



(12) 发明专利

(10) 授权公告号 CN 102280098 B

(45) 授权公告日 2014. 11. 05

(21) 申请号 201010194227. 6

(22) 申请日 2010. 06. 08

(73) 专利权人 康佳集团股份有限公司

地址 518053 广东省深圳市南山区华侨城康佳集团

(72) 发明人 彭文剑

(74) 专利代理机构 深圳市顺天达专利商标代理有限公司 44217

代理人 高占元

(51) Int. Cl.

G09G 5/26 (2006. 01)

(56) 对比文件

CN 101510421 A, 2009. 08. 19,

US 2010/0079490 A1, 2010. 04. 01,

Kevin L. Kloker 等. “THE MOTOROLA

DSP96002 IEEE FLOATING-POINT DIGITAL

SIGNAL PROCESSOR”. 《Twenty-second

asilomar conference on signals, systems and computers》. 1989, 第 2480-2483 页.

审查员 乔元昆

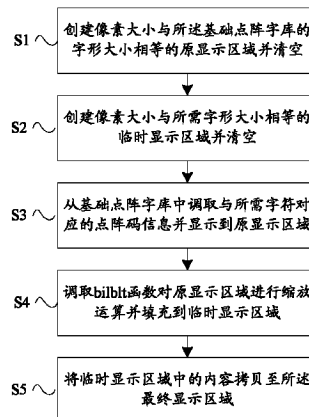
权利要求书1页 说明书5页 附图1页

(54) 发明名称

点阵字缩放的方法及系统

(57) 摘要

本发明涉及一种点阵字缩放的方法及系统, 用于根据基础点阵字库输出所需字形大小的字符至最终显示区域, 所述方法包括以下步骤: S1、创建像素大小与所述基础点阵字库的字形大小相等的原显示区域; S2、创建像素大小与所需字形大小相等的临时显示区域; S3、从所述基础点阵字库中调取与所需字符对应的点阵码信息并显示到原显示区域; S4、调取 bilblt 函数对所述原显示区域中的像素点信息进行缩放运算并填充到所述临时显示区域中; S5、将所述临时显示区域中的内容拷贝至所述最终显示区域。本发明利用硬件的 bilblt 功能, 由一个基础点阵字库平滑得到所需的其它字形大小的点阵点阵字符, 其算法简单、转换快速、效果好, 所需的存储空间较小, 同时又不占用 CPU 资源。



1. 一种点阵字缩放的方法,用于根据基础点阵字库输出所需字形大小的字符至最终显示区域,其特征在于,包括以下步骤:

S0、预存基础点阵字库;

S1、创建像素大小与所述基础点阵字库的字形大小相等的原显示区域并清空所述原显示区域中所有内容;

S2、创建像素大小与所需字形大小相等的临时显示区域,并清空所述临时显示区域中所有内容;

S3、根据所需字符从所述基础点阵字库中调取与该字符对应的点阵码信息,按所需颜色着色到所述原显示区域;

S4、调取 bilblt 函数对所述原显示区域中的像素点信息进行缩放运算并填充到所述临时显示区域中;

S5、对所述临时显示区域中的像素点信息逐点判断是否被着色,如果是则拷贝该像素点信息至所述最终显示区域的对应像素位置。

2. 根据权利要求 1 所述的点阵字缩放的方法,其特征在于,所述基础点阵字库为大小为 36*36 及以上的点阵字库。

3. 一种点阵字缩放的系统,用于根据基础点阵字库输出所需字形大小的字符至最终显示区域,其特征在于,包括:

存储模块(100),用于存储基础点阵字库;

原显示区域模块(200),与所述存储模块(100)相连,用于创建像素大小与所述基础点阵字库的字形大小相等的原显示区域并清空所述原显示区域中所有内容,随后根据所需字符从所述存储模块(100)中存储的基础点阵字库中调取与该字符对应的点阵码信息,按所需颜色着色到所述原显示区域;

临时显示区域模块(300),用于创建像素大小与所需字形大小相等的临时显示区域,并清空所述临时显示区域中所有内容;

缩放计算模块(400),与所述原显示区域模块(200)和临时显示区域模块(300)相连,用于调取 bilblt 函数对所述原显示区域中的像素点信息进行缩放运算并填充到所述临时显示区域中;

