在用户触摸表示该命令的不同图标之后),响应于手指在显示的立方体的表面上的任意位置向右或向左移动(指示宽度分别从立方体的表面的右边缘或左边缘的所期改变),随后继续触摸或敲击(在触摸屏上的任意位置),计算装置在用户继续触摸或敲击时分别从表面的右边缘或左边缘自动减小或增大立方体的宽度。类似地,响应于手指在立方体表面上的向上或向下移动,随后继续触摸或敲击触摸屏上的任意位置,计算装置在用户继续触摸或敲击时分别从表面的顶部边缘或底部边缘自动减小或增大立方体的高度。此外,响应于敲击或触摸沿着立方体的图形图像的两个连接表面的边缘附近的点,计算装置自动增大或减小两个连接表面之间的角度。或者,在计算装置识别出要在图形图像的表面上插入盲孔和点的命令之后(例如,在检测到该点处的长按之后,指示在该处要钻孔的表面的点),响应于连续的敲击或触摸(在触摸屏上的任何地方),计算装置分别在图形矢量中逐渐且自动地增大或减小孔的深度并更新矢量图像。类似地,响应于识别在矢量图像的表面上用户指示出的点处钻通孔的命令,计算装置自动在矢量图形中插入通孔,并用插入的通孔更新矢量图像。此外,响应于在沿着孔的圆周的点处的敲击或触摸,计算装置自动地增大或减小孔的半径。或者,响应于触摸孔的内表面,计算装置自动调用标准螺纹的选择表/菜单,用户可以从中

选择标准螺纹以施加到孔的外表面。

[0237] 图40A-图40D涉及插入线的命令。它们示出了用户与触摸屏之间的交互,由此用户在两个点A和B之间徒手绘制了线3705(图40B)。在一些实施例中,在绘制线的同时显示线3710的估计距离。响应于用户手指从触摸屏抬起(图40C),计算装置在触摸屏上由点A和B表示的存储器位置处将直线对象自动插入装置存储器中,在其中存储了绘图,并在触摸屏上显示直线对象3715及其实际距离3720。

[0238] 图41A-图41C涉及删除对象的命令。用户通过触摸所期对象3725来选择它(图41A),并且然后可以绘制命令指示符3730(例如字母“d”)以指示命令“删除”(图41B)。作为响应,计算装置识别该命令并删除对象(图41C)。应当注意,用户可以通过听觉信号等通过选择表示命令的图标来指示该命令。

[0239] 图42A-图42D涉及改变线长的命令。首先,用户通过触摸线3735来选择它(图42A),并且然后可以绘制命令指示符3740(例如字母“L”)以指示所期命令(图42B)。应当注意,在绘制命令指示符3740之前选择线3735是可选的,例如,以查看其距离或复制或剪切它。然后,响应于从线3735的点3745开始的在触摸屏上用户选出的定位位置的每个逐渐改变,计算装置自动地致使存储在装置存储器中的线长的每个相应逐渐改变,并更新显示框3750上的长度(图42B-图42C)。

[0240] 图43A-图43D涉及改变线角的命令。用户可以可选地首先选择线3755(图43A),并且然后可以绘制命令指示符3760(例如字母“a”)以指示所期命令(图43B)。然后,以类似于改变线长的方式,响应于从线3755的边缘3765开始的在触摸屏上用户选出的定位位置(上或下)中的每个逐渐改变,计算装置自动地致使存储在装置存储器中的线角度的每个相应的逐渐改变,并更新例如相对于装置存储器中的x轴的线的角度,并且还更新显示框3770上的角度(图43B-图43C)。

[0241] 应当注意,如果用户同时指示两个命令:在绘制以上两段中讨论的手势之前改变线长并改变线角(例如,通过选择两个不同的图标,每个图标表示命令中的一个),然后,计算装置将基于手势的移动方向自动致使线的长度和/或角度逐渐改变,并将相应地在触摸屏上用户选出的定位位置的逐渐改变中的每个处更新显示框上的长度和角度中的任一者或两者的值。

[0242] 图44A-图44D涉及将半径应用于线或改变A和B之间的圆弧半径的命令。用户可以可选地首先选择显示的线或圆弧,在此示例中为线3775(图44A),并且然后可以绘制命令指示符3780(例如字母“R”)以指示所期命令(图44B)。然后,以类似于改变线长或线角的方式,响应于在触摸屏上用户选出的定位位置跨显示的线/圆弧3785中的每个逐渐改变,从沿着显示的线/圆弧3775的位置开始,计算装置自动致使存储在装置存储器中的图形中的图形的线/圆弧的半径的每个相应逐渐改变,并在显示框3790上更新圆弧的半径(图44C)。

[0243] 图45A-图45C涉及使一条线与另一条线平行的命令。首先,用户可以绘制命令指示符3795(例如字母“N”)以指示所期命令,并且然后触摸参考线3800(图45A)。然后,用户选择目标线3805(图45B)并抬起手指(图45C)。响应于手指被抬起,计算装置自动将装置存储器中的目标线3805改变为与参考线3800平行,并更新触摸屏上显示的目标线(图45C)。

[0244] 图46A-图46D涉及添加圆角(在拐角的2D表示处或在拐角的内表面的3D表示处)或圆弧(在拐角的外表面的3D表示处)的命令。首先,用户可以绘制命令指示符3810来指示所

期命令,并且然后触摸要向其应用圆角的拐角3815(图46A)。作为响应,计算装置将尖角3815转换为圆角3820(具有默认半径值)并放大该拐角(图46B)。然后,响应于在触摸屏上用用户选出的定位位置在沿显示的圆弧3825的某个位置处跨该显示的圆弧3825中的每个逐渐改变,计算装置致使存储在装置存储器中的圆弧的半径及其由A和B表示的存储器中的位置中的每个相应逐渐改变,使得圆弧与相邻的线3830和3835相切(图46C)。接下来,用户触摸屏幕,并且作为响应,计算装置将图形缩小到其原始缩放百分比(图46D)。否则,即使在抬起手指之后,用户也可能会指示出半径的额外改变。

[0245] 图47A-图47D涉及添加倒角的命令。首先,用户可以绘制命令指示符3840以指示所期命令,并且然后触摸向其应用倒角/斜角的所期拐角3845(图47A)。作为响应,计算装置在触摸屏上修剪由A和B表示的两个位置之间的拐角,并将高度H和宽度W设置为默认值,并且因此将角度 α 也设置为默认值(图47B)。然后,响应于在触摸屏上用户选出的定位位置中的每个逐渐改变(以与线3850和/或线3855平行运动),计算装置分别致使宽度W和/或高度H的逐渐改变,如存储在装置存储器以及存储在存储器中的位置A和B中,并更新它们的所显示的表示(图47C)。接下来,用户触摸屏幕,并且作为响应,计算装置将图形缩小到其原始缩放百分比(图47D)。否则,即使在手指抬起之后,用户也可以指示参数W和/或H的额外变化。

[0246] 图48A-图48F涉及修剪对象的命令。首先,用户可以绘制命令指示符3860以指示所期命令(图48A)。接下来,用户触摸目标对象3865(图48B),并且然后触摸参考对象3870(图48C);应当注意,这些步骤是可选的。然后,用户移动参考对象3870以指示目标对象3865中的所期修剪(图48D-图48E)。然后,响应于手指从触摸屏抬起,计算装置自动将所期修剪3875应用于目标对象3865(图48F)。

[0247] 图49A-图49D涉及移动弧形对象的命令。首先,用户可以可选地选择对象3885(图49A),并且然后绘制命令指示符3880以指示所期命令,并且然后触摸显示的目标对象3885(图49B)(此时选择了对象)并移动它,直到圆弧3885的边缘3890位于或接近线3897的边缘3895为止(图49C)。然后,响应于手指从屏幕上抬起,计算装置自动移动圆弧3885,使得其与边缘相交的线3897相切(图49D)。

[0248] 图50A-图50D涉及“不捕捉”命令。首先,用户可以触摸命令指示符3900以指示所期命令(图50A),并且然后用户可以触摸所期交点3905以取消捕捉(图50B)。然后,响应于手指从触摸屏抬起,计算装置在交点3905处自动应用不捕捉3910并放大该交点(图50C)。再次触摸会致使计算装置将图形缩小到其原始缩放百分比(图50D)。

[0249] 图51A-图51D示出了使用“不捕捉”命令的另一示例。首先,用户可以触摸命令指示符3915来指示所期命令(图51A)。接下来,用户可以绘制命令指示符3920(例如字母“U”)以指示改变线长的所期命令(图51B)。然后,响应于在触摸屏上用户选出的定位位置的每个逐渐变化,从线3930的边缘3925开始并在触摸屏上的位置3935处结束,跨线3940,如果计算装置将捕捉操作设置为默认操作,则计算装置自动取消捕捉交点3945或避免捕捉交点3945。

[0250] 图52A-图52D示出了使用用于修剪对象的命令的另一示例。首先,用户可以绘制命令指示符3950以指示所期命令(图52A)。接下来,用户移动参考对象3955以指示目标对象3960中的所期修剪(图52B-图52C)。然后,响应于用户手指从触摸屏抬起,计算装置自动将所期修剪3965应用于目标对象3960(图52D)。

[0251] 可以将复制和剪切图形对象的命令添加到上面讨论的手势集中,并例如通过选择

一个或多个图形对象来执行(例如,如图42A中示出的),并且然后用户可以绘制命令指示符或触摸触摸屏上相关联的不同图标以指示所期命令,以进行复制或剪切。用于粘贴的命令也可以被添加,并且例如可以通过绘制命令指示符(诸如字母“P”)(或通过触摸表示该命令的不同图标)来执行,并且然后指向触摸屏上的位置,它表示在存储器中粘贴剪贴板内容的位置。复制、剪切和粘贴命令可能有用,例如,在复制表示特征(诸如浴室标签)的CAD绘图的一部分并将其粘贴到表示整修场地的第二浴室的绘图的另一个位置处。

[0252] 图53是用户界面的示例,其中图标与上图中讨论的可用用户命令相对应,并且每个不同图标的“手势帮助”指示字母/符号,其可以被绘制以指示命令,而不是通过其表示命令来选择图标。

[0253] 图54A-图54B示出了在与立方体的矢量图形的三维表示进行交互之前和之后的示例。响应于用户在预定时间段内触摸表示立方体的图形矢量的矢量图像3975的拐角3970(图54A),计算装置将拐角3970处的触摸解释/识别为按比例减小立方体尺寸的命令。然后,响应于在拐角3970处的持续触摸,计算装置以相同速率自动并逐渐减小分别在3977、3980和3985处显示的矢量图形中立方体的长度、宽度和高度,并更新矢量图像4005中显示的长度3990、宽度3950和高度4000(图54B)。

[0254] 图54C-图54D示出了与球体的矢量图形的三维表示进行交互之前和之后的示例。响应于在预定时间段内表示球体的图形矢量的在球体的矢量图像4015上的点4010处或任何地方的持续触摸(图54C),计算装置将在点4010处的触摸解释/识别为减小球体半径的命令。然后,响应于在点4010处的持续触摸,计算装置自动并逐渐减小球体的矢量图形的半径,并在触摸屏上更新矢量图像4017(图54D)。

[0255] 图54E-图54F示出了在与斜坡的矢量图形的三维表示进行交互之前和之后的示例。响应于用户在预定时间段内在表示斜坡的图形矢量的斜坡的矢量图像4035的点4020或沿着沿斜坡的矢量图像4035的基底4030的边缘4025的任意点4025处触摸(图54E),计算装置将触摸解释/识别为增加图形对象中的倾斜角度4040和减小基底4030的距离4045的命令,使得沿斜坡的距离4050保持不变。然后,响应于在点4020处的持续触摸,计算装置自动且逐渐地增加倾斜角度4040并减小图形矢量中的基底4030的距离4045,使得沿斜坡高度的距离4050保持不变,并将显示的倾斜角度4040和距离4045更新为矢量图像4065中的倾斜角度4055和距离4060(图54F)。类似地,响应于敲击,在点4020处,计算装置可以被配置为自动且逐渐地减小倾斜角度4040并增加距离4045,使得沿着斜坡的距离4050将保持不变。

[0256] 图55A-图55B示出了下面讨论的用于文本编辑、选择模式的用户界面菜单的示例。

[0257] 图56是在命令模式下标记文本的手势示例。首先,用户例如通过触摸表示所期命令的图标4055来指示该命令,诸如用于下划线的命令。然后,响应于用户从右到左或从左到右在A和B之间的徒手绘制线4060,以指示在存储器中要在文本上加下划线的位置,取决于用户预定义的首选项,当用户继续绘制手势时或当手指从触摸屏抬起时,计算装置在指示出的位置处给文本自动加下划线并在触摸屏上显示带下划线的文本的表示。

[0258] 图57是在命令模式下标记文本的手势的另一个示例。首先,用户例如通过触摸表示该命令的图标4065来指示所期命令,诸如移动文本的命令。然后,响应于用户从右到左或从左到右在A和B之间徒手绘制锯齿形线4070,以指示在存储器中选择要移动的文本的位置,取决于用户预定义的首选项,当用户继续绘制手势时或当手指从触摸屏抬起时,计算装

置会自动选择存储器中指示出的位置处的文本,并在触摸屏上突出显示该文本。此时,计算装置将自动切换到数据输入模式。接下来(未示出),响应于用户指向触摸屏上的位置(该位置指示存储器中要粘贴所选文本的位置),计算装置从该指示出的位置开始自动粘贴所选文本。一旦粘贴文本,计算装置就将自动恢复为命令模式。

[0259] 在一个实施例中,计算装置调用命令模式或数据输入模式;当识别出旨在应用于已经存储在存储器中并显示在触摸屏上的文本或图形的命令时,调用命令模式,并且当识别出插入或粘贴文本或图形的命令时,调用数据输入模式。在命令模式下,数据输入模式被禁用,以允许在计算装置的触摸屏上无限制/无约束的用户输入,以便指示显示的文本/图形的位置,在该位置处:应用一个或多个用户预定义的命令,并且在数据输入模式下,禁用命令模式,以使能指向触摸屏上的位置,该位置指示存储器中要插入文本、插入绘制的形状(诸如线)或粘贴文本或图形的位置。命令模式可以设置为默认模式。

[0260] 当在命令模式下时,计算装置不会将用户在所显示的文本或图形上的用于指示存储器中位置(在此处应用一个或多个预定义的命令)所进行的绘制(在此定义为“标记手势”)解释为用于插入线的命令,并且计算装置不会将在绘制标记手势或仅触摸触摸屏上的位置时停止移动解释为指示存储器中要插入文本或图形的位置的定位,这是因为在此模式下,数据输入模式被禁用。然而,在一个实施例中,当处于数据输入模式时,仅在将手指从触摸屏抬起之后,计算装置才将这种位置解释为指示存储器中的插入位置,以进一步提高鲁棒性/用户友好性;下面将进一步讨论该特征在控制缩放功能方面的优势。用户可以在触摸屏上的所显示文本上徒手绘制标记手势以指示应该应用所期命令(诸如加粗、加下划线、移动或删除)的文本字符在存储器中的所期位置,或在所显示图形上(即,在矢量图像上)徒手绘制标记手势以指示应该应用所期命令(诸如选择、删除、替换、改变对象颜色、色差、大小、样式或线条粗细)的图形对象在存储器中的所期位置。

[0261] 在绘制标记手势之前,用户可以通过从触摸屏上的条形菜单中选择表示命令的不同图标来定义命令,例如,如图53中示出的。可替代地,用户可以通过绘制表示命令的字母/符号来定义所期命令;但是,在这种情况下,在绘制字母/符号时可以禁用命令模式和数据输入模式,以允许在触摸屏上的任意位置无限制地徒手绘制字母/符号,使得不会将字母/符号的绘制解释为要插入的标记手势或绘制的特征(诸如绘制的线),并且从触摸屏抬起手指将不会被解释为插入或粘贴数据。

[0262] 应当注意,在所显示的文本/图形上绘制标记手势以指示在存储器中将用户指示出的命令应用于文本/图形所在的所期位置,可以在单个步骤中完成,并且如果需要的话,可以在一个或多个时间间隔下完成,例如,如果用户将他/她的手指从触摸屏上抬起预定的时间段,或者在其他预定条件下(诸如两次敲击之间),在此期间用户可能例如希望先审查另一文档中的一部分,然后再决定是在休息时间之前从最后指示出的位置继续标记额外的所显示文本/图形还是在其他所显示文本/图形上进行标记,或者简单地结束标记。还应该注意,标记手势可以以任何形状徒手绘制,诸如锯齿形(图57)、跨显示的文本/图形的线(图56)或显示的文本/图形上方或下方的线。用户还可以选择在标记手势被绘制时显示它,并沿该手势(或沿其的任何位置)向后绘制,以撤消对显示的文本/图形的一个或多个先前标记区域指示出的文本/图形应用的一个或多个命令。

[0263] 在另一实施例中,特别是在文本编辑但不限于文本编辑中,响应于在命令模

式下时在触摸屏上绘制手势以标记显示的文本或图形,并且在绘制手势之前未选择命令,计算装置自动调用选择模式,在手指从触摸屏抬起时在触摸屏上选择标记/指示出的文本/图形,并自动调用图标集,每个图标表示不同的命令,这些图标由所选文本/图形布置在菜单和/或工具提示中(图55A-图55B)。在这些示例中,当用户选择一个或多个显示的图标时,计算装置自动将对应的一个或多个命令应用于所选文本。用户可以通过简单地关闭屏幕来退出选择模式,作为响应,计算装置将自动返回到命令模式。在移动所选文本之后(如果用户已指示移动文本的命令,则指向触摸屏上的表示存储器中要移动所选文本的位置的定位,并且然后抬起手指),计算装置也将自动恢复到命令模式。与在命令模式下一样,在选择模式下禁用数据输入模式,以允许无限制/无约束地绘制标记手势以标记所显示的文本或图形。例如,当用户希望专注于文本的特定部分并在结束对文本的那部分进行编辑之前执行一些尝试和错误时,选择模式可能会很有用。当所选文本是单个单词时,用户可以例如指示用于建议同义词、将该单词大写或将其字体改变为全部大写的命令。

[0264] 图58A-图58B示出了在绘制手势以标记文本时自动缩放文本的示例,如下面讨论的。

[0265] 在另一实施例中,在命令模式或数据输入模式下时,或者在选择模式期间(在手指从触摸屏抬起之前)绘制标记手势时,响应于在绘制标记手势或诸如要插入的线之类的形状时检测到触摸屏上两个位置之间的速度减小或增加,计算装置分别自动放大或缩小触摸屏上显示的文本/图形中沿标记手势或绘制线最接近当前位置的部分。另外,响应于在命令模式或数据输入模式下在预定时间段内检测到在触摸屏上用户选出的位置无移动,计算装置自动放大触摸屏上显示的文本/图形中最接近所选位置的部分,并随着用户继续指向该所选位置而继续逐渐放大至多最大预定缩放百分比;此特征可能非常有用,尤其是沿手势或沿绘制线的起点和终点处或附近,这是因为用户可能需要查看其附近的更多详细信息,以便指向更接近所期所显示文本字符/图形对象或其位置处;自然地,手指在起点(在绘制手势或线之前)以及潜在终点处都处于静止状态。如所讨论的,在一个实施例中,当处于数据输入模式时,在触摸屏上静止的手指(或书写工具)将不被解释为在存储器中要插入文本/图形的插入位置,直到手指(或书写工具)从触摸屏抬起之后为止,并且因此,用户在接近所期位置时可能会使其手指周期性地静止(以放大)。此外,响应于检测到持续敲击,计算装置可以被配置为在用户继续敲击时自动缩小。

[0266] 公开的实施例可以进一步提供一种设施,该设施允许用户指定用于与图形对象的所显示的表示进行交互的定制手势。可以提示用户选择与所期手势相关联的一个或多个参数。在一些方面,可以向用户呈现可用参数的列表,或者可以向用户提供输入定制参数的设施。一旦指定了参数,就可以提示用户将指示指定参数的一个或多个改变的一个或多个所期手势与矢量图像中的几何特征相关联;在一些方面,可以提示用户输入指示指定参数的值增加的所期手势,并且然后输入指示指定参数的值减小的另一个所期手势,在其他方面,可以提示用户将指示其形状的一个或多个改变的一个或多个所期手势相关联(当一个或多个图形对象的形状/几何形状是指定参数时),并且在其他方面,可以提示用户将绘制的手势的一个或多个移动方向与几何特征中的特征相关联,等等。然后,计算装置可以将一个或多个定制参数与一个或多个函数相关联,或者可以向用户呈现可用函数的列表,或者可以向用户提供用于指定一个或多个定制函数的设施,使得当用户在同一矢量图像内或另一矢

量图像内的其他类似几何特征内输入一个或多个指定手势时,计算装置将自动影响在计算装置的存储器中由矢量图像表示的矢量图形中的指示出的改变。

[0267] 注意,本文描述的实施例可以单独使用或以其任何组合使用。应当理解,以上描述仅是实施例的说明。在不脱离实施例的情况下,本领域技术人员可以设计出各种替代和修改。因此,本实施例旨在涵盖落入所附权利要求的范围内的所有这样的替代、修改和变化。

[0268] 当结合附图阅读时,鉴于前面的描述,各种修改和改编对于相关领域的技术人员而言将变得显而易见。然而,所公开实施例的教导的所有这些和类似修改仍将落入所公开实施例的范围内。

