

[19] 中华人民共和国国家知识产权局



[12] 发明专利说明书

专利号 ZL 02150636.1

[51] Int. Cl.

G06F 3/03 (2006.01)

G06T 7/20 (2006.01)

G06K 11/06 (2006.01)

[45] 授权公告日 2008 年 2 月 6 日

[11] 授权公告号 CN 100367165C

[22] 申请日 2002.11.6 [21] 申请号 02150636.1

[30] 优先权

[32] 2001.11.6 [33] US [31] 10/011593

[73] 专利权人 全视技术有限公司

地址 美国加利福尼亚州

[72] 发明人 X·罗

[56] 参考文献

CN1215286A 1999.4.28

US5786804A 1998.7.28

US6281882B1 2001.8.28

US-5557341A 1996.9.17

JP2000-267799A 2000.9.29

US6256016B1 2001.7.3

KR2001-0045766A 2001.6.5

EP0666696A2 1995.8.9

US5578813A 1996.11.26

审查员 吴黄飞

[74] 专利代理机构 中国专利代理(香港)有限公司

代理人 邹光新 张志醒

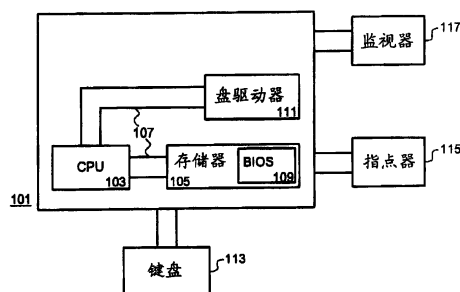
权利要求书 3 页 说明书 10 页 附图 4 页

[54] 发明名称

确定光鼠标相对运动的方法和装置

[57] 摘要

本发明揭示了一种确定两个图像之间的运动向量的方法。光鼠标相继获取两个图像。这种方法包括首先对于在第一图像与第二图像之间的多个方向的位移计算方向相关值。选择最小的方向相关值，与门限值进行比较。如果计算得的最小的方向相关值小于门限，就确定与这个最小的方向相关值关联的运动向量。然而，如果最小的方向相关值不小于门限，就对第一和第二图像压缩后再进行计算。这种压缩一直继续到确定合格的运动向量。



1. 一种确定在两个图像之间的运动向量的方法，所述方法包括下列步骤：

(a) 截获一个第一图像和一个第二图像；

(b) 对于在所述第一图像与所述第二图像之间的多个方向的位移计算相应的方向相关值；

(c) 从所述方向相关值中确定最小的方向相关值；

(d) 确定所计算的最小的方向相关值是否小于一个预定门限值；以及

(i) 如果是这样的话，确定与所述最小方向相关值关联的运动向量，以及

(ii) 如果不是这样的话，将所述第一图像和所述第二图像压缩成使所述第一图像和第二图像具有较小的规模，再重复(b)至(d)，直到确定所述运动向量。

2. 权利要求1的方法，其中所述第一图像和第二图像是由一个并入光鼠标的图像传感器截获的。

3. 权利要求1的方法，其中所述第二图像是在所述第一图像截获后截获的。

4. 权利要求1的方法，其中所述压缩通过将一个规模为 $S \times S$ 的像素块平均成一个压缩图像的单个像素来实现的。

5. 权利要求4的方法，其中所述 S 是一个整数，随着执行进一步的压缩而递增。

6. 权利要求1的方法，其中所述方向相关值为所述第一图像与第二图像的相应像素之间的差的绝对值之和。

7. 权利要求4的方法，其中所述运动向量具有一个方向和一个幅度，所述运动向量的幅度与对所述第一图像和所述第二图像执行的压缩的程度成正比。

8. 权利要求7的方法，其中所述幅度取决于 S 。

9. 权利要求1的方法，其中所述截获一个第一图像和一个第二图像包括截获一个第一 $M \times N$ 规模图像和一个第二 $M \times N$ 规模图像，其中每一个图像包括 $M \times N$ 数据值， M 和 N 大于2。

10. 权利要求9的方法，其中压缩所述第一图像和所述第二图像

包括：将所述第一图像和所述第二图像压缩成 $(M/S) \times (N/S)$ 规模图像， $(M/S) \times (N/S)$ 规模图像包括 $(M/S) \times (N/S)$ 数据值，其中 S 是大于1的整数，以及其中压缩图像的每一个数据点的值是压缩数据点替代的原始图像的 $S \times S$ 数据值的平均值，从而使所述第一图像和第二图像具有较小的规模；用增加的 S 重复(b)至(d)，直到确定所述运动向量，其中， M 和 N 是 S 的偶倍数。