最终显示区域模块(500),与所述临时显示区域模块(300)相连,用于对所述临时显示区域中的像素点信息逐点判断是否被着色,如果是则拷贝该像素点至所述最终显示区域的对应像素位置。

4. 根据权利要求 3 所述的点阵字缩放的系统,其特征在于,所述基础点阵字库为大小为 36*36 及以上的点阵字库。

点阵字缩放的方法及系统

技术领域

[0001] 本发明涉及字符显示技术,更具体地说,涉及一种点阵字缩放的方法及系统。

背景技术

[0002] 在字符的输出过程中,需要使用到字库中存储的字符信息。字库按照其描述字符信息原理的不同分为点阵字库和矢量字库。其中,矢量字库记录了字符的轮廓信息,且具备无级缩放的功能,但构造字库的难度大成本高,同时字符输出的算法也较为复杂。

[0003] 点阵字库则描述了字符每个像素点上是否着色的信息。点阵字库构造简单,字符输出算法也很简单,同时字库占用的 CPU 和存储资源都比较低。点阵字库已经广泛的使用在各种嵌入式系统中。

[0004] 然而,点阵字库中的字符形状不能无级缩放。而在使用点阵字库的嵌入式系统中,往往会遇到需要多种不同大小字形的情况,在这种情况下,现有技术一般通过以下方法来实现:

[0005] 1、在系统中预备所需的多种大小字形的点阵字库。使用多个字形大小的点阵字库,显然需要成倍的存储空间,点阵字库一般用 1 位 (bit) 空间来描述一个像素是否着色的信息,那么对于一个 32*32 大小的 gb2312 点阵字库就需要使用 $7445*32*32/8 = 952960$ 字节存储空间,如果同时需要 24*24, 28*28, 32*32 等大小的字库,那么所需的存储空间是非常大的。

[0006] 2、先对字符的点阵信息进行扫描获得其轮廓信息,再对其进行矢量化,再放大到所需大小字形后通过画贝赛尔曲线或直线及填充操作等来输出显示。这种算法在本质上就是矢量算法。使用点阵字库矢量化来处理,除了算法本身难以实现外,在矢量化过程和字符输出过程的算法都涉及到贝赛尔曲线的复杂曲线及矢量运算,使得在运算能力比较低的嵌入式系统中导致字符输出显示速度慢,显示效率低下。

[0007] 此外,如果所需的几种字形大小成整数倍,那么可以通过对每个点放大整数倍的方法来输出显示。然而在实际应用中,绝大部分所需的字形大小并不是成整数倍的,而是比较临近的字形大小,比如 20*20, 24*24, 28*28 等等。

[0008] 点阵字库描述的是字符每个像素点的是否着色信息,要实现对其字形大小的缩放,如果是缩放整数倍,那么显然可以很简单的对每个点在水平方向和垂直方向进行相应倍数缩放就可以实现。然而在实际应用中,绝大部分所需的字形大小并不是成整数倍的,而是比较临近的字形大小,比如 20*20, 24*24, 28*28 等等,这些字形大小有较明显差距但有还不够成倍数关系。因此,无法简单地使用倍数缩放方法来进行输出,必须通过复杂的算法进行抽样、取样、补点等运算,对 CPU 要求比较高。而且缩放后的效果会产生不同程度的失真。

发明内容

[0009] 本发明要解决的技术问题在于,针对现有技术的点阵字库在需要输出不成整数倍

字形的字符时输出算法复杂且易失真的缺陷,提供一种点阵字缩放的方法。

[0010] 本发明解决其技术问题所采用的技术方案是:构造一种点阵字缩放的方法,用于根据基础点阵字库输出所需字形大小的字符至最终显示区域,包括以下步骤:

[0011] S1、创建像素大小与所述基础点阵字库的字形大小相等的原显示区域并清空所述原显示区域中所有内容;

[0012] S2、创建像素大小与所需字形大小相等的临时显示区域,并清空所述临时显示区域中所有内容;

[0013] S3、从所述基础点阵字库中调取与所需字符对应的点阵码信息并显示到所述原显示区域;

[0014] S4、调取 bilblt 函数对所述原显示区域中的像素点信息进行缩放运算并填充到所述临时显示区域中;