[0269] 本文描述的不同实施例的各种特征可以彼此互换。可以将各种所描述的特征以及任何已知的等同物混合并匹配以构造根据本公开的原理的额外的实施例和技术。

[0270] 此外,在没有对应使用其他特征的情况下,可以有利地使用示例性实施例的一些特征。像这样,前面的描述应被认为仅是对所公开的实施例的原理的说明,而不是对其的限制。

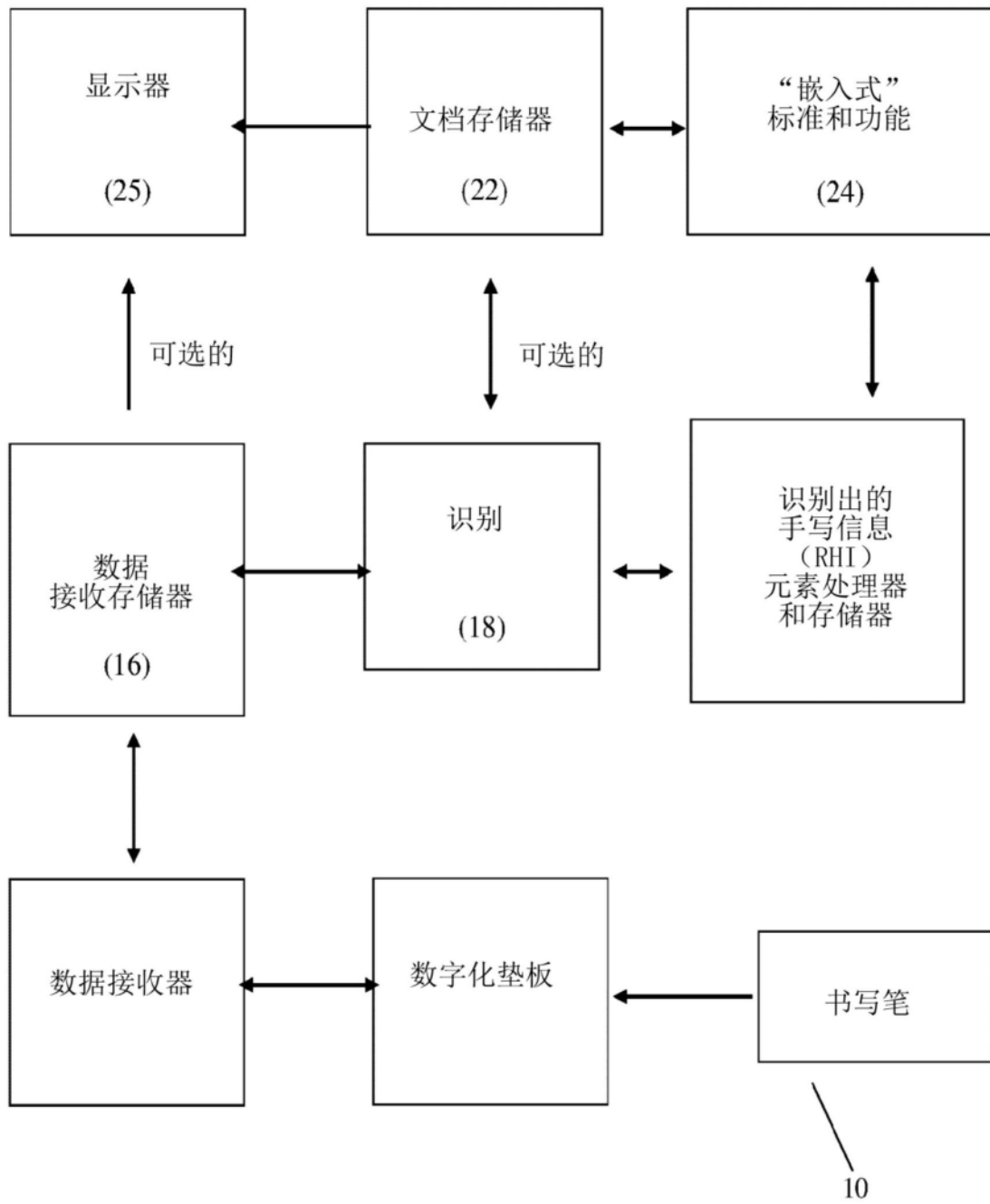


图1

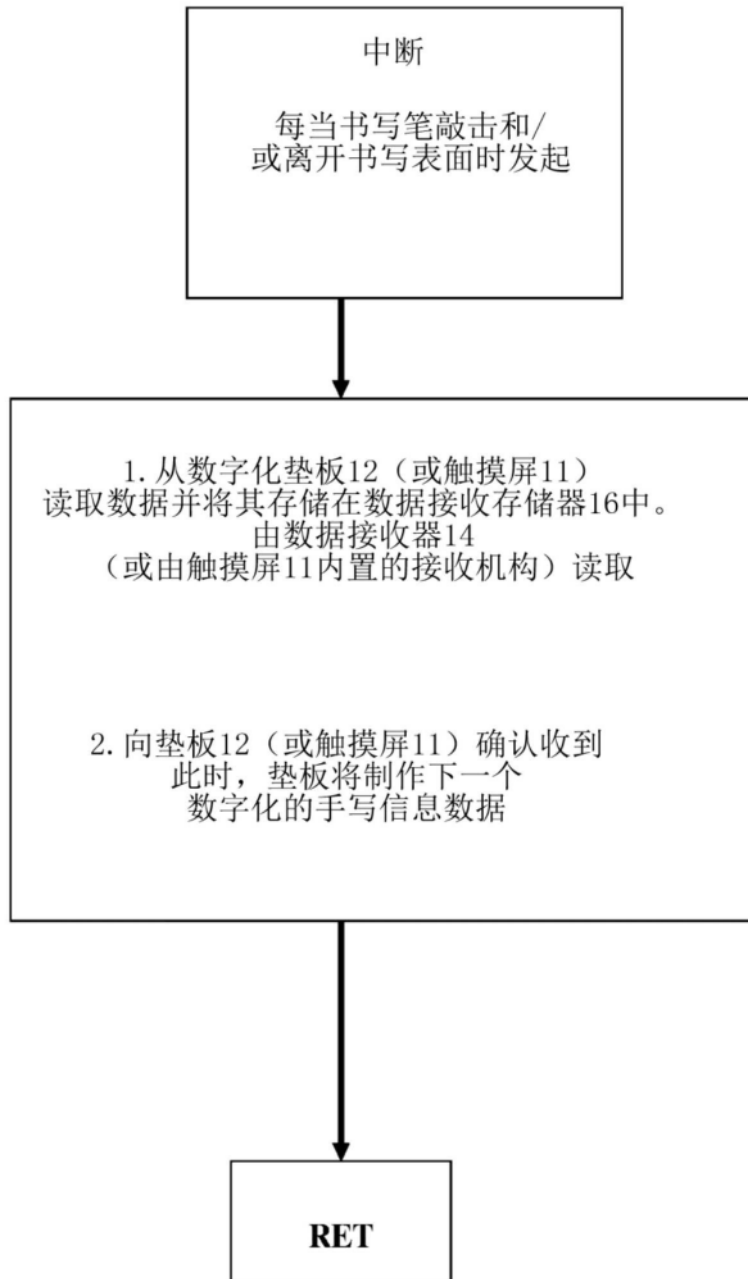


图2

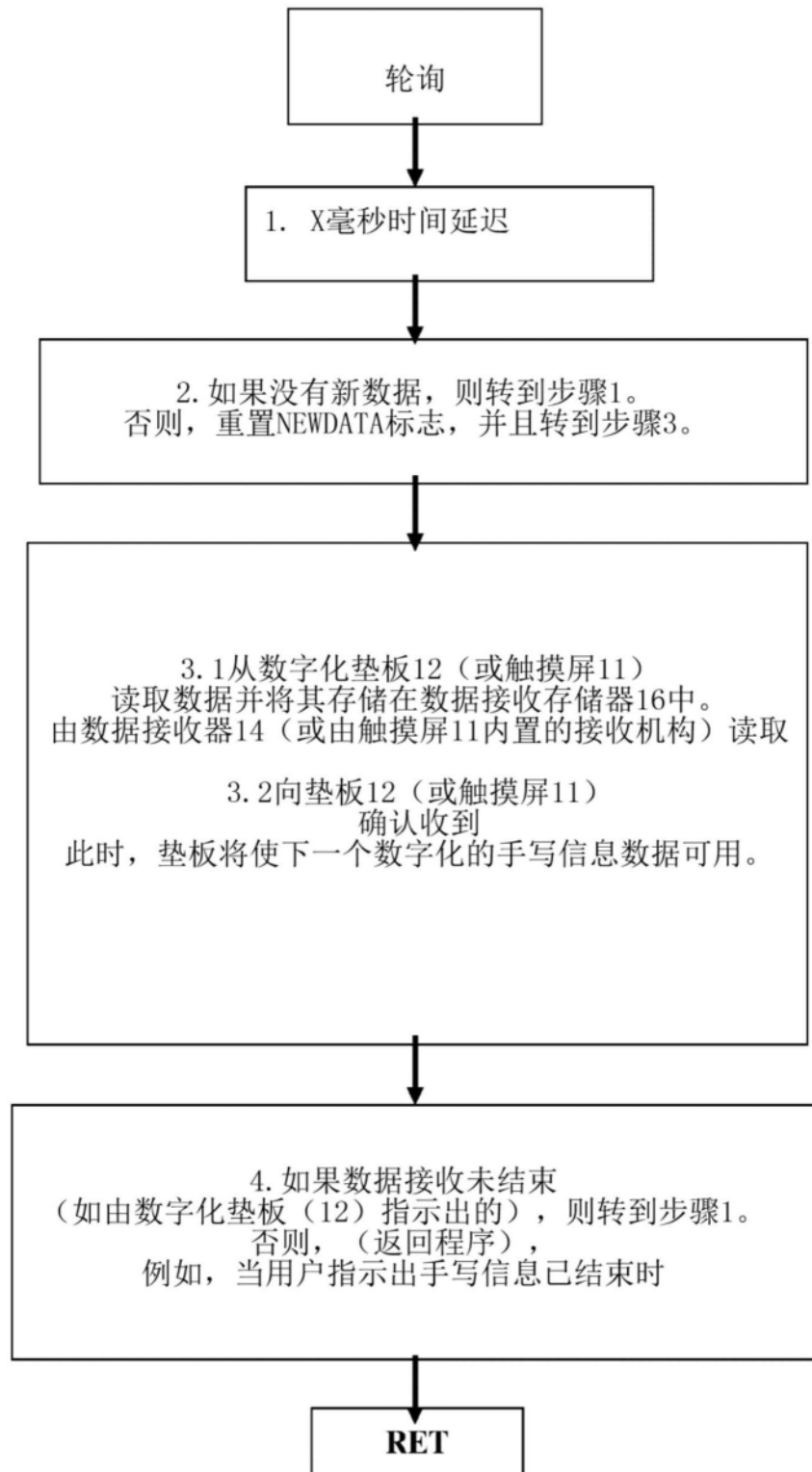


图3

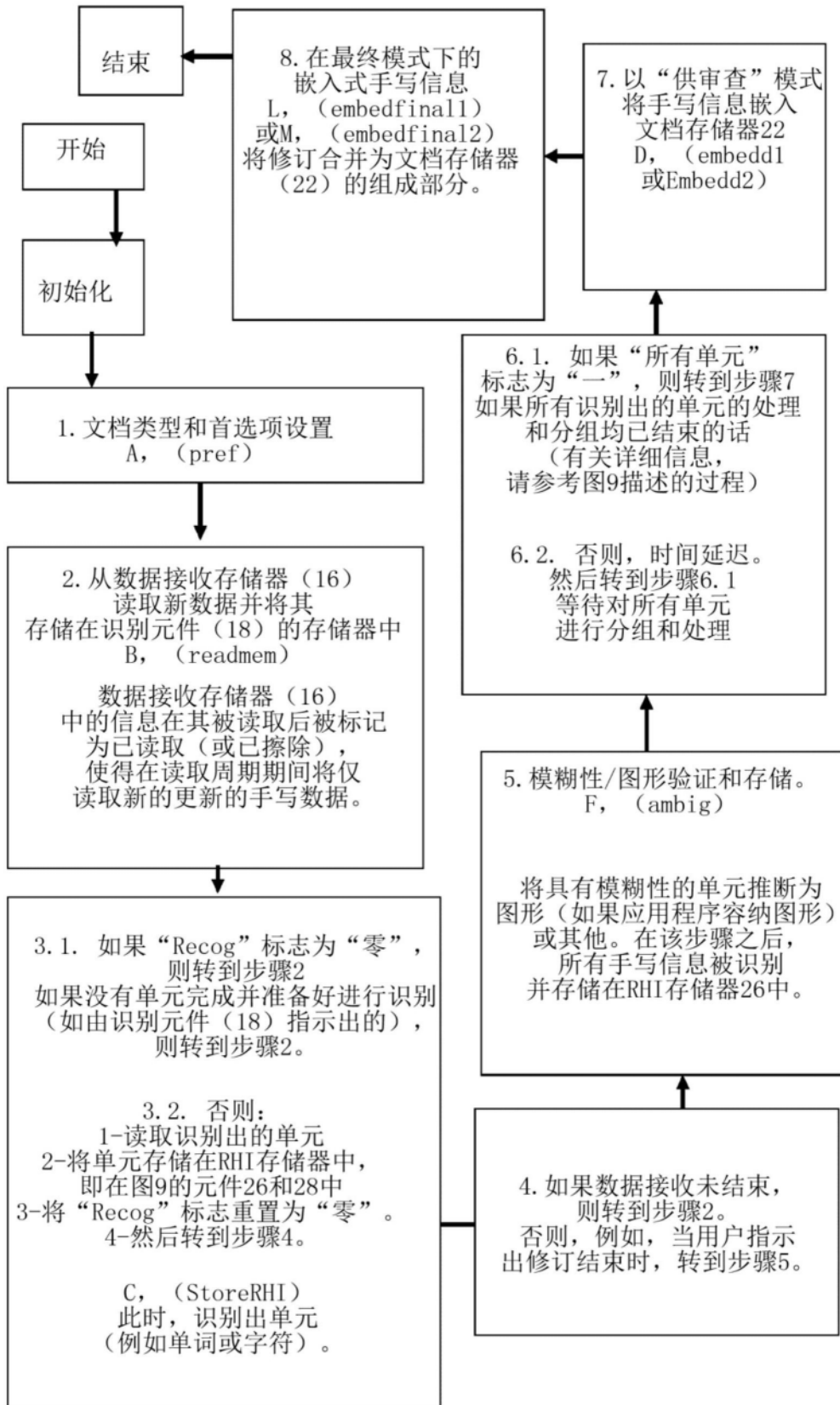


图4



图5

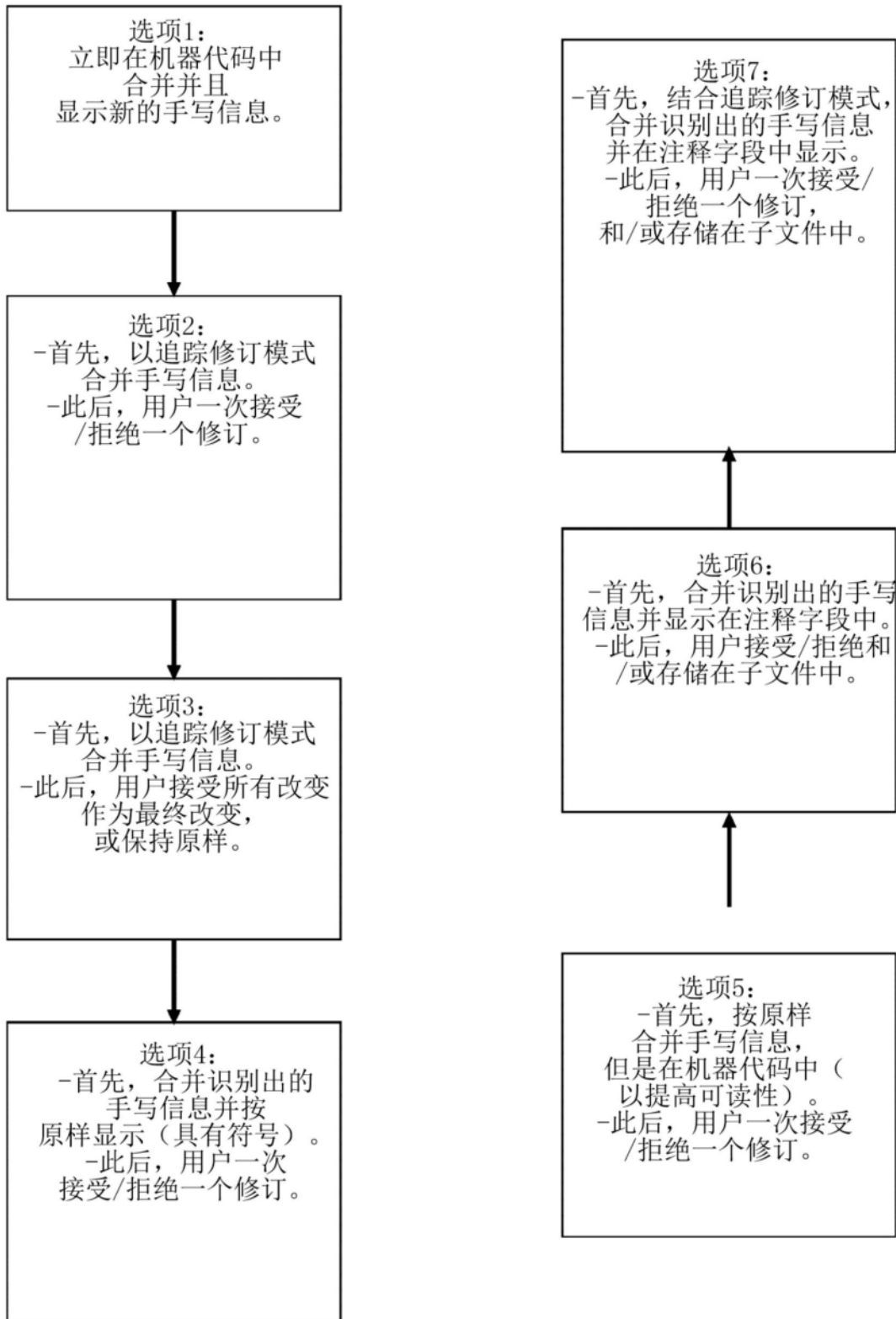


图6

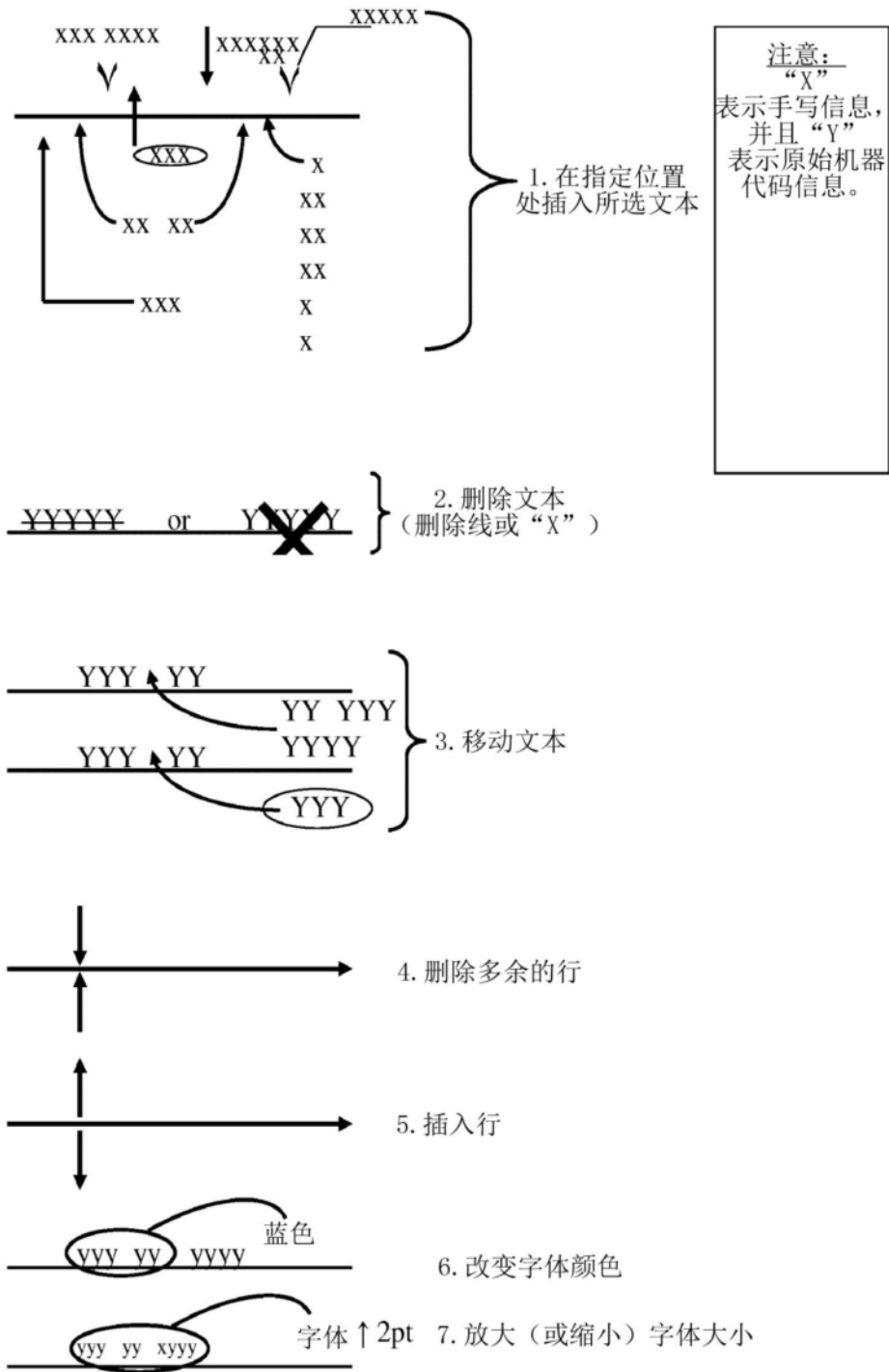


图7

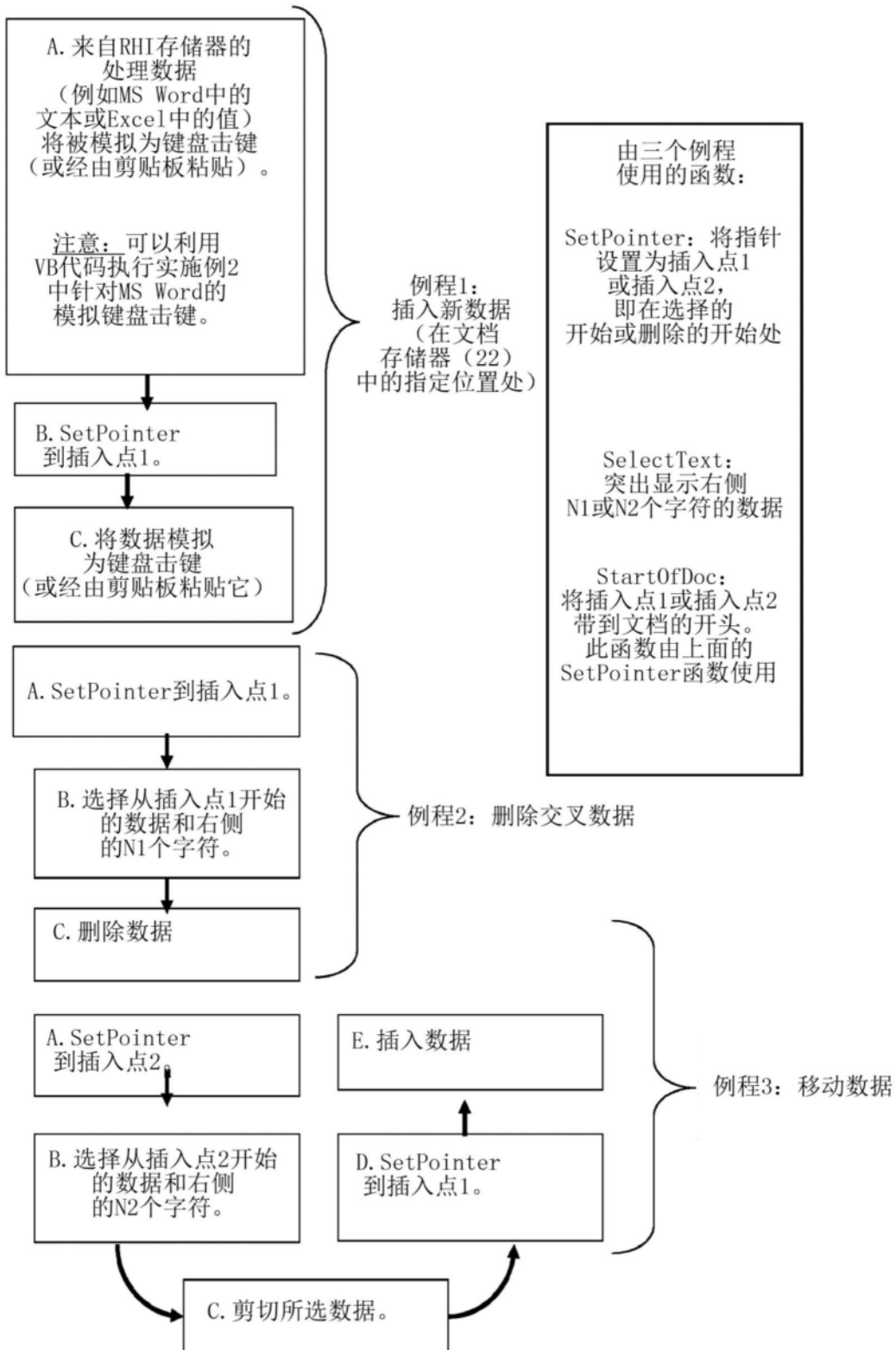


图8

1. 该标志由识别元件 (18) 设置为“一”，并在图4和图5中示出的子例程的步骤3.2中重置为“零”。
2. 该标志由识别元件 (18) 读取并且在图4和图5中示出的子例程的步骤3.1中读取。

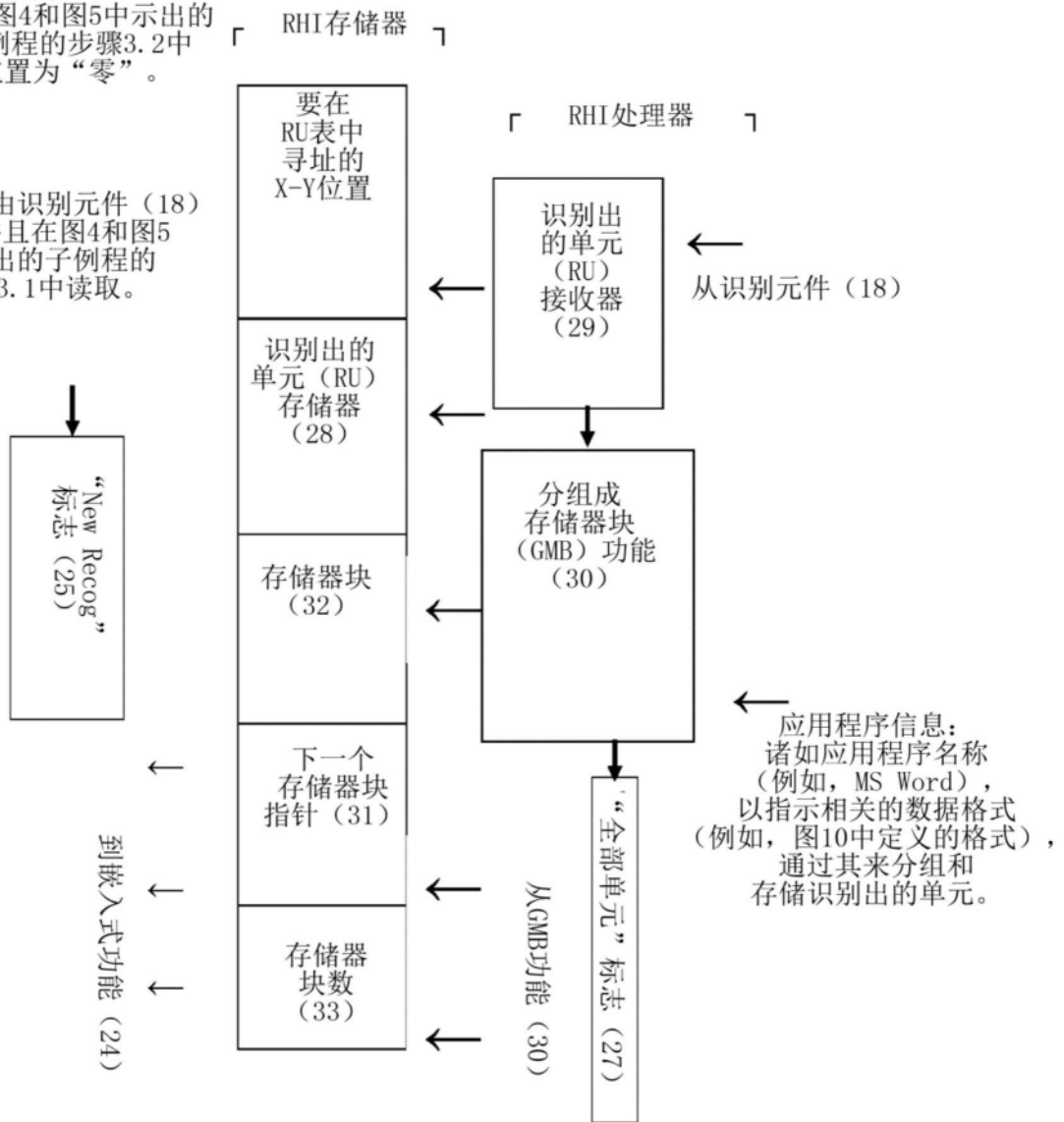


图9

命令	<p>示例： 1-插入文本，2-删除文本，3-移动文本， 4-删除多余的行，5-插入行，6-移动文本， 7-改变字体大小，8-改变字体颜色， 9-改变图像大小</p>	X-Y Location of InsertionPoint1 in the document memory 22
标志	<p>示例：文本/图像标志和标识标志</p>	
<p>x-y插入点1 例如，对于MS Word 3个参数： IP1-page# IP1-line# IP1-col#</p>	<p>指定位置1 x-y位置的定义取决于应用程序 例如，插入发生的位置或选择开始的位置 插入点1的页数 插入点1的页中的行数 插入点1的行中的列数</p>	X-Y Location of InsertionPoint2 in the document memory 22
N1	<p>从插入点1到右侧的字符长度， 例如，要选择的文本（并且然后复制或删除）</p>	
<p>x-y插入点2 例如，对于MS Word 3个参数： IP2-page# IP2-line# IP2-col#</p>	<p>指定位置2 x-y位置的定义取决于应用程序 例如，开始选择移动到选择点2的文本 插入点2的页数 插入点2的页中的行数 插入点2的行中的列数</p>	
N2	<p>从插入点2到右侧的字符长度， 用于文本选择和移动到插入点1</p>	
StartTextRHI	<p>识别出的文本在rhi存储器中的位置（要复制）</p>	
N3	<p>rhi存储器中以字符为单位的文本长度指定 要处理或要复制的字符</p>	
Text1Attributes	<p>基于用户首选项的修订属性， 例如字体和颜色修订</p>	
Text2Attributes	<p>在插入点2处插入的文本的属性， 例如字体和颜色修订</p>	
StartImageRHI	<p>图像在rhi存储器中的位置</p>	
AttributesImageRHI	<p>例如文件大小、帧信息、尺寸</p>	
AttributesImageDoc	<p>存储在文档存储器中之前要考虑的操作 例如文件格式（例如jpeg）、 文件大小/分辨率和帧信息</p>	