11. 一种输出具有幅度和方向的运动向量的集成电路，所述运动向量表示光鼠标的相对运动，所述集成电路包括：

一个在一个衬底上形成的用来截获一个第一图像和一个第二图像的传感器阵列；

在所述衬底上形成的用来存储所述第一图像和所述第二图像的存储器；以及

一个在所述衬底上形成的信号处理器，用来：

(a) 对于在所述第一图像与所述第二图像之间的多个方向的位移计算相应的方向相关值；

(b) 从所述方向相关值中确定最小的方向相关值；

(c) 确定所计算的最小的方向相关值是否小于一个预定门限值；以及

(i) 如果是这样的话，确定与所述最小方向相关值关联的运动向量，以及

(ii) 如果不是这样的话，将所述第一图像和所述第二图像压缩成使所述第一图像和第二图像具有较小的规模，再重复(a)至(c)，直到确定所述运动向量。

12. 权利要求11的集成电路，其中所述压缩是通过将一个规模为 $S \times S$ 的像素块平均成一个压缩图像的单个像素来实现的。

13. 权利要求12的集成电路，其中所述 S 是一个整数，随着执行进一步的压缩而递增。

14. 权利要求11的集成电路，其中所述方向相关值为所述第一图像与第二图像的相应像素之间的差的绝对值之和。

15. 权利要求13的集成电路，其中所述运动向量的幅度与对所述第一图像和所述第二图像执行的压缩的程度成正比。

16. 权利要求15的集成电路，其中所述幅度取决于 S 。

17. 权利要求 11 的集成电路, 其中压缩所述第一图像和所述第二图像包括压缩所述第一图像和所述第二图像使所述第一图像和第二图像具有较小的规模, 其中通过如下所述实现所述的压缩:

将所述图像分成多个像素块, 每一个像素块具有规模 $S \times S$, S 是大于 1 的整数,

求所述多个像素块的每一个像素块的平均值为一个压缩的像素;
以及

由从所述多个像素块的每一个像素块获得的所述一个压缩的像素的集合形成压缩的图像, 其中, 所述第一图像和所述第二图像包括数量为 S 的偶倍数的多个像素,

用增加的 S 重复 (b) 至 (d), 直到确定所述运动向量。

确定光鼠标相对运动的方法和装置

技术领域

本发明与光鼠标有关，具体地说，涉及用降低的计算要求确定光鼠标的快速运动的方法。

背景技术

有着各种类型的指点器用作输入装置，使用户可以操纵计算机的光标。在笔记本电脑中，通常用电容触摸屏来操纵光标。对于台式计算机来说，大多数都是用计算机鼠标作为指点器。鼠标用来对个人计算机或工作站的显示屏幕进行光标操作。光标操作包括诸如将光标从显示屏幕的一个区域快速移动到另一个区域或选择显示屏幕上的对象之类的操作。

在现有技术的“滚球”鼠标中，用户通过在一个诸如鼠标垫之类的表面上移动鼠标来控制光标，使得光标在显示屏幕上与鼠标的运动成比例地在一个方向上移动一段距离。然而，滚球鼠标有着与许多其他具有机械零件的器件共同的缺点。也就是说，机械零件会随时间磨损、变脏或损坏，如果这样，就不再能精确地操纵光标了。

近来，业已开发了一些采用图像传感器的“光”鼠标。例如，美国专利No. 6,281,882揭示了一种光鼠标，这种光鼠标可以产生平行光束对放置鼠标的表面进行照射。图像传感器的传感器阵列部分随着鼠标在一个表面上的移动接收表面的反射图像。然后，将相继反射图像的改变确定为鼠标的运动，于是就可以将它解释为所需的光标移动。

'882专利提供了有关光鼠标技术的极好的素材，详细说明了光鼠标的物理构成、探照光的产生、截获反射光形成图像帧的光学透镜系统和确定运动的方法。本发明与根据图像帧确定“运动向量”的方法有关。在这里所用的所谓“运动向量”描述了鼠标运动的幅度和方向。运动向量在提供给所关联的计算设备时可以用来操纵光标。

'882专利参考了先前的一些详细说明可以确定运动向量的各种方法的有关专利，包括美国专利No. 5,578,813、美国专利No. 5,644,139和美国专利No. 5,786,804。总的来说，通过对相继图像帧的相互比较来确定鼠标的相对运动。传感器阵列产生由各个像素形成的表面二维

图像。在'139专利所揭示的方法中，首先将第一帧存储起来。再截获和存储随后的第二帧。将第二帧与第一帧进行比较，确定运动向量。具体地说，对于后续帧的在每个紧邻像素的方向上的每个可能位移确定一个相关值。对于二维阵列来说，这相当于需要进行8个相关计算。此外，还要执行考虑后续图像没有位移的相关计算。因此，相对第一帧要执行总共9个相关计算。产生最大相关性的位移就给出了运动向量。这种技术的更详细的情况可以参见'139专利。