[0015] S5、将所述临时显示区域中的内容拷贝至所述最终显示区域。

[0016] 在本发明所述的点阵字缩放的方法中,所述步骤 S3 进一步包括:根据所需字符从基础点阵字库中调取与该字符对应的点阵码信息,按所需颜色着色到所述原显示区域。

[0017] 在本发明所述的点阵字缩放的方法中,所述步骤 S5 包括:对所述临时显示区域中的像素点信息逐点判断是否被着色,如果是则拷贝该像素点信息至所述最终显示区域的对应像素位置。

[0018] 在本发明所述的点阵字缩放的方法中,所述基础点阵字库为大小为 36*36 及以上的点阵字库。

[0019] 本发明还提供了一种点阵字缩放的系统,用于根据基础点阵字库输出所需字形大小的字符至最终显示区域,包括:

[0020] 存储模块,用于存储基础点阵字库;

[0021] 原显示区域模块,与所述存储模块相连,用于创建像素大小与所述基础点阵字库的字形大小相等的原显示区域并清空所述原显示区域中所有内容,随后从所述存储模块中存储的基础点阵字库中调取与所需字符对应的点阵码信息并显示到所述原显示区域;

[0022] 临时显示区域模块,用于创建像素大小与所需字形大小相等的临时显示区域,并清空所述临时显示区域中所有内容;

[0023] 缩放计算模块,与所述原显示区域模块和临时显示区域模块相连,用于调取 bilblt 函数对所述原显示区域中的像素点信息进行缩放运算并填充到所述临时显示区域中;

[0024] 最终显示区域模块,与所述临时显示区域模块相连,用于将所述临时显示区域中的内容拷贝至所述最终显示区域。

[0025] 在本发明所述的点阵字缩放的系统中,所述原显示区域模块根据所需字符从所述存储模块中存储的基础点阵字库中调取与该字符对应的点阵码信息,按所需颜色着色到所述原显示区域。

[0026] 在本发明所述的点阵字缩放的系统中,所述最终显示区域模块对所述临时显示区域中的像素点信息逐点判断是否被着色,如果是则拷贝该像素点信息至所述最终显示区域的对应像素位置。

[0027] 在本发明所述的点阵字缩放的系统中,所述基础点阵字库为大小为 36*36 及以上

的点阵字库。

[0028] 实施本发明的点阵字缩放的方法及系统,具有以下有益效果:本发明利用硬件的 bilblt 功能,由一个基础点阵字库平滑得到所需的其它字形大小的点阵点阵字符;在该方法中,由于只需存储一个基础点阵字库因此所需的存储空间较小;此外,通过利用 bilblt 函数进行转换,其算法简单、转换快速、效果好,同时又不占用 CPU 资源。

附图说明

[0029] 下面将结合附图及实施例对本发明作进一步说明,附图中:

[0030] 图 1 是本发明优选实施例中点阵字缩放的方法流程图;

[0031] 图 2 是本发明优选实施例中点阵字缩放的系统的模块示意图。

具体实施方式

[0032] 为了使本发明的目的、技术方案及优点更加清楚明白,以下结合附图及实施例,对本发明进行进一步详细说明。

[0033] 由于硬件性能的提高,很多硬件的显示模块都提供了硬件 bilblt 功能,硬件 bilblt 功能是指通过硬件(多为专用的 DSP 处理器)来对显示内容进行无级缩小,无级放大,搬运,混合等处理。具有速度快,效果好,不占用 CPU 资源等优势。本发明在点阵字的缩放过程中,利用到上述硬件的 bilblt 功能,从而通过简单的软件流程来实现对点阵字库中字符进行缩放以获得需要大小的字符输出。

[0034] 请参阅图 1,为本发明优选实施例中点阵字缩放的方法流程图。如图 1 所示,本发明提供了一种点阵字缩放的方法,用于根据基础点阵字库输出所需字形大小的字符至最终显示区域。该方法首先需要在系统中存储基础点阵字库,即任意选择一种大小的点阵字库进行存储。考虑到过小的点阵字库在转换过程中可能会存在像素不全不均匀和笔画丢失情况,因此为了保证转换的效果,也可以优选字形较大的点阵字库,例如 36*36 以上的点阵字库。