图10

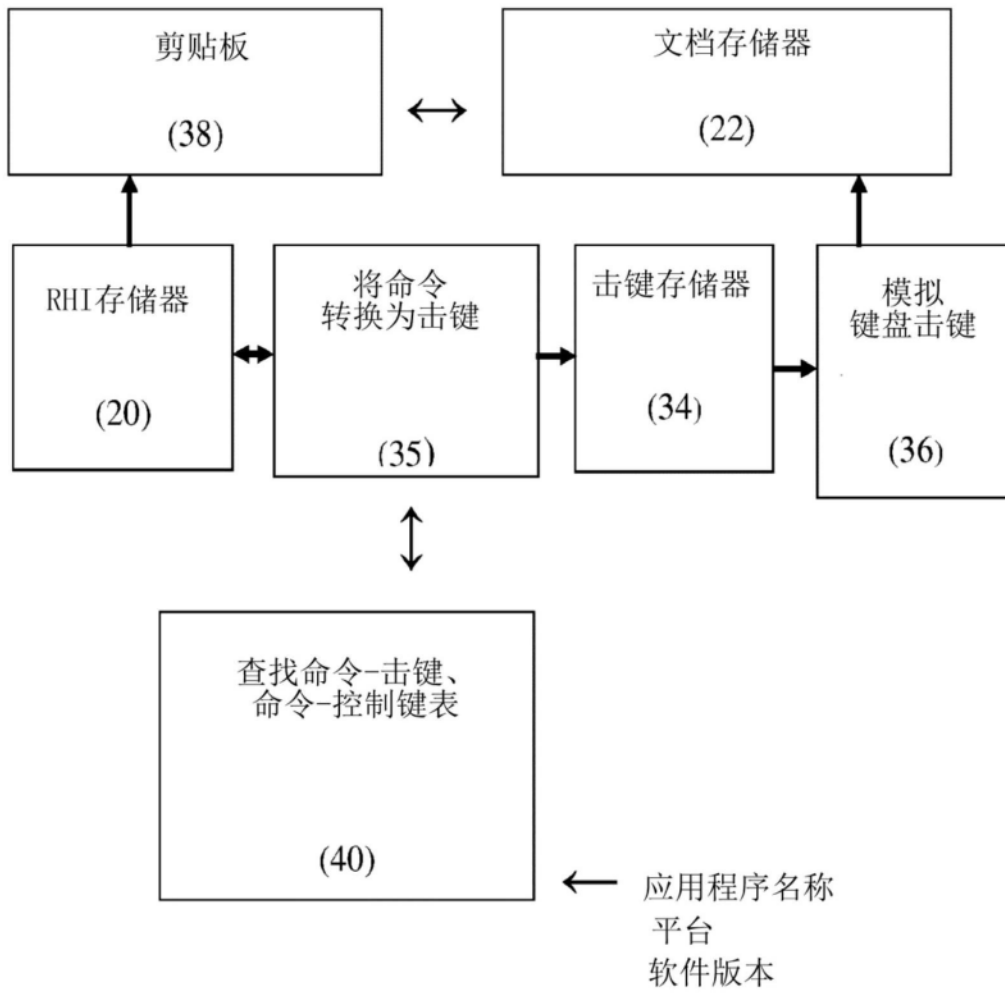


图11

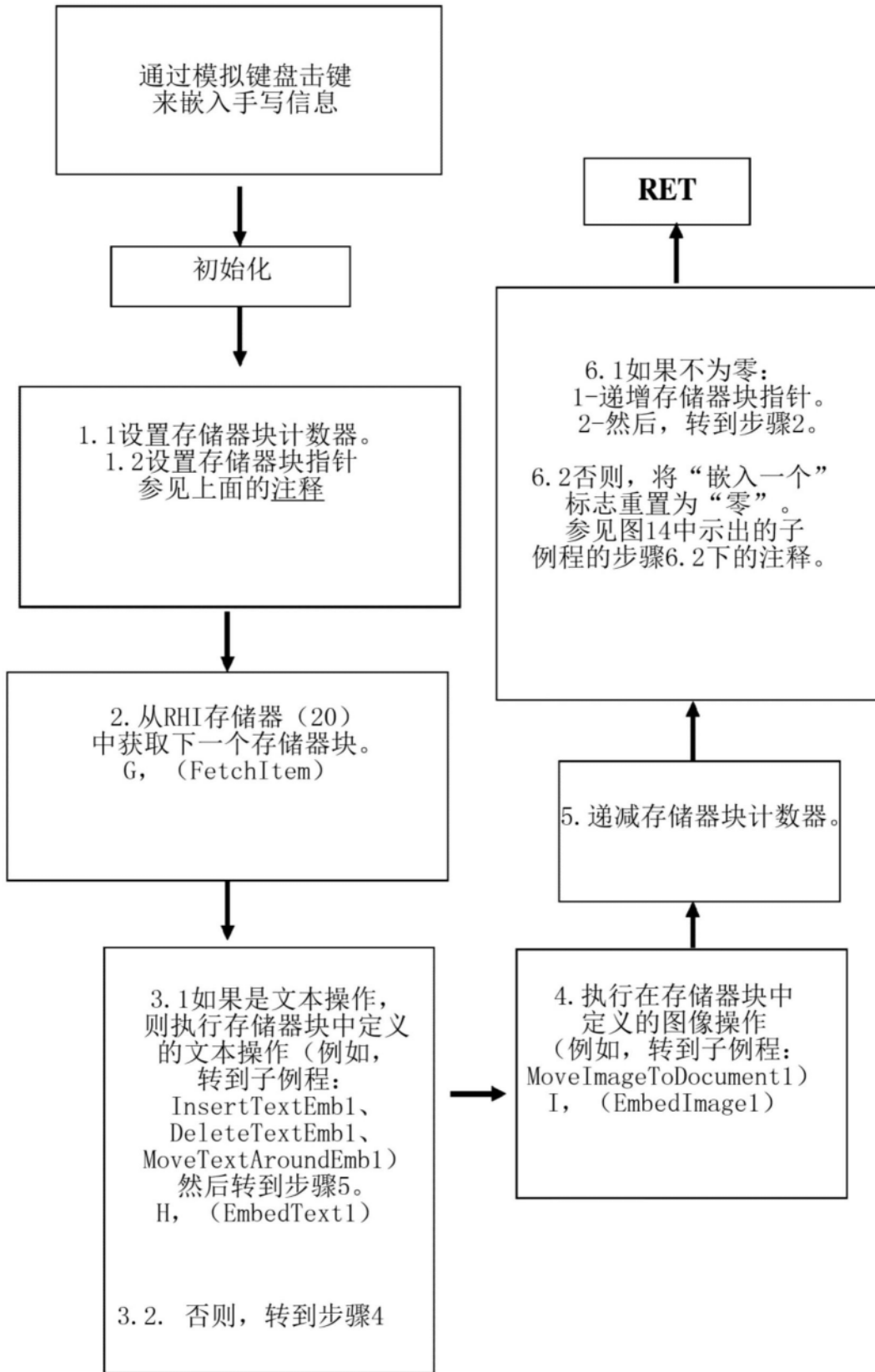


图12

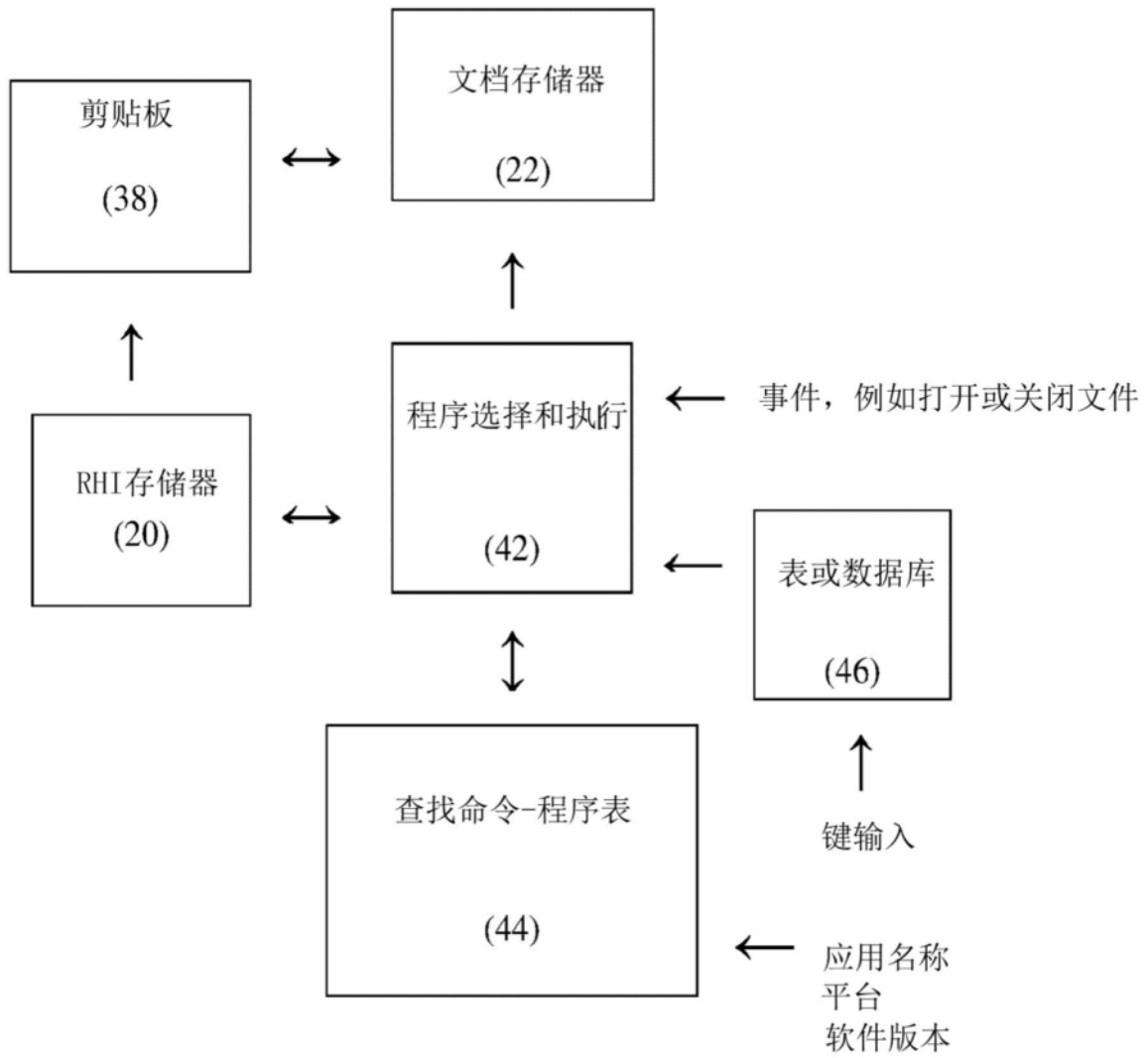


图13

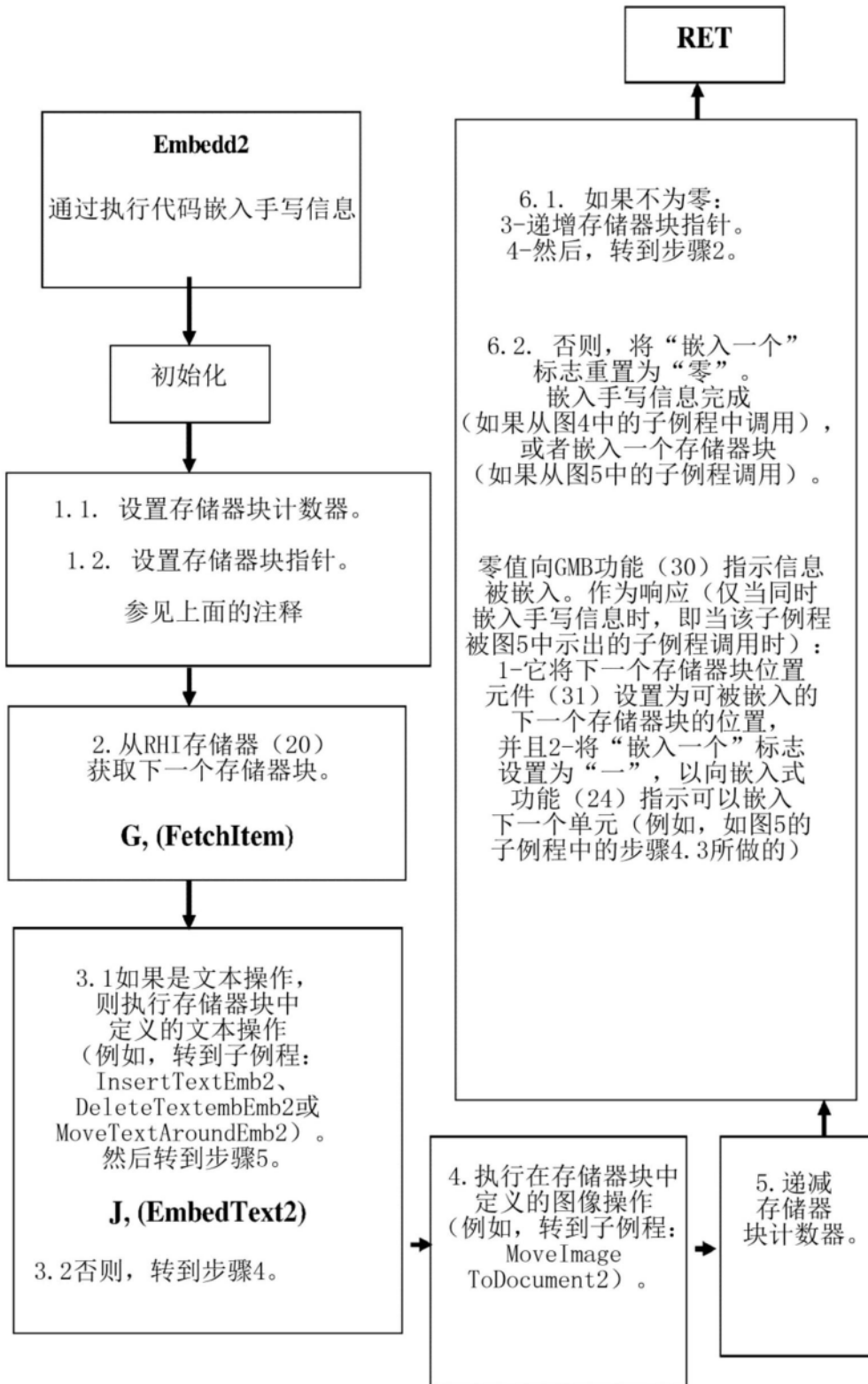


图14

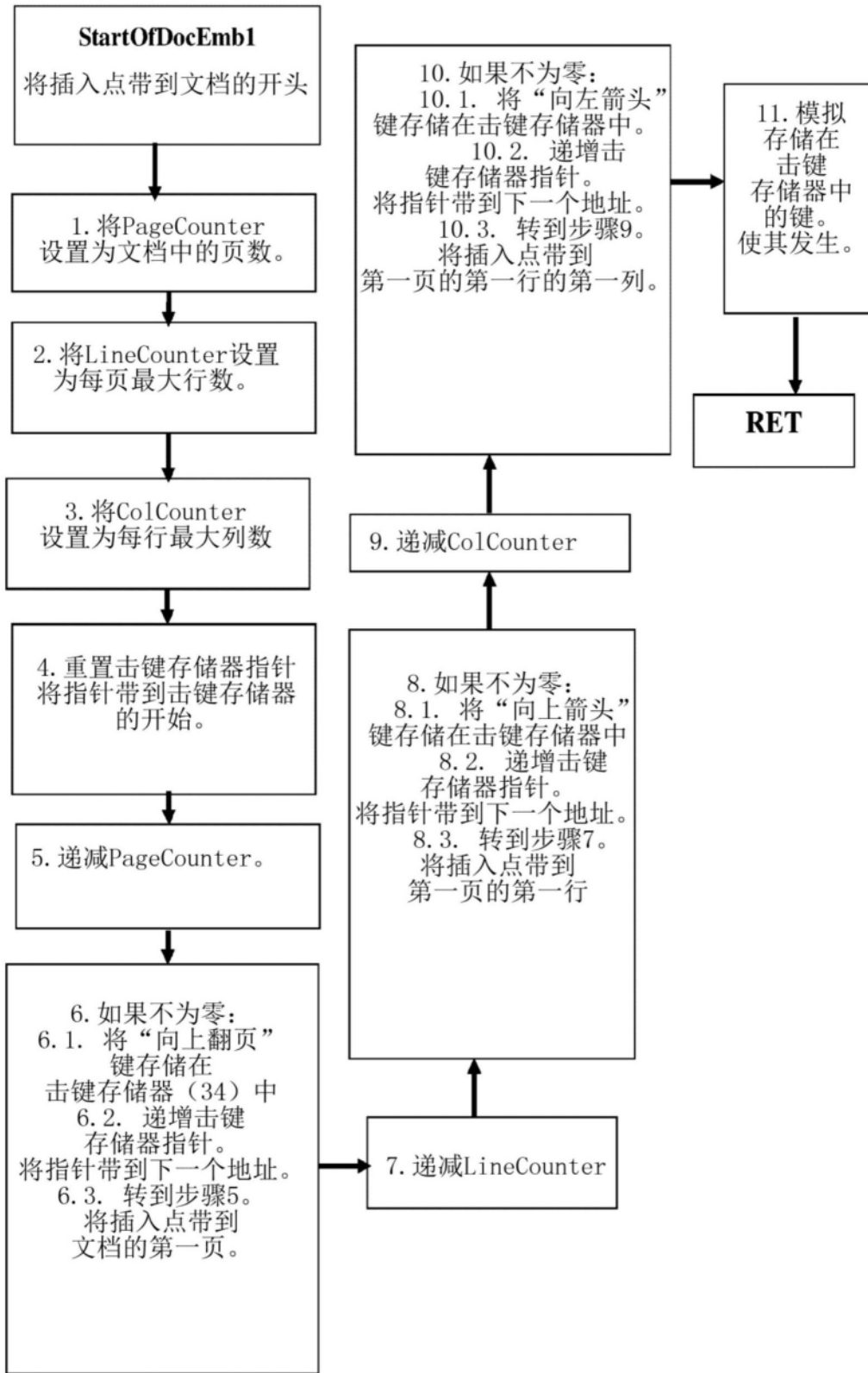


图15

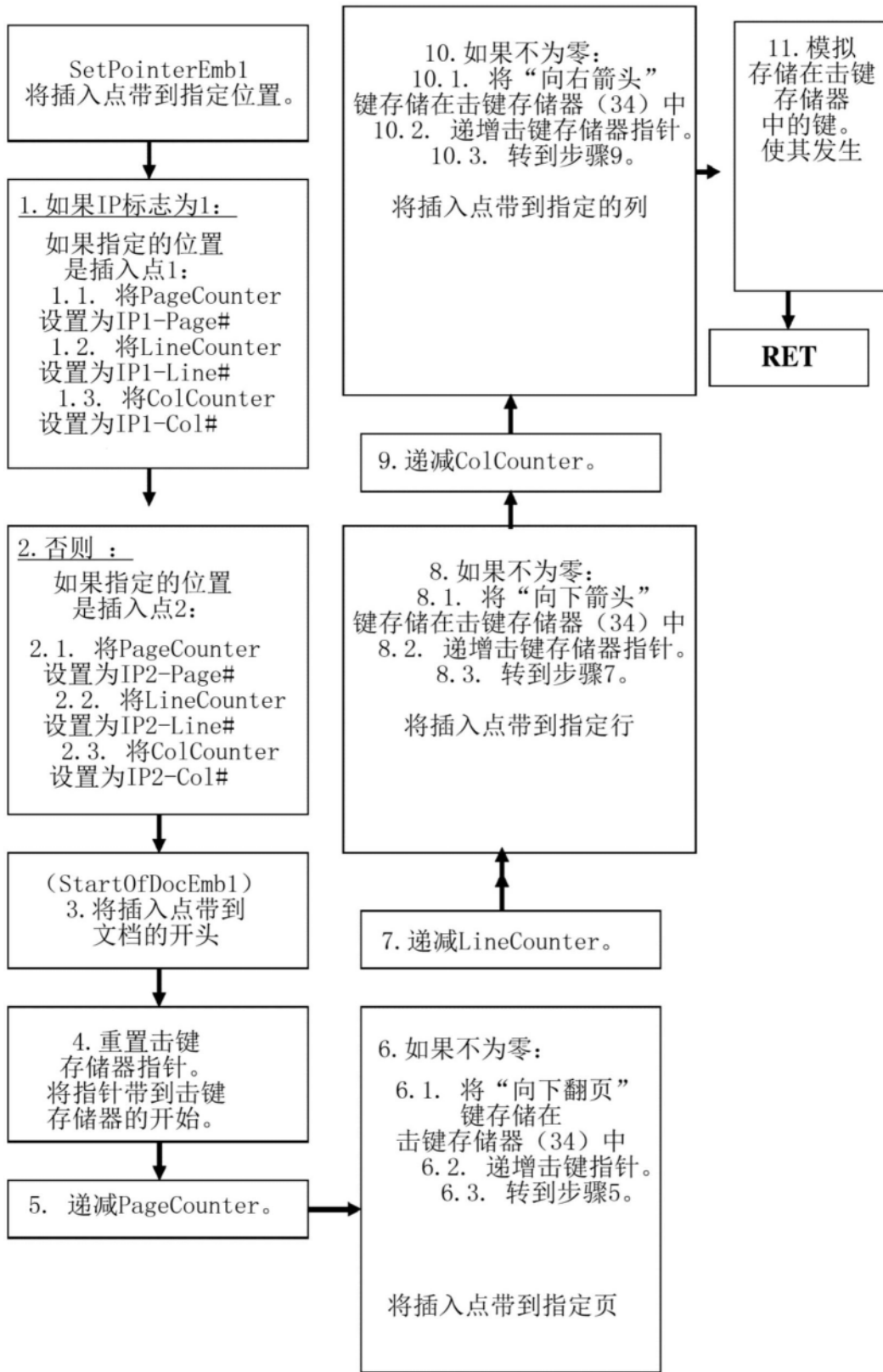


图16

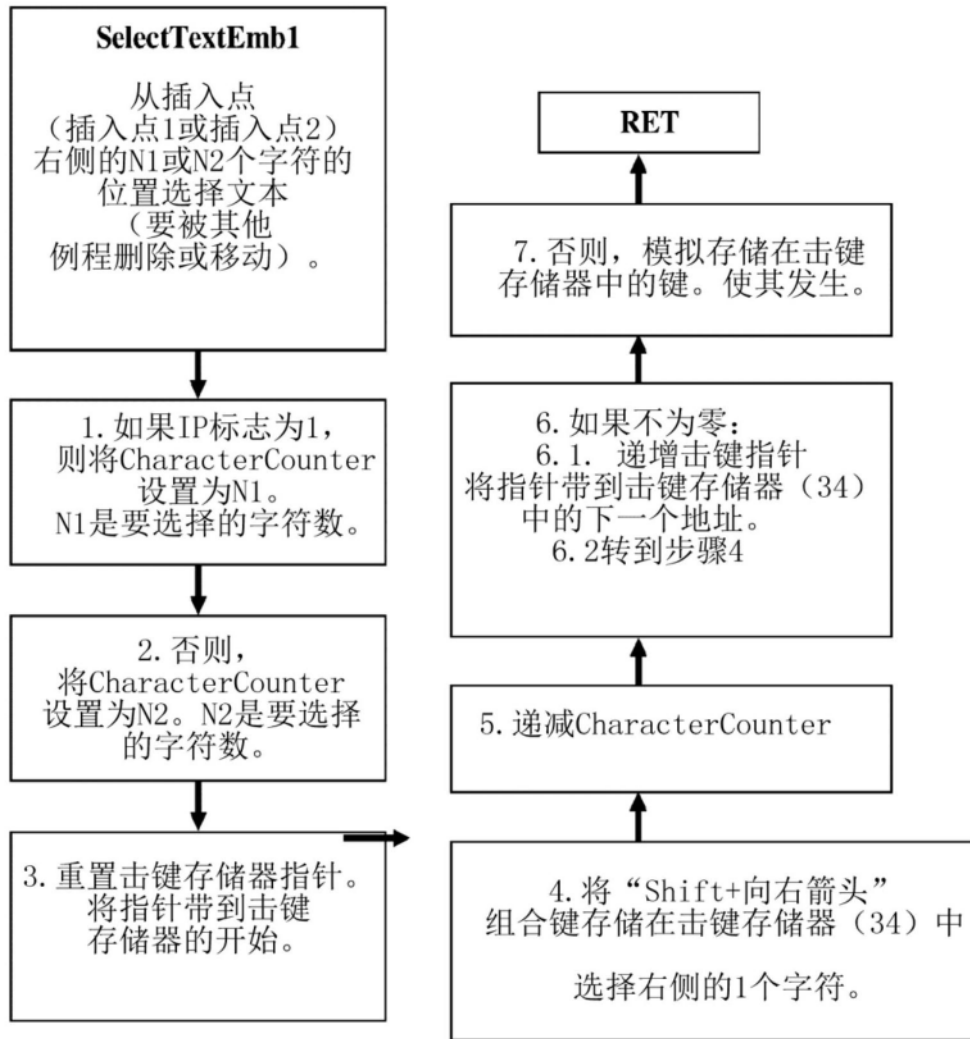


图17

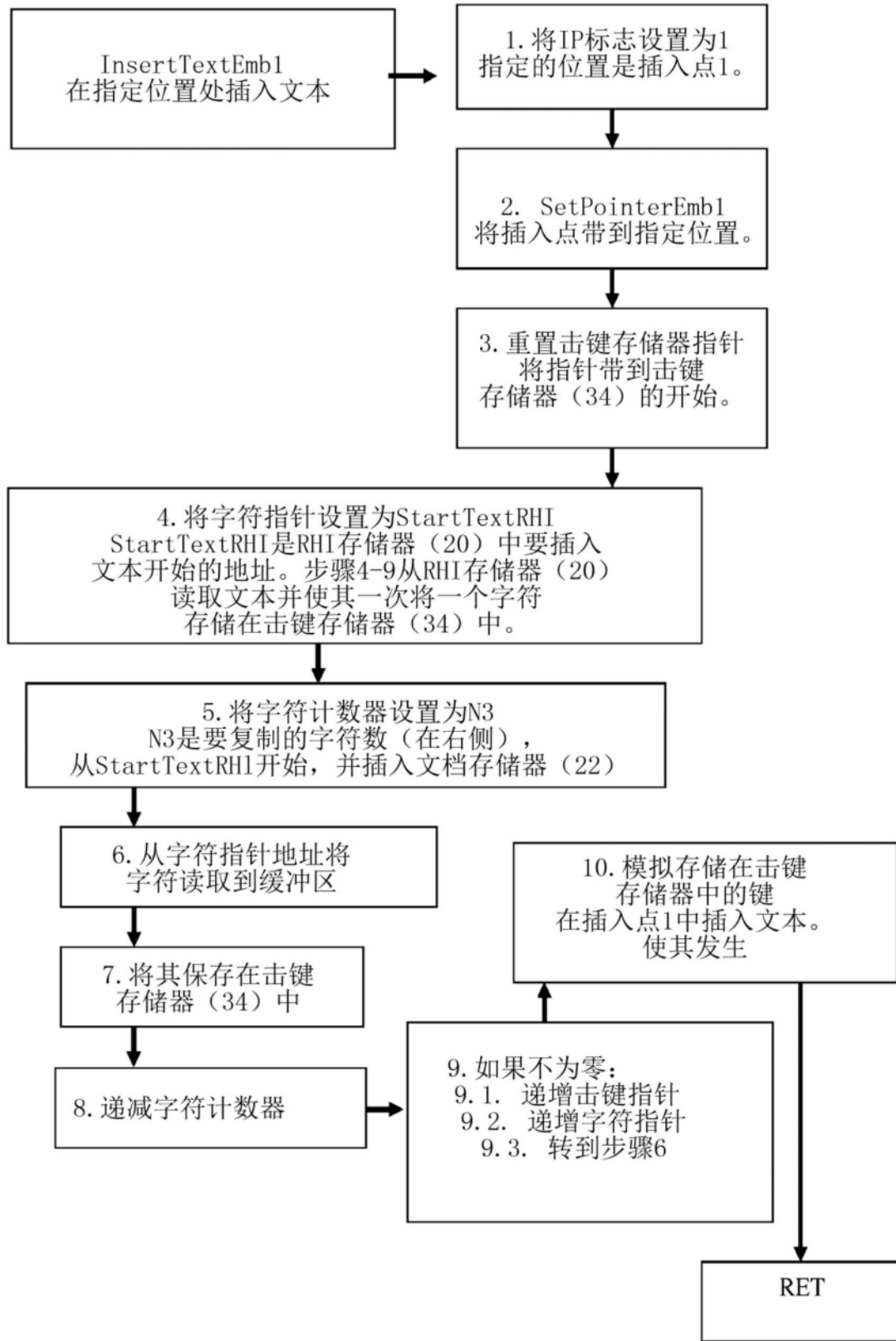


图18

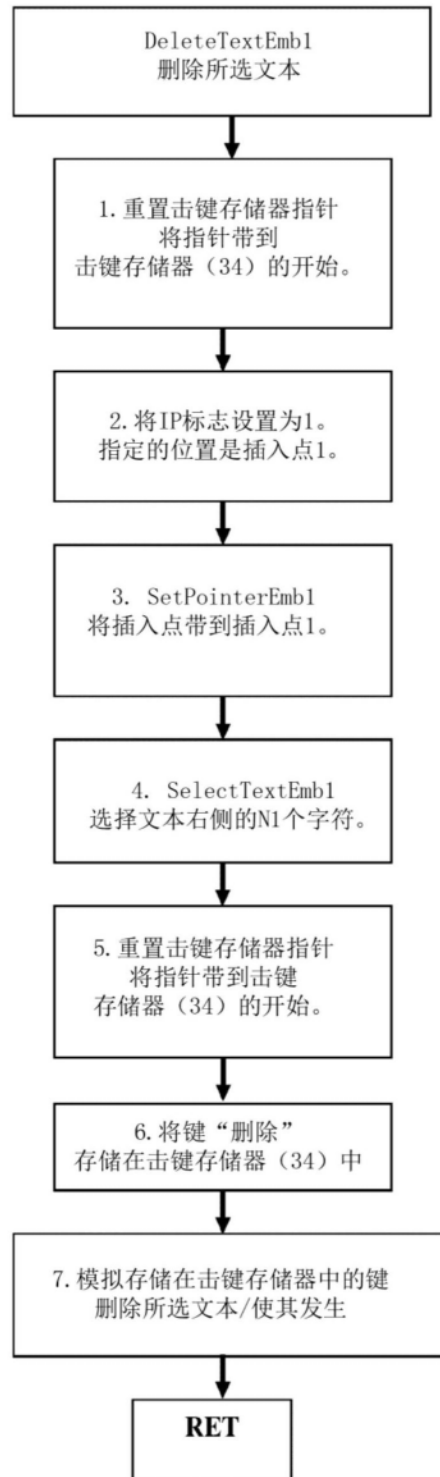


图19

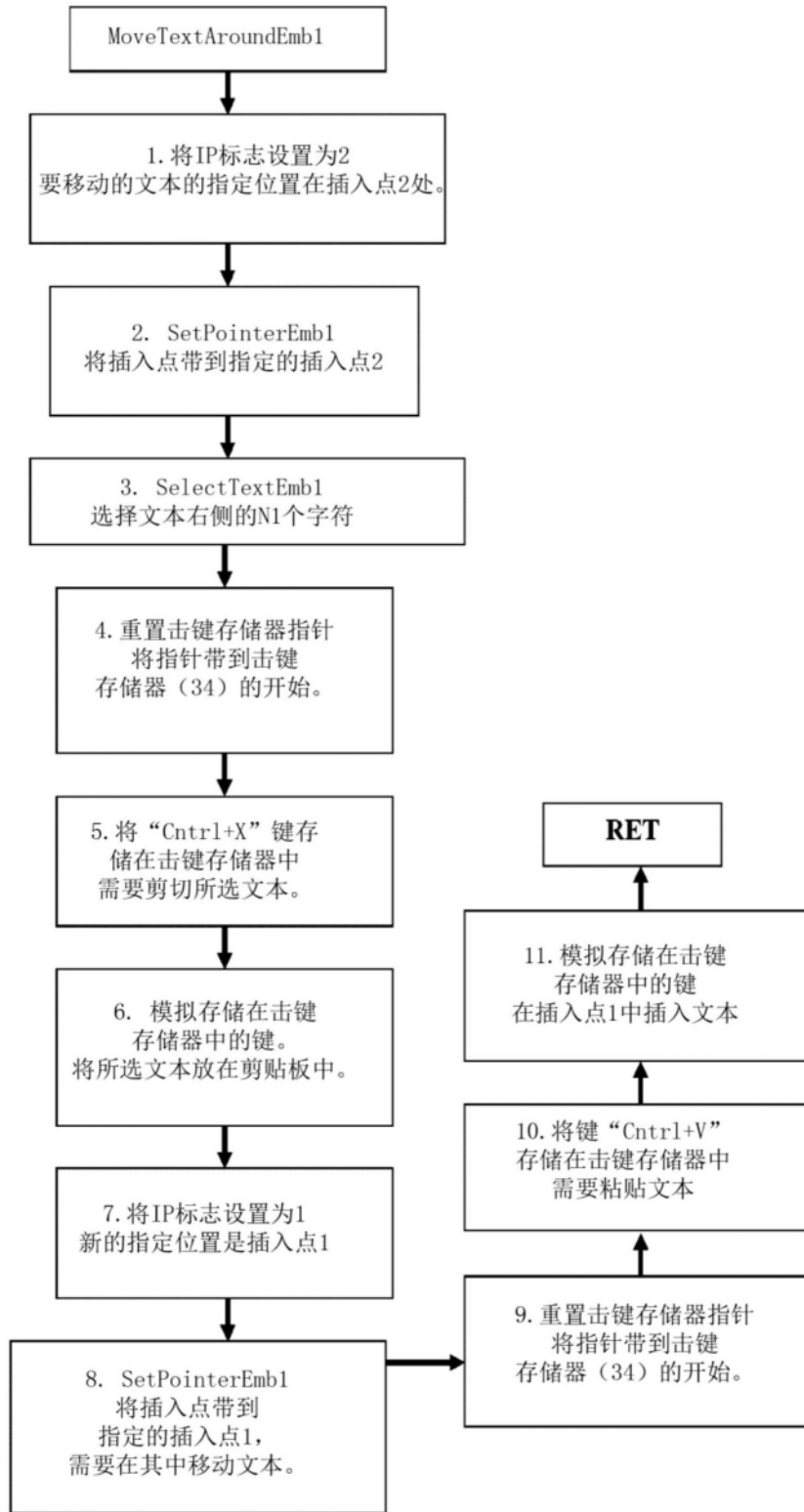


图20

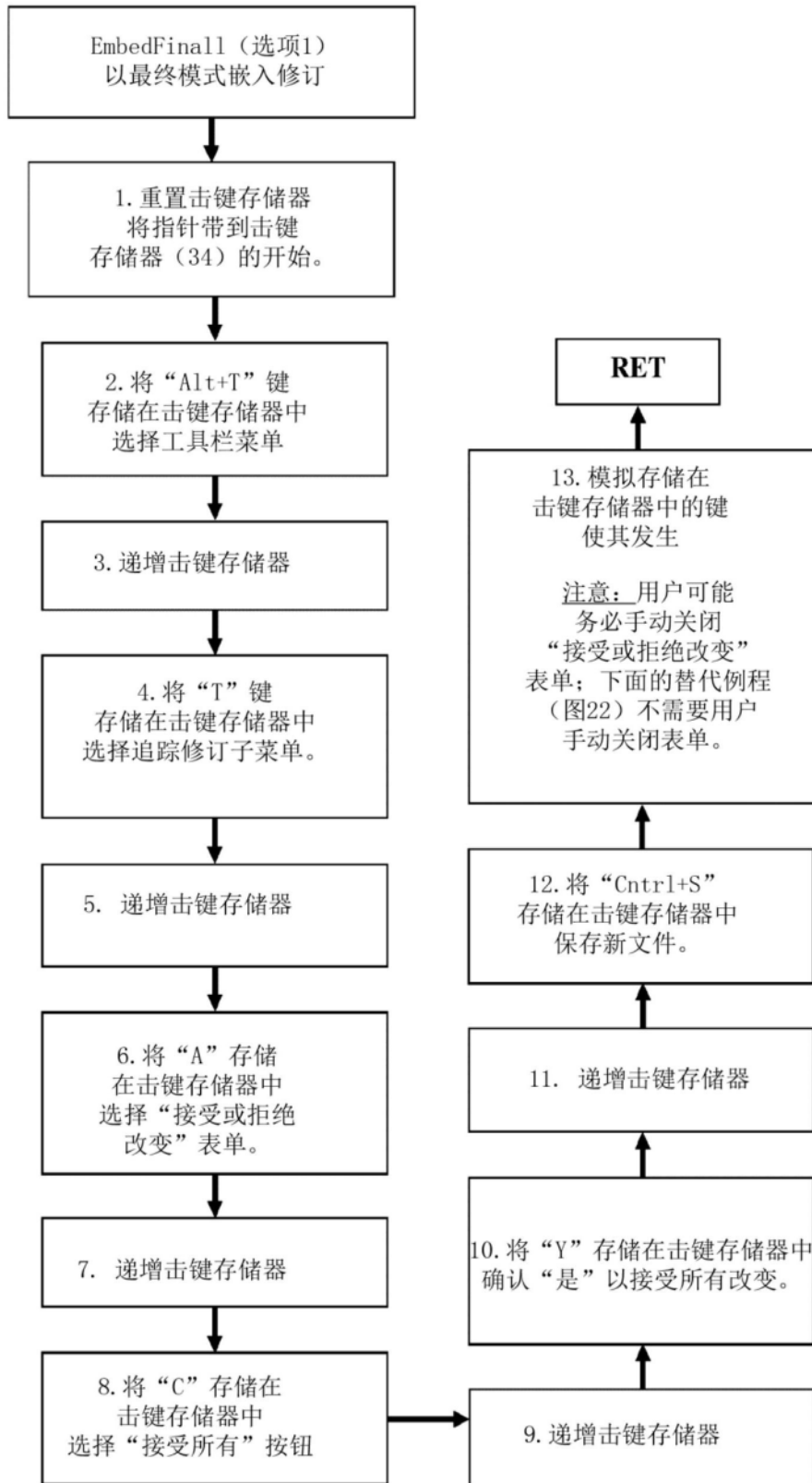


图21

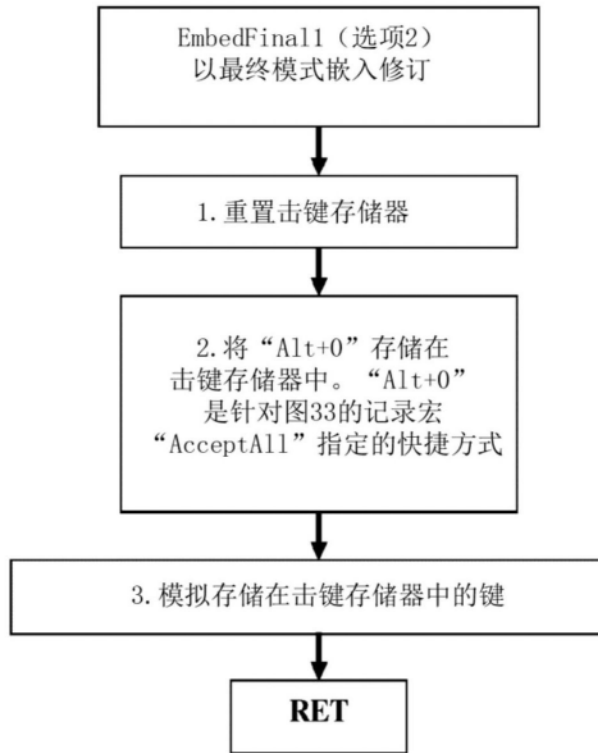


图22