[0035] 该方法具体包括以下步骤:

[0036] 首先在步骤 S1 中,创建像素大小与所述基础点阵字库的字形大小相等的原显示区域并清空所述原显示区域中所有内容。

[0037] 随后在步骤 S2 中,创建像素大小与所需字形大小相等的临时显示区域,并清空所述临时显示区域中所有内容。

[0038] 随后在步骤 S3 中,从所述基础点阵字库中调取与所需字符对应的点阵码信息并显示到所述原显示区域。在本发明的进一步优选实施例中,步骤 S3 具体为:根据所需字符从基础点阵字库中调取与该字符对应的点阵码信息,按所需颜色着色到所述原显示区域。

[0039] 随后在步骤 S4 中,调取 bilblt 函数对所述原显示区域中的像素点信息进行缩放运算并填充到所述临时显示区域中。

[0040] 随后在步骤 S5 中,将所述临时显示区域中的内容拷贝至所述最终显示区域。在本发明的进一步优选实施例中,步骤 S5 具体为对所述临时显示区域中的像素点信息逐点判断是否被着色,如果是则拷贝该像素点信息至所述最终显示区域的对应像素位置。

[0041] 下面以 24*24 大小的点阵字库为基础点阵字库进行说明。如果需要输出不同字形

大小,例如 28*28 的字符,则其缩放过程如下:

[0042] 1、先建立一个 24*24 像素的原显示区域,并清空该区域内所有内容。

[0043] 2、建立一个 28*28 像素的临时显示区域,并清空该区域内所有内容。

[0044] 3、对于某个要输出的字符,例如“大”字,可以从基础点阵字库中获得该字符“大”的点阵码信息,按所需颜色着色到 24*24 像素的显示区域。

[0045] 4、利用硬件 bilblt 功能该函数将 24*24 像素的原显示区域放大到 28*28 像素的临时显示区域。即调取 bilblt 函数对所述原显示区域中的像素点信息进行放大运算并填充到临时显示区域中。

[0046] 5、对 28*28 像素的临时显示区域内容逐点判断是否被着色,如果是则拷贝到该像素点内容到最终显示区域的相应位置。

[0047] 从上面过程可以看到,基础点阵字库的字形大小为 24*24,而输出到最终显示区域的字形大小为 28*28,两者不成整数倍数,因此无法用现有的直接缩放的方法来实现。而本发明的方法则可以快速地对其进行缩放,且最后输出的字符像素均匀,曲线平滑。

[0048] 请参阅图 2,为本发明优选实施例中点阵字缩放的系统的模块示意图。如图 2 所示,本发明提供了一种点阵字缩放的系统,用于根据基础点阵字库输出所需字形大小的字符至最终显示区域。该系统主要包括存储模块 100、原显示区域模块 200、临时显示区域模块 300、缩放计算模块 400 和最终显示区域模块 500。

[0049] 其中,存储模块 100,用于存储基础点阵字库。同样该存储模块 100 存储的基础点阵字库可以选择任意大小的点阵字库。考虑到过小的点阵字库在转换过程中可能会存在像素不全不均匀和笔画丢失情况,因此为了保证转换的效果,也可以优选字形较大的点阵字库,例如 36*36 以上的点阵字库。

[0050] 原显示区域模块 200 与所述存储模块 100 相连。原显示区域模块 200 首先创建像素大小与所述基础点阵字库的字形大小相等的原显示区域并清空所述原显示区域中所有内容,随后从所述存储模块 100 中存储的基础点阵字库中调取与所需字符对应的点阵码信息并显示到所述原显示区域。在本发明更进一步的优选实施例中,原显示区域模块 200 根据所需字符从所述存储模块中存储的基础点阵字库中调取与该字符对应的点阵码信息,按所需颜色着色到所述原显示区域。

[0051] 临时显示区域模块 300 用于创建像素大小与所需字形大小相等的临时显示区域,并清空所述临时显示区域中所有内容。

[0052] 缩放计算模块 400 与原显示区域模块 200 和临时显示区域模块 300 相连。缩放计算模块 400 用于在原显示区域和临时显示区域创建后,且临时显示区域已显示了从基础点阵字库中调取的字符图像后,调取 bilblt 函数对原显示区域中的像素点信息进行缩放运算并填充到所述临时显示区域中。