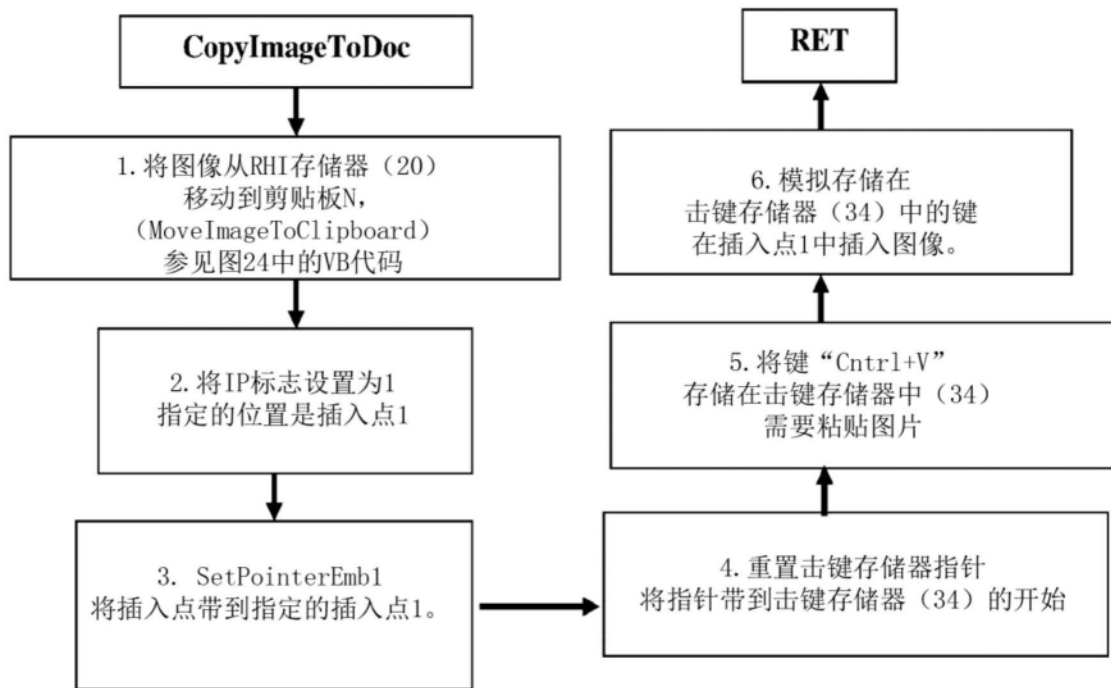


图23

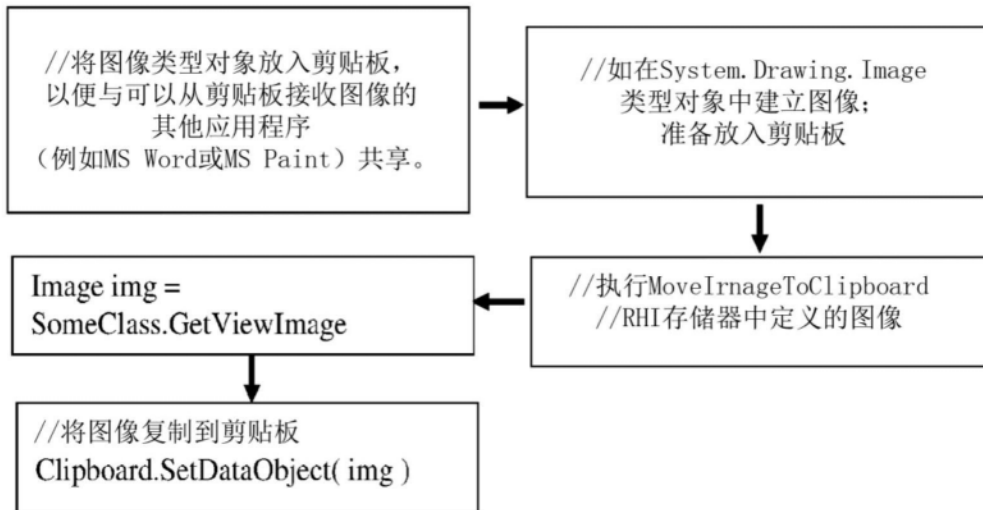


图24

内置宏	指定的快捷方式	宏到VB代码的转换
STARTOFDOCUMENT	CNTRL+HOME	SELECTION.HOMEKEY UNIT:=WDSTORY
GOTONEXTPAGE	用户选择	SELECTION.GOTONEXT WDGOTOPAGE
GOTONEXTLINE	向下	SELECTION.MOVEDOWN UNIT:=WDLIN, COUNT:=1
CHARRIGHT	向右	SELECTION.MOVERIGHT UNIT:=WDCHARACTER, COUNT:=1
CHARRIGHTEXTEND	SHIFT+向右	SELECTION.MOVERIGHT UNIT:=WDCHARACTER, COUNT:=1, EXTEND:=WDEXTEND
EDITCLEAR	DEL	SELECTION.DELETE UNIT:=WDCHARACTER, COUNT:=1
EDITCUT	CNTRL+X OR SHIFT+DEL	SELECTION.CUT
EDITPASTE	CNTRL+V OR SHIFT+插入	SELECTION.PASTE