[0053] 最终显示区域模块 500,与所述临时显示区域模块 300 相连,用于将临时显示区域中的内容拷贝至所述最终显示区域。在本发明更进一步的优选实施例中,所述最终显示区域模块 500 对所述临时显示区域中的像素点信息逐点判断是否被着色,如果是则拷贝该像素点信息至所述最终显示区域的对应像素位置。

[0054] 综上所述,本发明的点阵字缩放的方法及系统利用硬件的 bilblt 功能,由一个基础点阵字库平滑得到所需的其它字形大小的点阵点阵字符;在该方法中,由于只需存储一

个基础点阵字库因此所需的存储空间较小 ;此外,通过利用 bilb1t 函数进行转换,其算法简单、转换快速、效果好,同时又不占用 CPU 资源。

[0055] 本发明是根据特定实施例进行描述的,但本领域的技术人员应明白在不脱离本发明范围时,可进行各种变化和等同替换。此外,为适应本发明技术的特定场合或材料,可对本发明进行诸多修改而不脱离其保护范围。因此,本发明并不限于在此公开的特定实施例,而包括所有落入到权利要求保护范围的实施例。

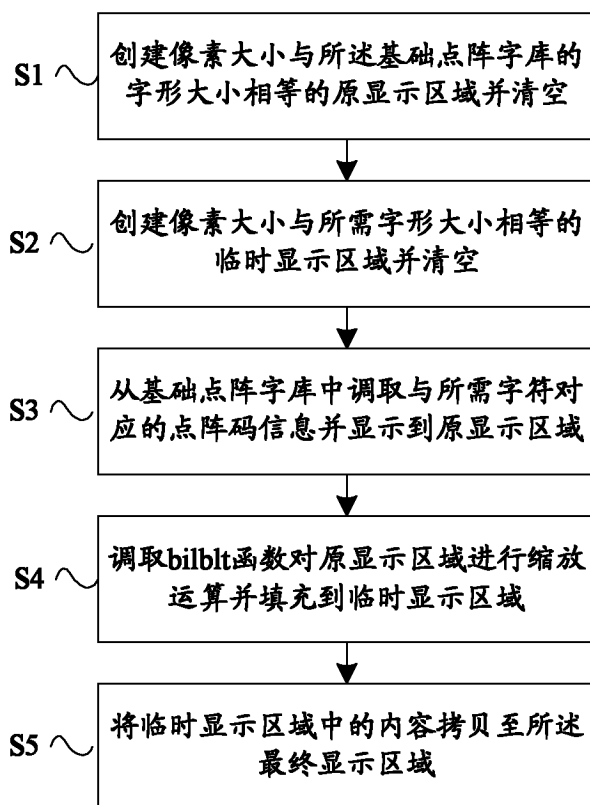


图 1

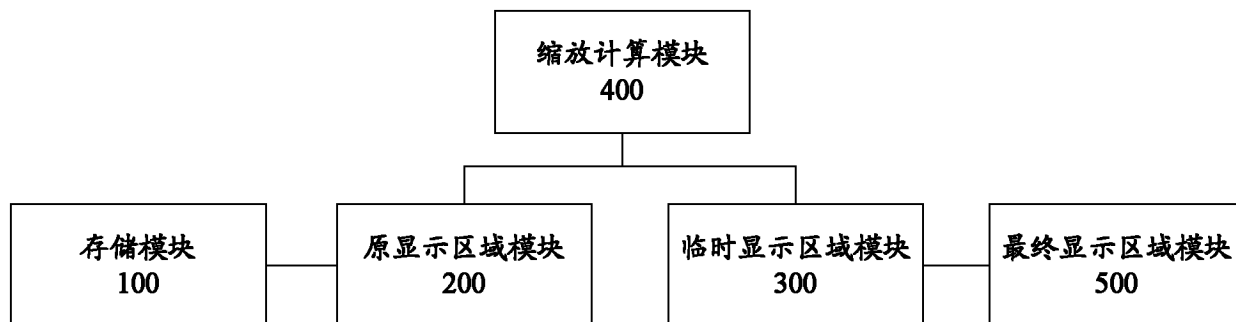


图 2