图25

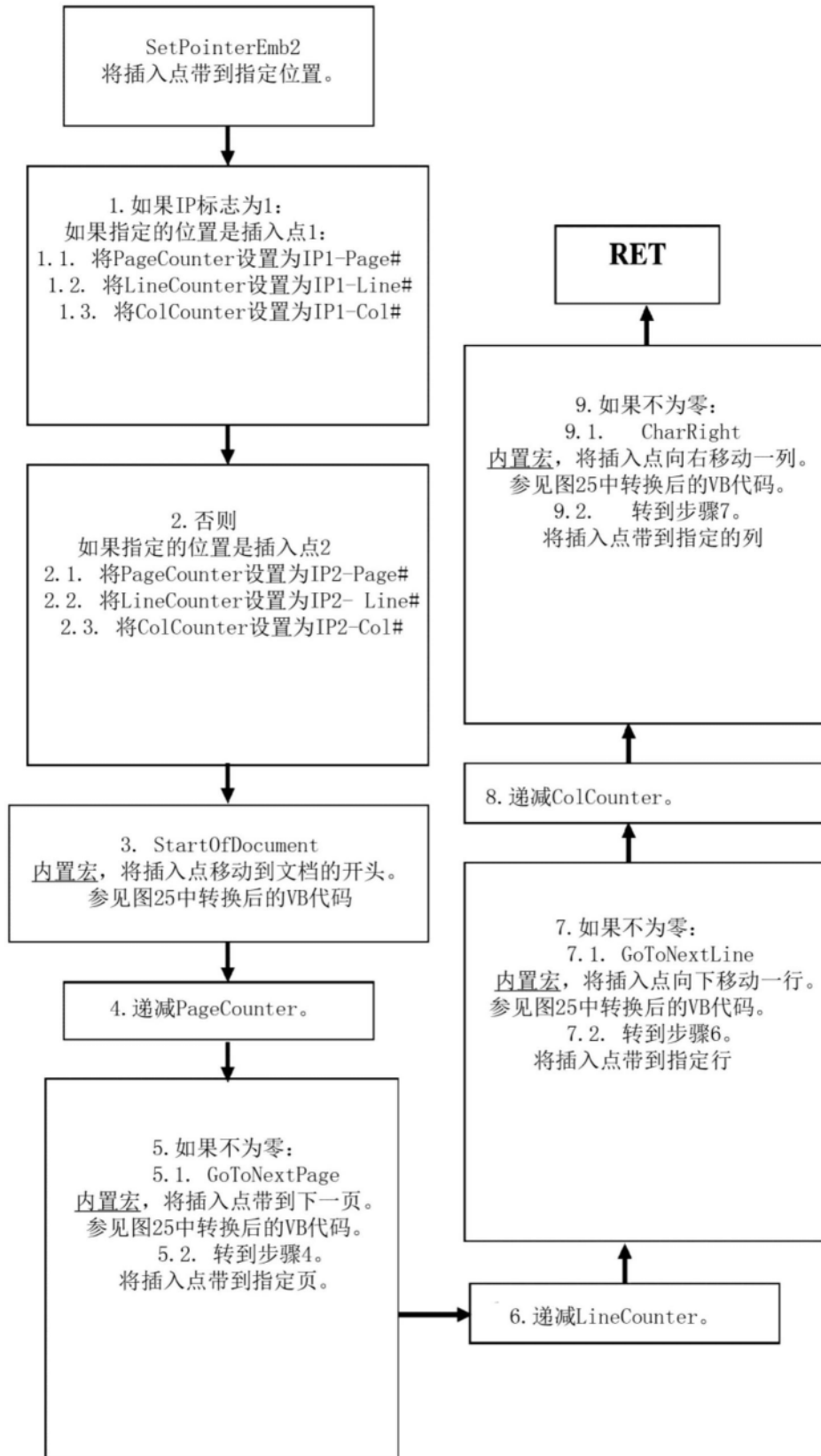


图26

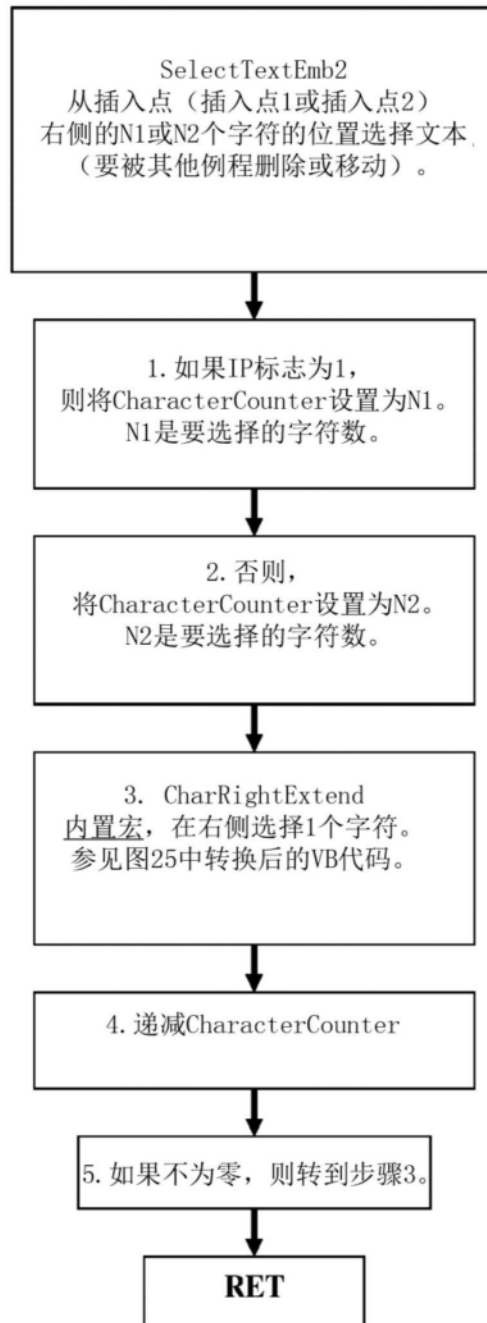


图27

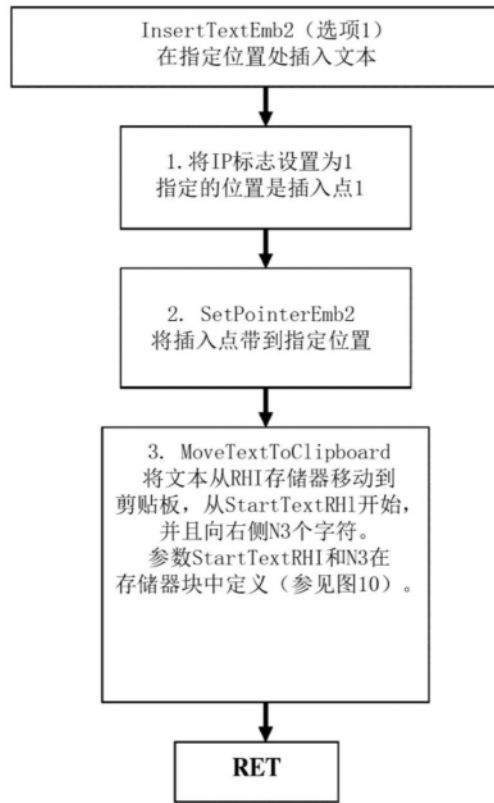


图28

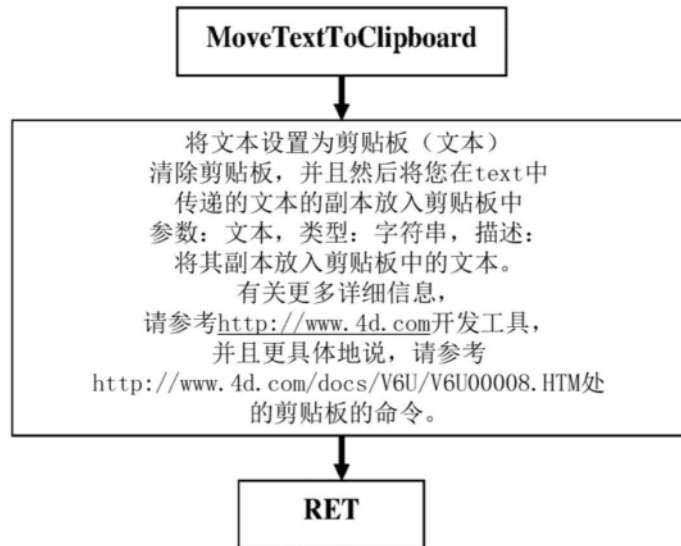


图29

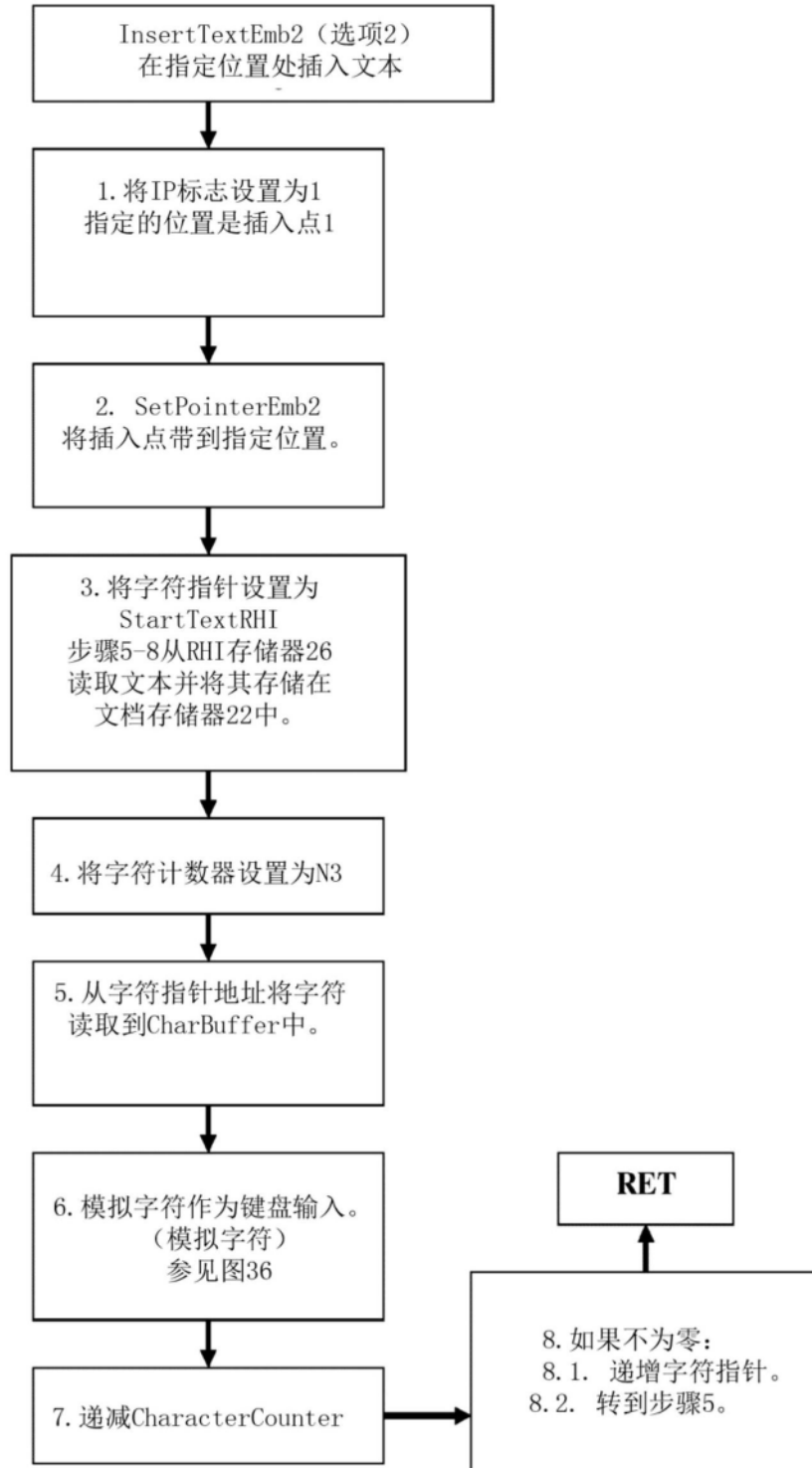


图30

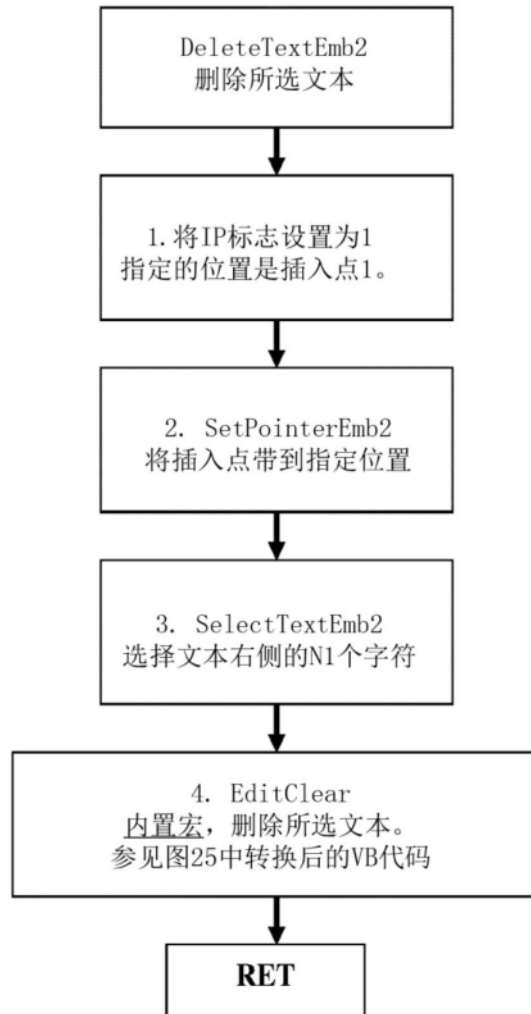


图31

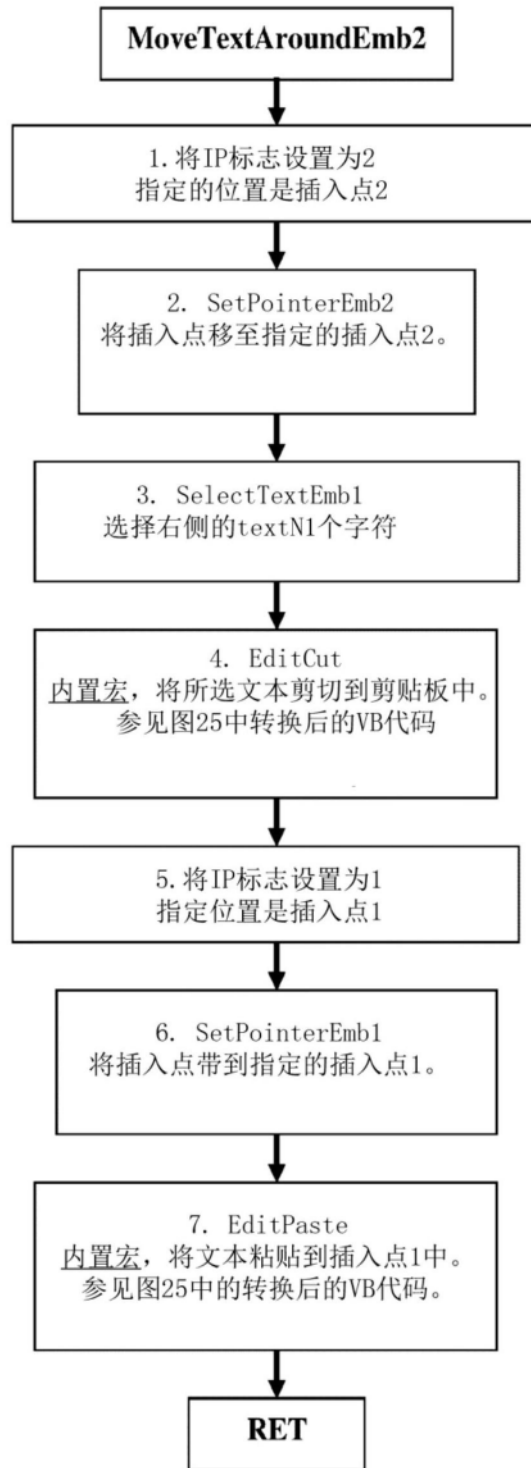


图32

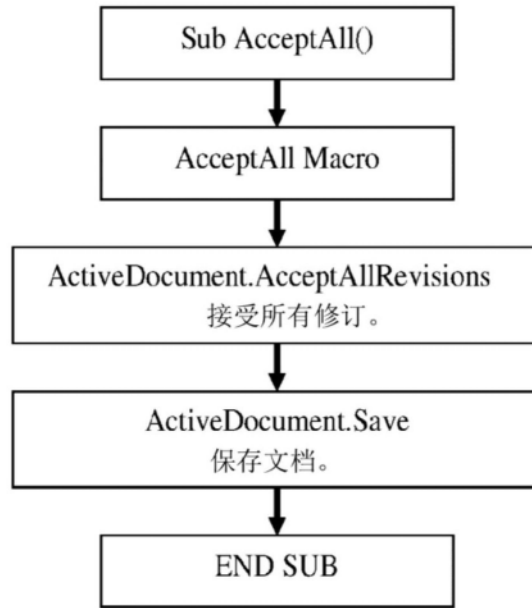


图33

内置宏	指定的快捷方式	宏到VB代码的转换
GOTOCOMMENTSCOPE	用户选择	SELECTION.COMMENTS(1).RANGE. SELECT
GOTONEXTCOMMENT	用户选择	SELECTION.GOTONEXT WDGOTOCOMMENT
GOTOPREVIOUSCOMMENT	用户选择	SELECTION.GOTOPREVIOUS WDGOTOCOMMENT
INSERTANNOTATION	ALT+CNRL+M	SELECTION.COMMENTS.ADD RANGE:=SELECTION.RANGE
DELETEANNOTATION	ALT+SHIFT+C	ACTIVIEWINDOW.ACTIVEPANE.CLOSE
VIEWANNOTATION	用户选择	ACTIVIEWINDOW.VIEW.SPLITSPECIAL = WDPANECOMMENTS

图34

键	MS WORD 的宏到VB代码的转换
A	SELECTION.TYPETEXT TEXT:="A"
A	SELECTION.TYPETEXT TEXT:="A"
B	SELECTION.TYPETEXT TEXT:="B"
C	SELECTION.TYPETEXT TEXT:="C"
C	SELECTION.TYPETEXT TEXT:="C"
D	SELECTION.TYPETEXT TEXT:="D"
E	SELECTION.TYPETEXT TEXT:="E"
F	SELECTION.TYPETEXT TEXT:="F"
G	SELECTION.TYPETEXT TEXT:="G"
H	SELECTION.TYPETEXT TEXT:="H"
I	SELECTION.TYPETEXT TEXT:="I"
J	SELECTION.TYPETEXT TEXT:="J"
K	SELECTION.TYPETEXT TEXT:="K"
L	SELECTION.TYPETEXT TEXT:="L"
M	SELECTION.TYPETEXT TEXT:="M"
0	SELECTION.TYPETEXT TEXT:="0"
1	SELECTION.TYPETEXT TEXT:="1"
2	SELECTION.TYPETEXT TEXT:="2"
3	SELECTION.TYPETEXT TEXT:="3"
4	SELECTION.TYPETEXT TEXT:="4"
5	SELECTION.TYPETEXT TEXT:="5"
6	SELECTION.TYPETEXT TEXT:="6"
@	SELECTION.TYPETEXT TEXT:="@"
↑	SELECTION.MOVEUP UNIT:=WDLIN, COUNT:=1
↓	SELECTION.MOVEDOWN UNIT:=WDLIN, COUNT:=1
→	SELECTION.MOVERIGHT UNIT:=WDCHARACTER, COUNT:=1
←	SELECTION.MOVELEFT UNIT:=WDCHARACTER, COUNT:=1
PAGE UP	SELECTION.MOVEUP UNIT:=WDScreen, COUNT:=1
PAGE DOWN	SELECTION.MOVEDOWN UNIT:=WDScreen, COUNT:=1
HOME	SELECTION.HOMEKEY UNIT:=WDLIN
END	SELECTION.ENDKEY UNIT:=WDLIN
TAB	SELECTION.TYPETEXT TEXT:=VBTAB
TAB W/I A TABLE	SELECTION.MOVERIGHT UNIT:=WDCELL
ESQ	SELECTION.ESCAPEKEY
SHIFT	MACRO MAY ONLY BE RECORDED IN COMBINATION WITH OTHER KEYS
CAPS LOCK	MACRO FOR THIS KEY MAY NOT BE RECORDED

图35

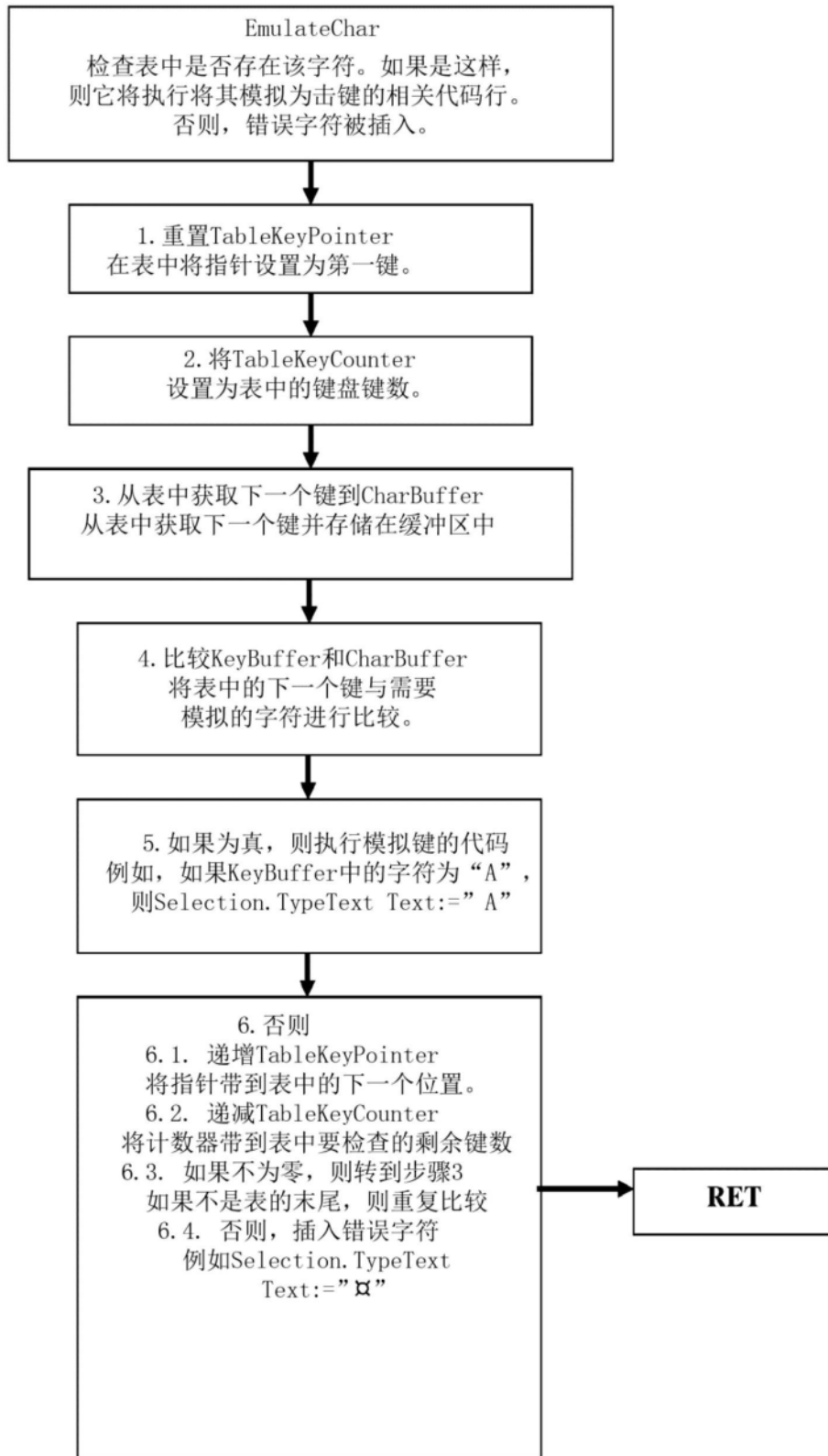


图36

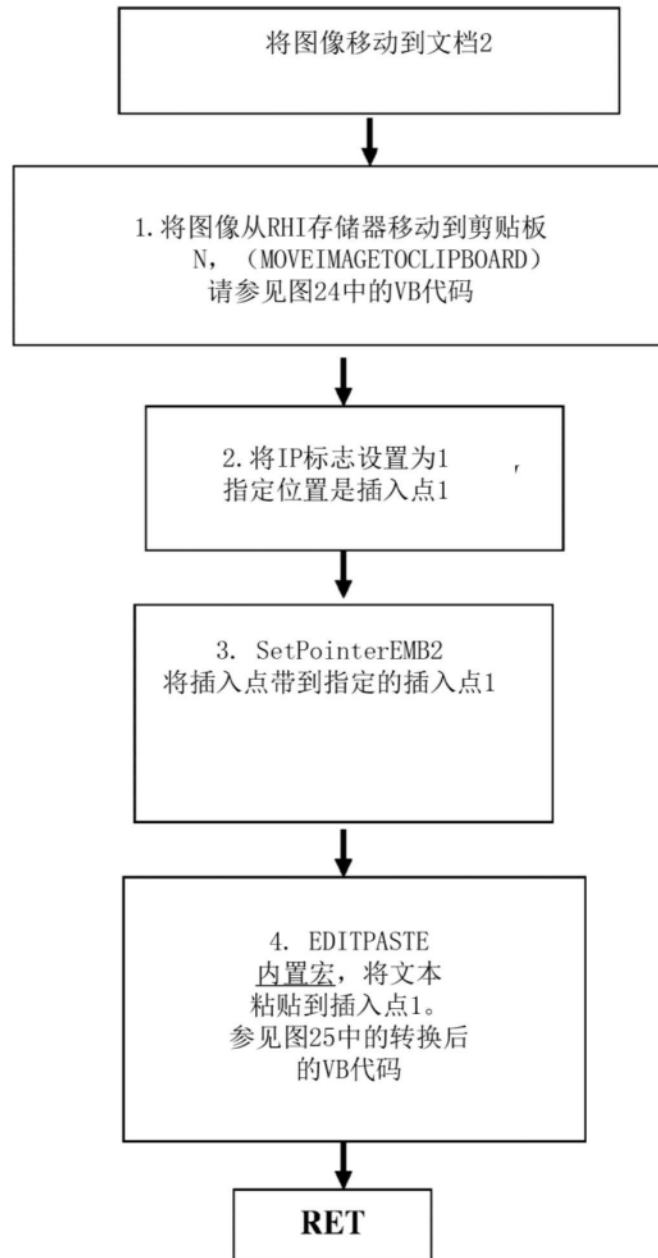


图37

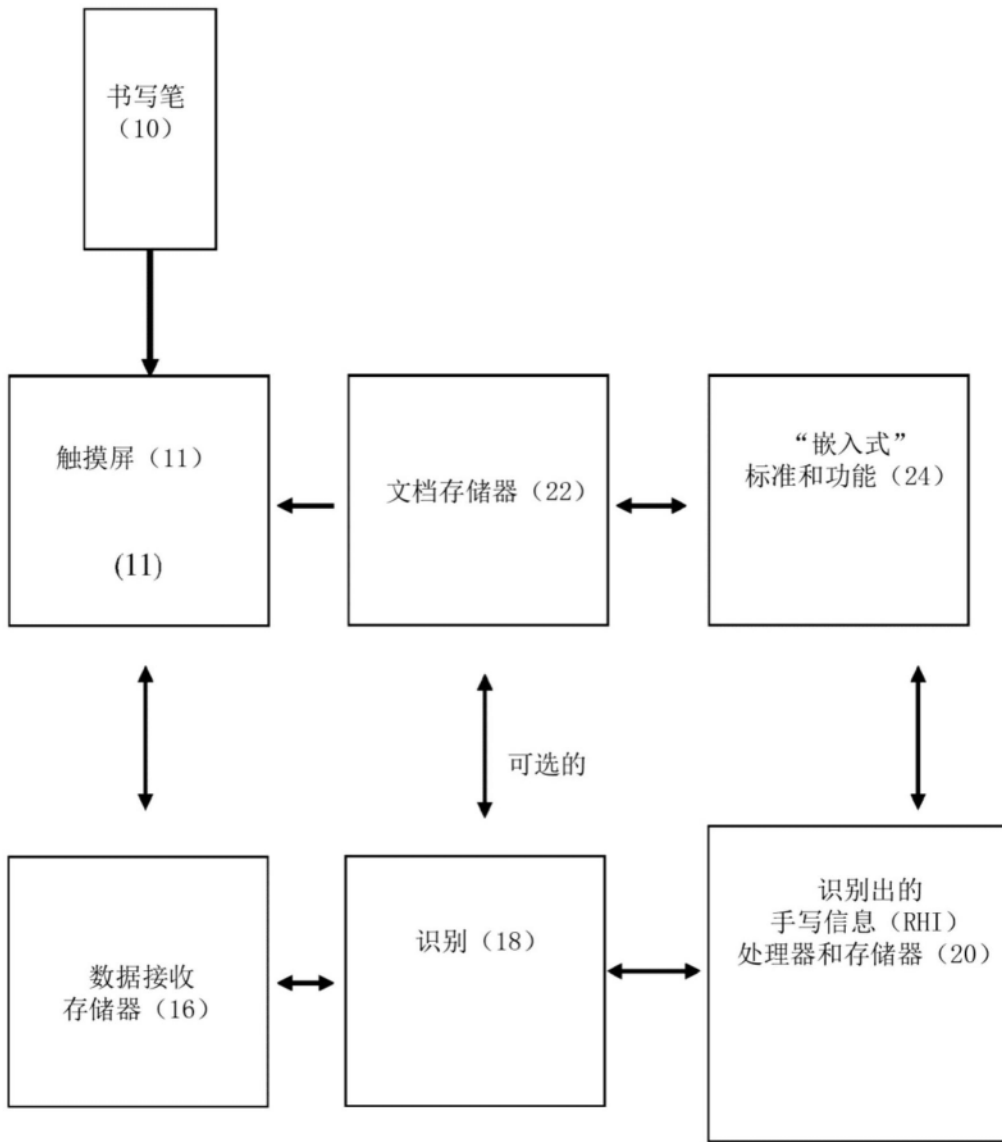


图38

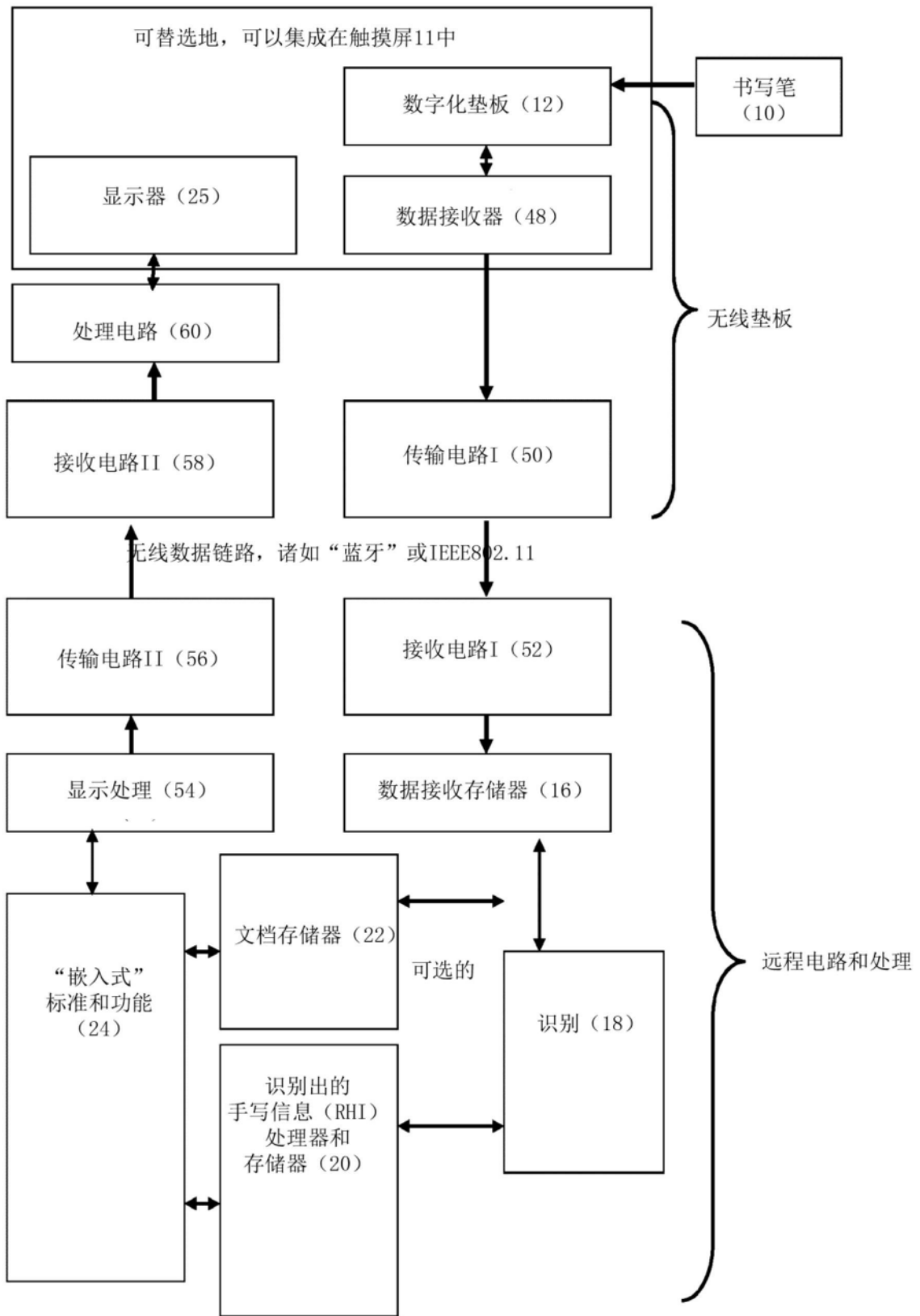


图39

“徒手”在“A”和“B”之间绘制线

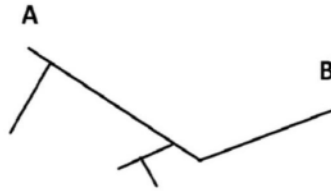
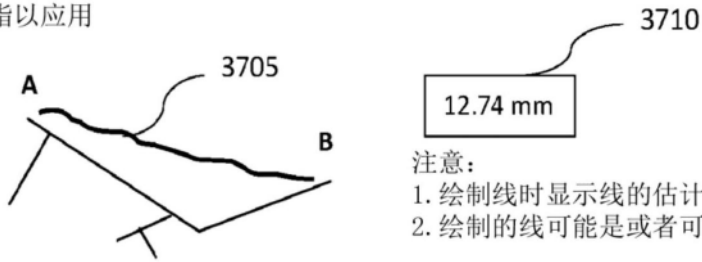


图40A

抬起手指以应用

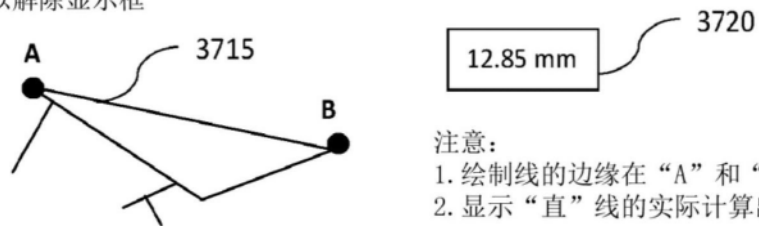


注意:

- 1. 绘制线时显示线的估计距离。
- 2. 绘制的线可能是或者可能不是直线，并且其每个

图40B

手指被抬起
触摸屏幕以解除显示框



注意:

- 1. 绘制线的边缘在“A”和“B”处自动捕捉
- 2. 显示“直”线的实际计算出的距离

图40C

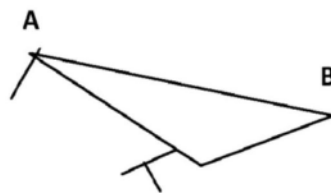


图40D

触摸对象以选择它

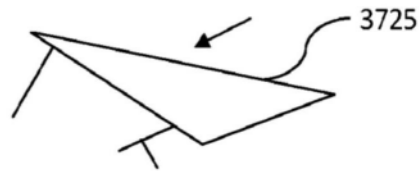


图41A

在触摸屏上的任意位置绘制“d”以删除它

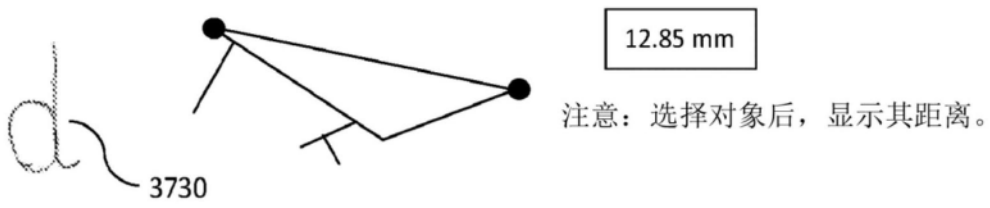


图41B



图41C

触摸线以选择它（可选步骤）



图42A

在触摸屏上的任意位置绘制“L”，并且然后将所期边缘向上或向下移动到所期位置

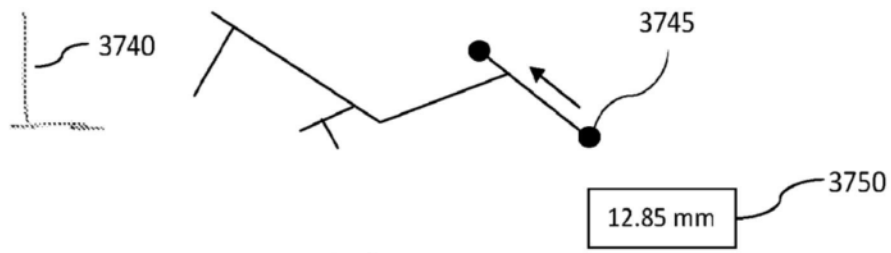


图42B

抬起手指以应用

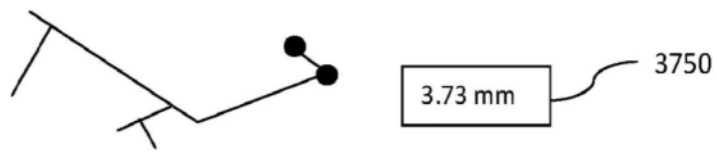


图42C

手指被抬起

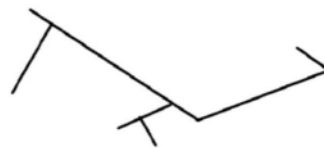


图42D

触摸线以选择它（可选步骤）

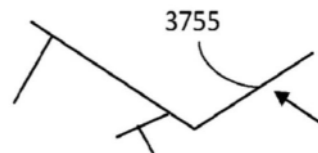


图43A

在触摸屏上的任意位置绘制“a”，并且然后向上或向下旋转所期边缘以减小或增大目标角度。

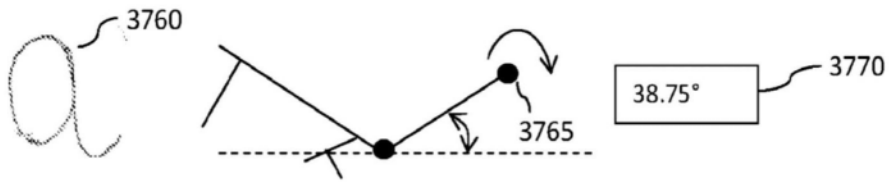


图43B

抬起手指以应用

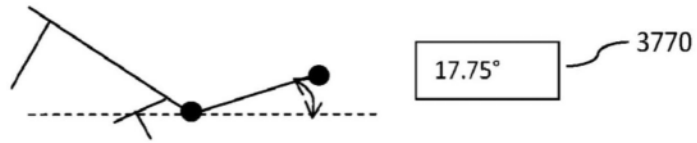


图43C

手指被抬起

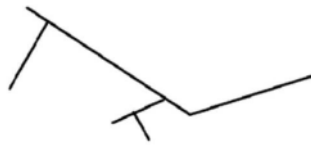


图43D

触摸线以选择它

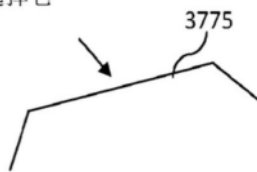


图44A

绘制“R”，并且然后跨上/下移动以应用或改变半径

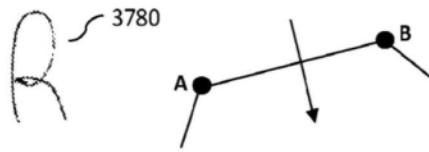


图44B

抬起手指以应用

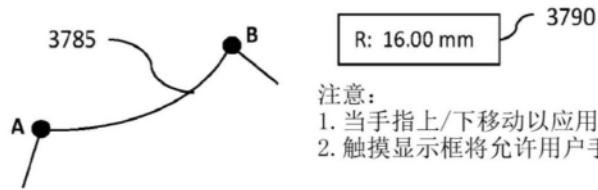


图44C

手指被抬起



图44D

绘制“N”，并且然后触摸参考线

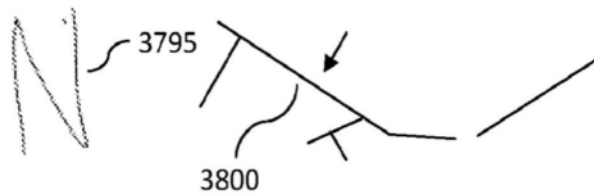


图45A

触摸目标线，并且抬起手指

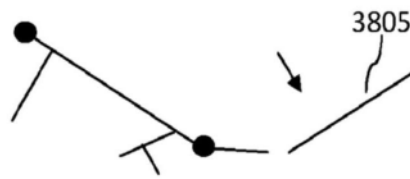


图45B

手指被抬起

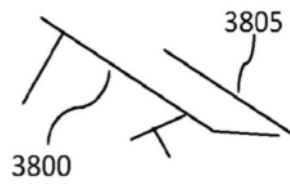


图45C

绘制“n”，并且然后触摸您想要向其应用圆角或圆弧的拐角，即，使外角变圆。

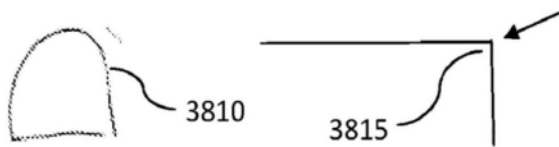


图46A

应用半径
向内或向外移动弧形拐角以增加或减小圆弧半径

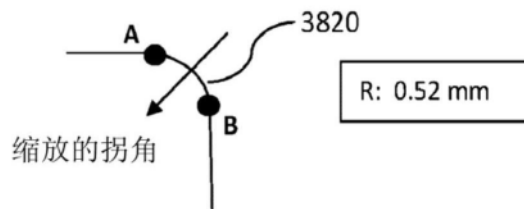


图46B

手指被抬起
触摸屏以应用和解除显示框

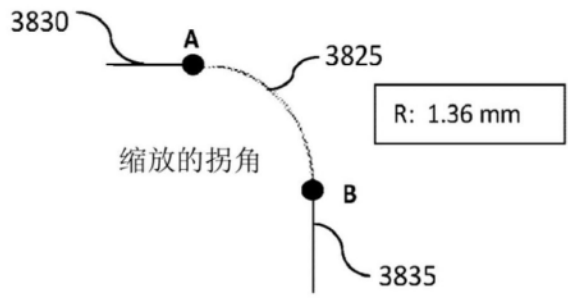


图46C



图46D

绘制“△”，并且然后触摸您想要向其应用倒角的拐角

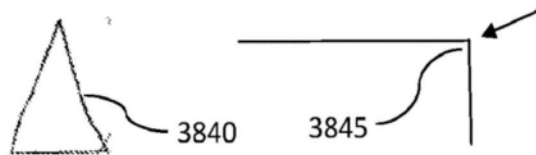


图47A

应用倒角
移动边缘“A”和/或“B”以实现所期倒角

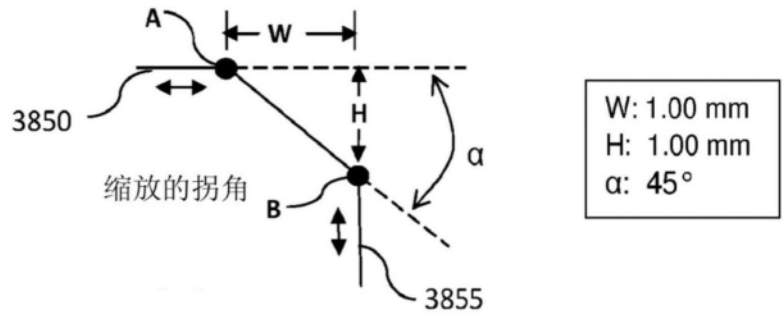


图47B

手指被抬起
触摸屏以应用和解除显示框

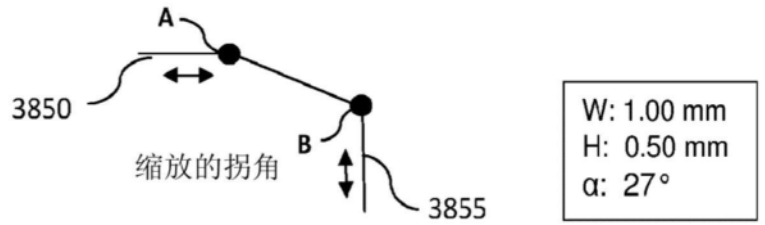



图47C



图47D

绘制“”来修剪对象

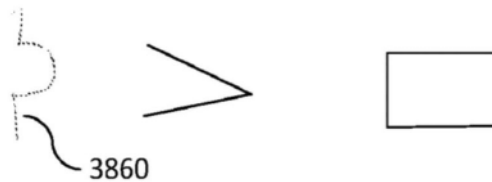


图48A

触摸目标对象

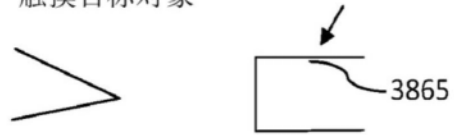


图48B

触摸参考对象

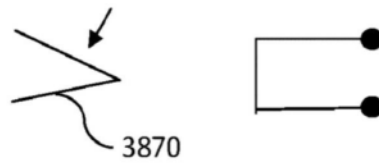


图48C

移动参考对象以实现所期修剪

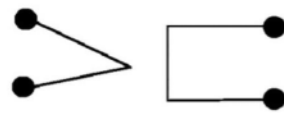


图48D

抬起手指以应用

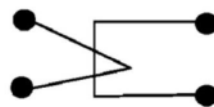


图48E

手指被抬起

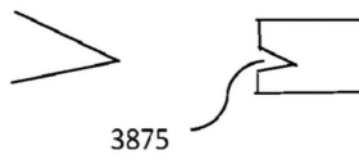


图48F

触摸对象以选择它

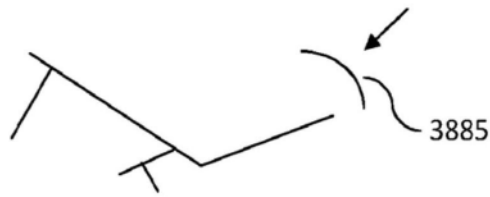


图49A

绘制“M”，并且然后将对象移到所期位置



图49B

抬起手指以应用

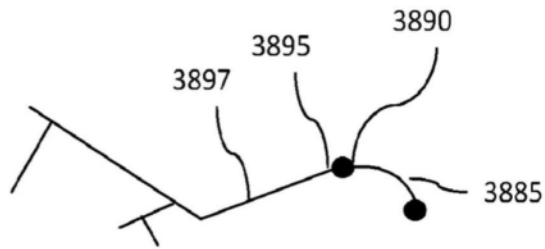


图49C

手指被抬起

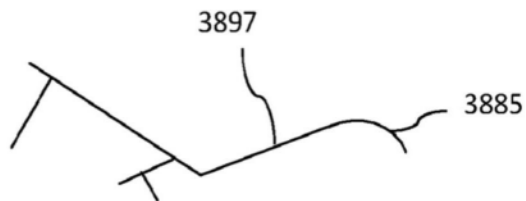


图49D

触摸“不捕捉”图标

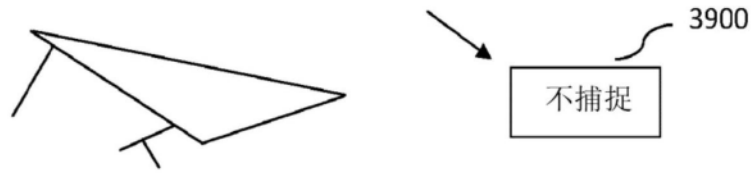


图50A

触摸您想要取消捕捉的交点

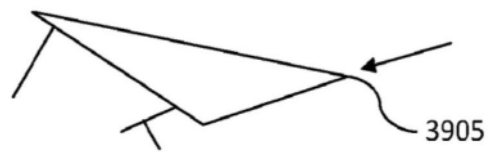


图50B

手指被抬起

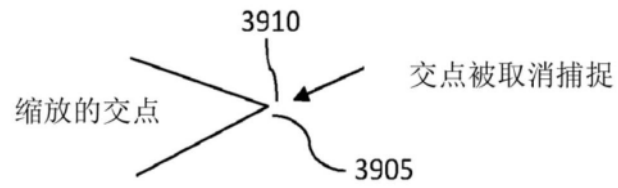


图50C

在任意位置触摸交点以缩小



图50D

触摸“不捕捉”图标

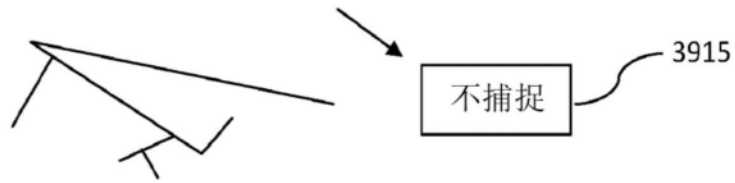


图51A

绘制“L”，并且然后将所期边缘跨上方的线向上移动到所期位置



图51B

抬起手指以应用

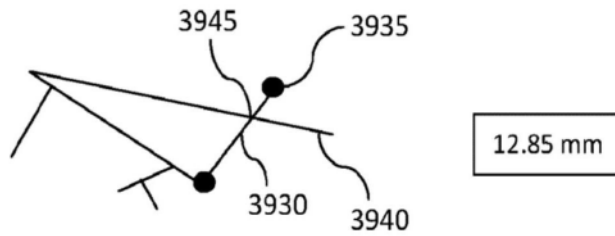


图51C

手指被抬起

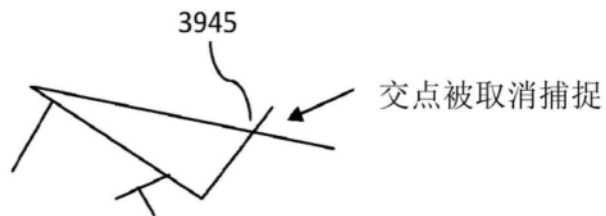


图51D

绘制“3”来修剪对象



图52A

移动参考对象以实现所期修剪

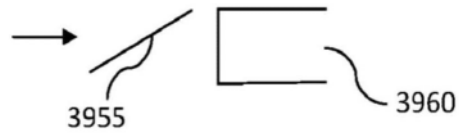


图52B

抬起手指以应用

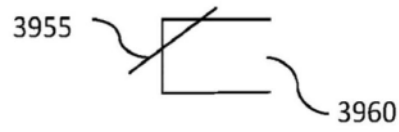


图52C

手指被抬起

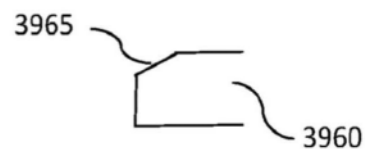


图52D

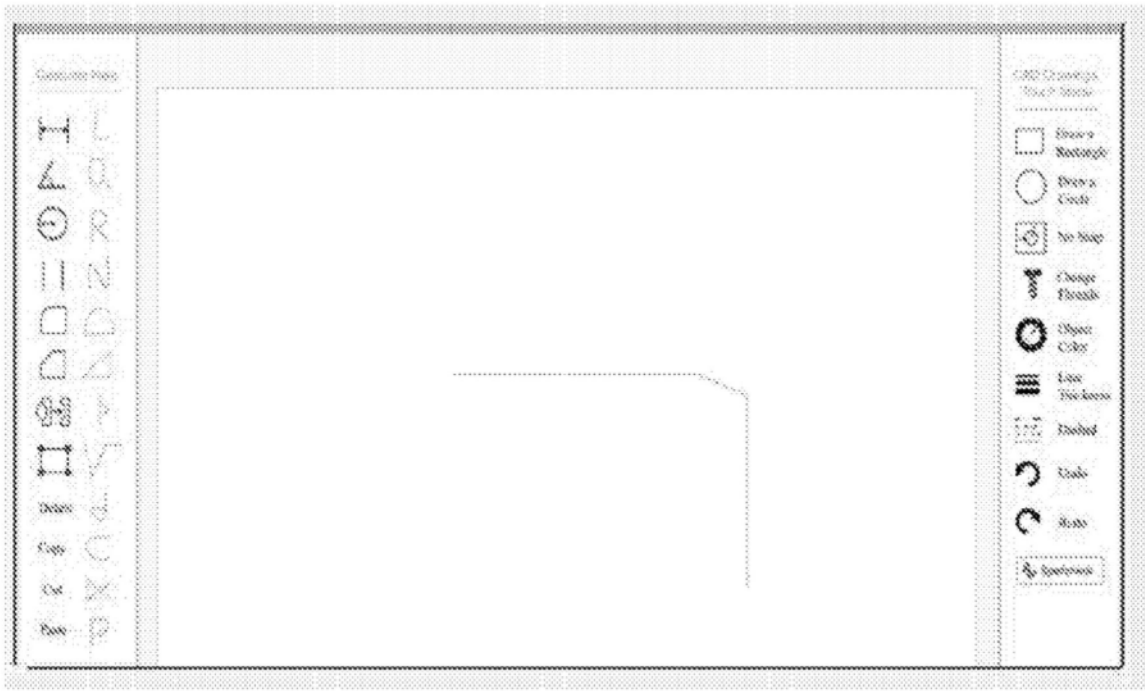


图53

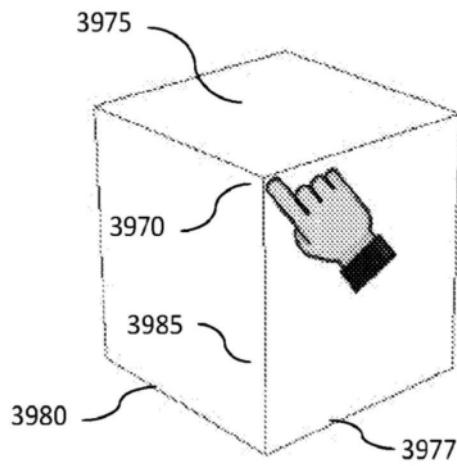


图54A

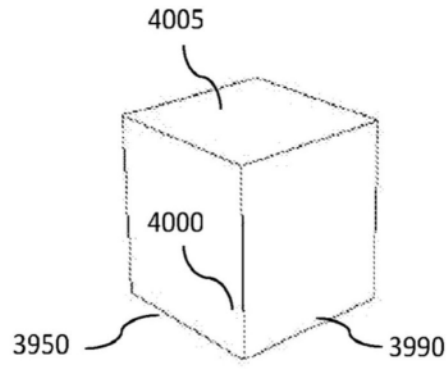


图54B

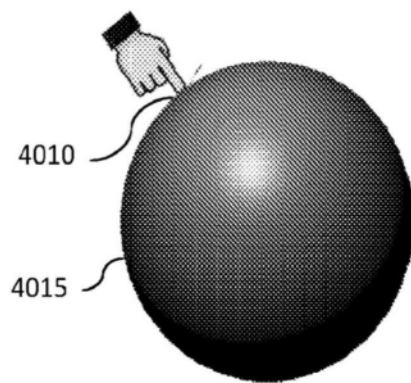


图54C

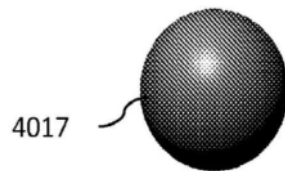


图54D

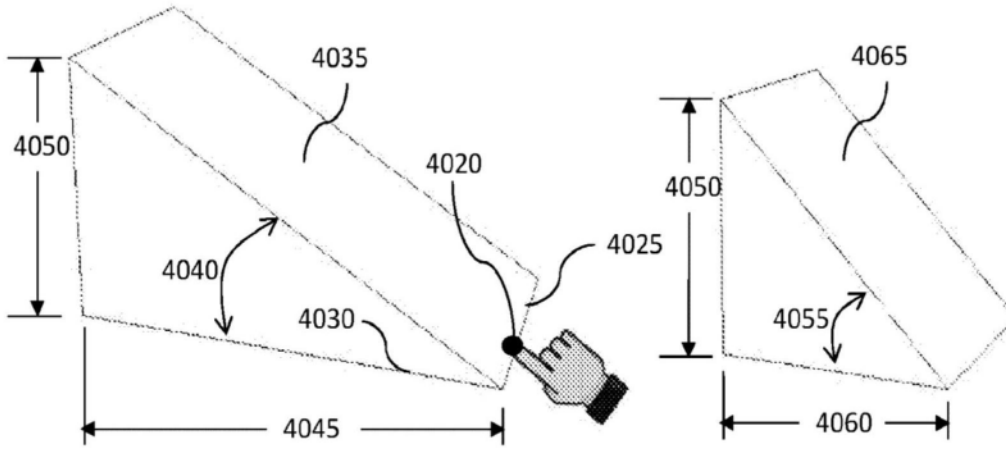


图 54E

图 54F

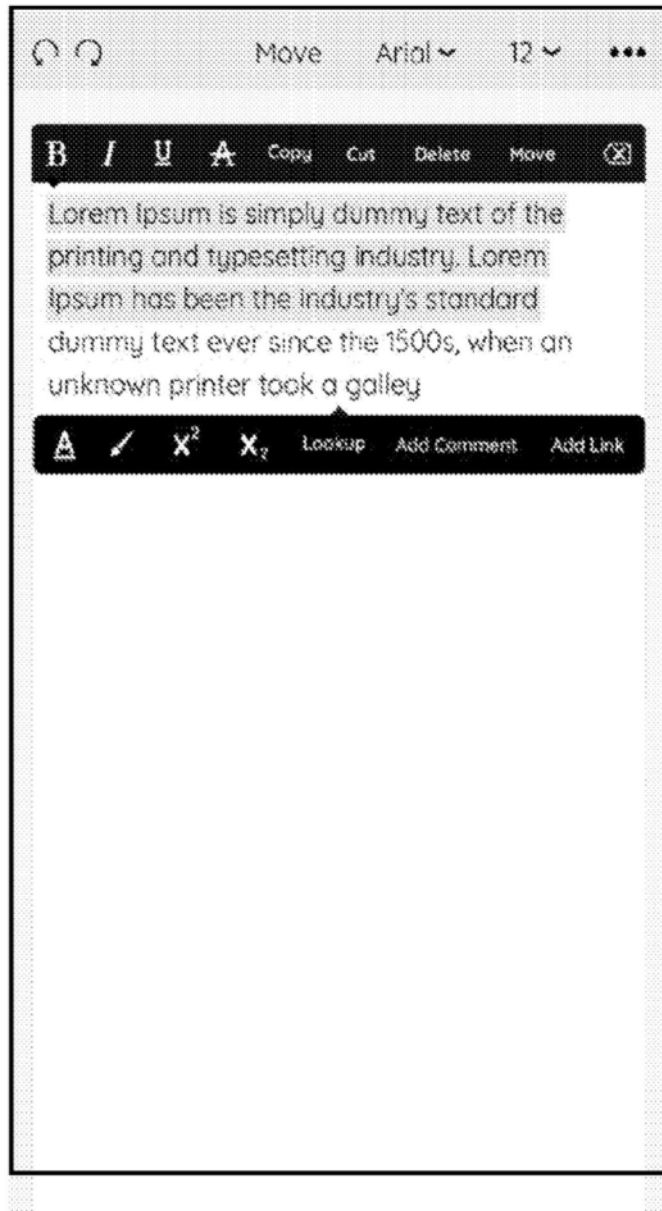


图55A

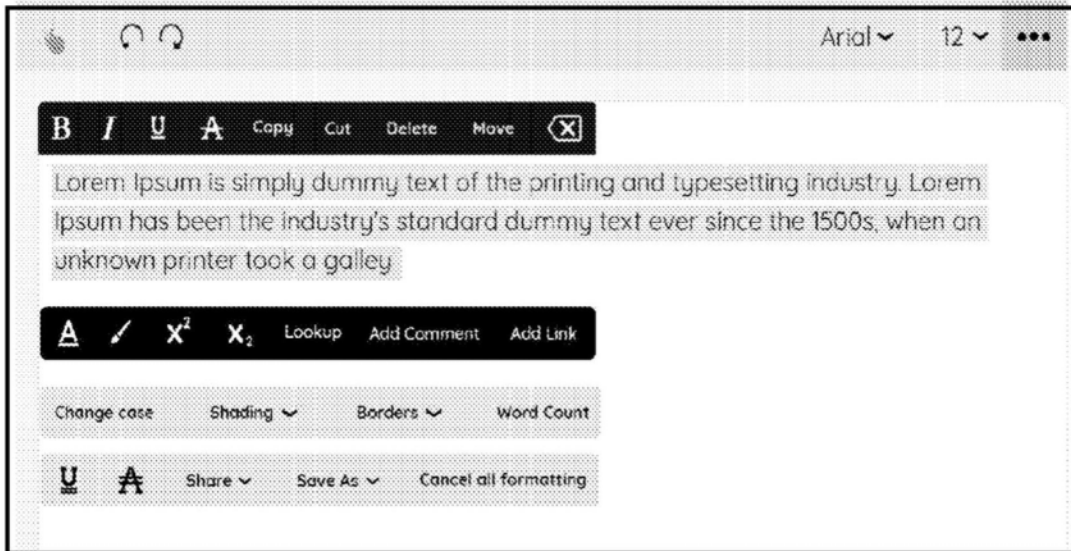


图55B

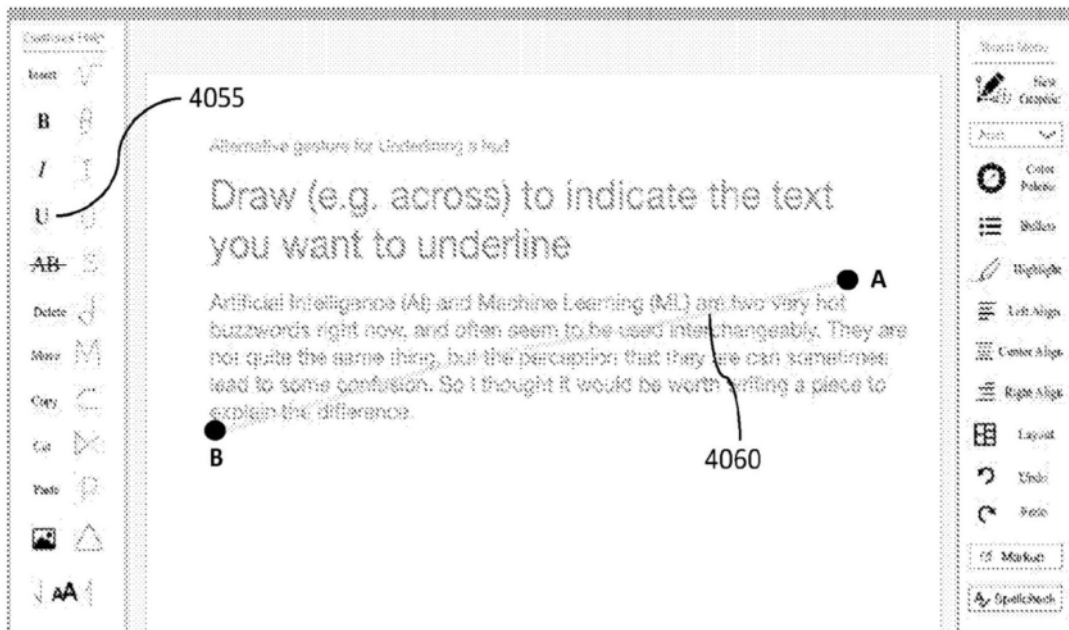


图56

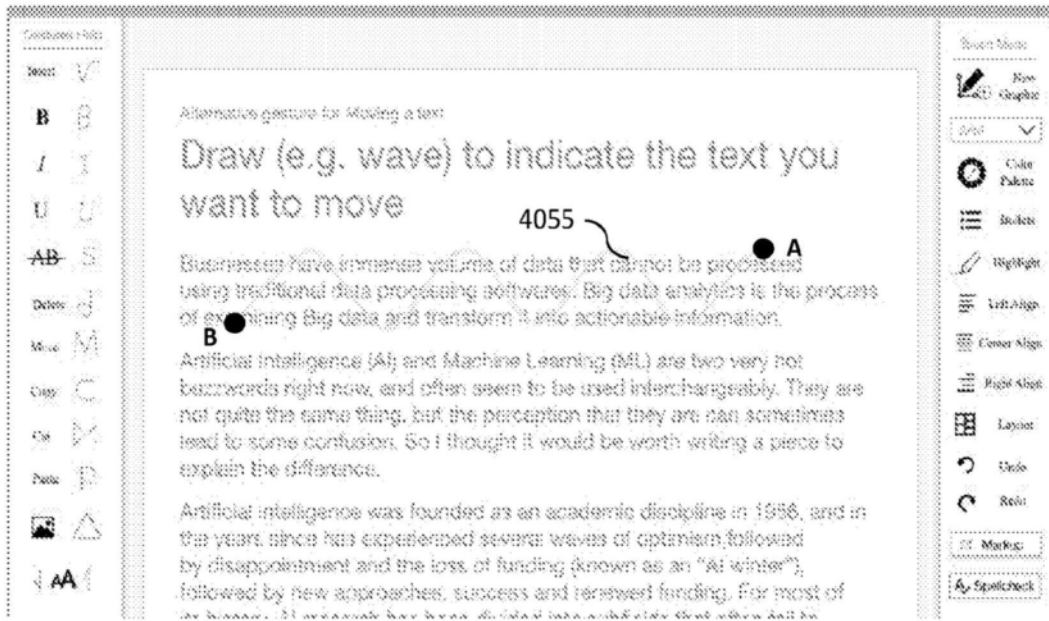


图57

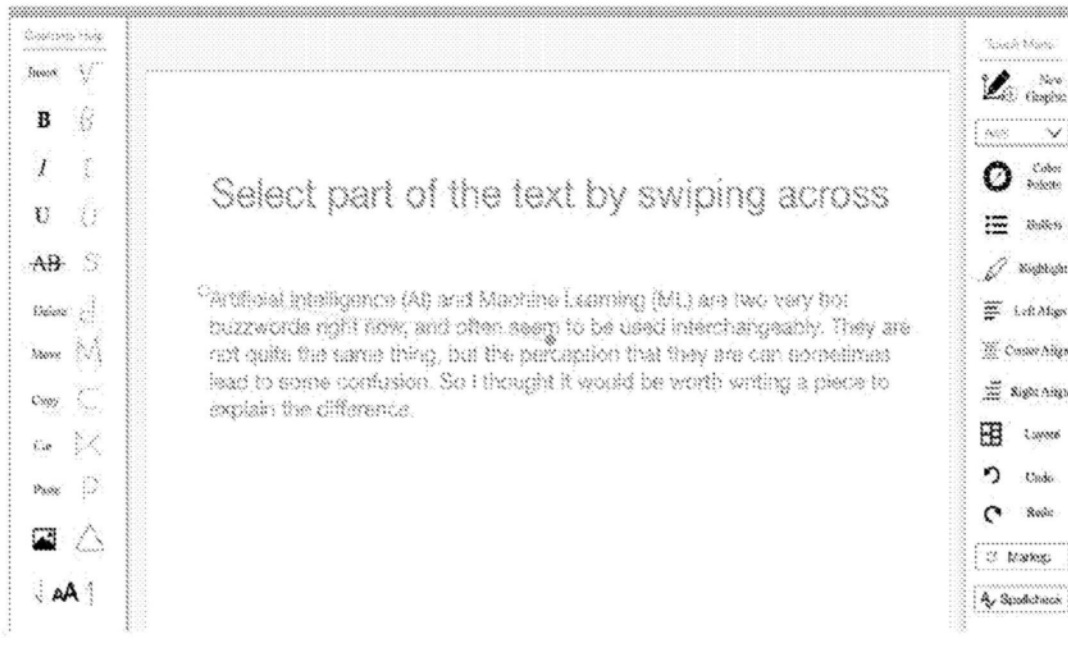


图58A

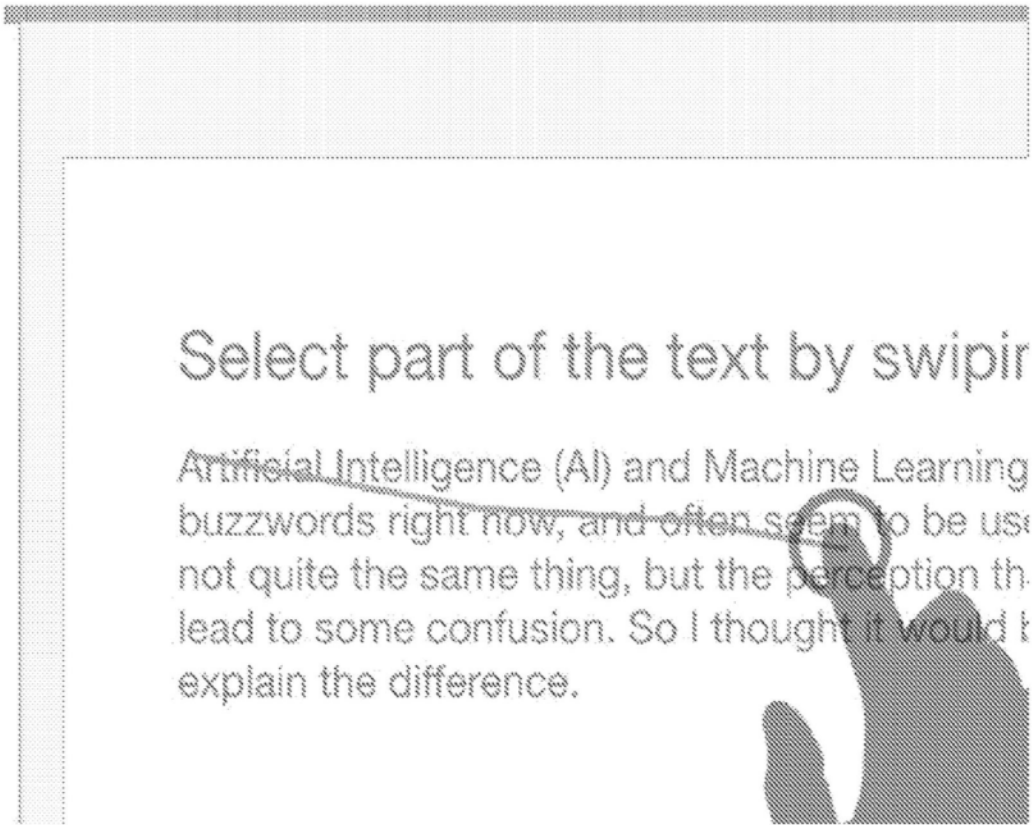


图58B