然而，由于用在每个方向位移单个像素执行相关计算，因此限制了光鼠标可以移动(同时仍然提供可用的信号)的最高速度。具体地说，在相继截获的图像之间光鼠标不能移动一个以上的像素，否则就会明显地增大相关计算量。因此，如果在相继图像之间光鼠标移动一个至两个像素，就必须执行25个相关计算。如果在相继图像之间光鼠标移动二至三个像素，就必须执行49个相关计算。

发明内容

为了克服上述已有技术存在的不足，本申请提出了确定光鼠标相对运动的方法和装置。

根据本发明的第一方面，提供了一种确定在两个图像之间的运动向量的方法，所述方法包括下列步骤：

- (a) 截获一个第一图像和一个第二图像；
- (b) 对于在所述第一图像与所述第二图像之间的多个方向的位移计算相应的方向相关值；
- (c) 从所述方向相关值中确定最小的方向相关值；
- (d) 确定所计算的最小的方向相关值是否小于一个预定门限值；以及

(i) 如果是这样的话，确定与所述最小方向相关值关联的运动向量，以及

(ii) 如果不是这样的话，将所述第一图像和所述第二图像压缩成使所述第一图像和第二图像具有较小的规模，再重复(b)至(d)，直到确定所述运动向量。

根据本发明的第二方面，提供了一种输出具有幅度和方向的运动向量的光鼠标，所述运动向量表示光鼠标的相对运动，所述光鼠标包括：

一个截获一个第一图像和一个第二图像的传感器阵列；
存储所述第一图像和所述第二图像的存储器；以及
一个信号处理器，用来：

(a) 对于在所述第一图像与所述第二图像之间的多个方向的位移计算相应的方向相关值；

(b) 从所述方向相关值中确定最小的方向相关值；

(c) 确定所计算的最小的方向相关值是否小于一个预定门限值；以及

(i) 如果是这样的话，确定与所述最小方向相关值关联的运动向量，以及

(ii) 如果不是这样的话，将所述第一图像和所述第二图像压缩成使所述第一图像和第二图像具有较小的规模，再重复(a)至(c)，直到确定所述运动向量。

根据本发明的第三方面，提供了一种输出具有幅度和方向的运动向量的集成电路，所述运动向量表示光鼠标的相对运动，所述集成电路包括：

一个在一个衬底上形成的用来截获一个第一图像和一个第二图像的传感器阵列；

在所述衬底上形成的用来存储所述第一图像和所述第二图像的存储器；以及

一个在所述衬底上形成的信号处理器，用来：

(a) 对于在所述第一图像与所述第二图像之间的多个方向的位移计算相应的方向相关值；

(b) 从所述方向相关值中确定最小的方向相关值；

(c) 确定所计算的最小的方向相关值是否小于一个预定门限值；以及

(i) 如果是这样的话，确定与所述最小方向相关值关联的运动向量，以及

(ii) 如果不是这样的话，将所述第一图像和所述第二图像压缩成使所述第一图像和第二图像具有较小的规模，再重复(a)至(c)，直到确定所述运动向量。

附图说明

在这些附图中，相同的标注数字所标的是本发明的这些非限制和非穷举性的实施例的各个示意图中相同的部分，其中：

图1为现有技术的采用指点器的个人计算机的示意图；

图2为适合与图1的光鼠标配合使用的集成电路的示意图；

图3为由一些配置成二维阵列的独立像素组成的传感器阵列的示意图；

图4为图3传感器阵列按照块规模为 2×2 组合后的示意图。

具体实施方式

在以下说明中，提供了许多具体细节，诸如各个系统部件的标识之类，以便对本发明的实施例有全面的理解。然而，熟悉该技术领域的人员可以看到，本发明可以在没有这些具体细节中的一个或多个细节或者用其他方法、部件、材料等的情况下实现。在其他一些例子中，没有示出或详细说明众所周知的结构、材料或操作，以免反而使本发明的各个实施例的特征模糊不清。

在本说明中，所谓本发明的“一个实施例”是指结合这个实施例所揭示的具体特色、结构或特征包括在本发明的至少一个实施例内。因此，在本说明中各处出现的“在一个实施例中”不一定都是指同一个实施例。此外，这些具体特色、结构或特征可以用任何适当方式在一个或多个实施例中组合在一起。

图1和有关说明旨在对可以实现本发明的适当计算环境进行简短的总体说明。虽然没有必要，但至少部分地将在由个人计算机或其他计算装置执行的诸如程序模块之类的计算机可执行指令的一般环境中对本发明进行说明。概括地说，程序模块包括执行具体任务或实现具体抽象数据类型的例行程序、对象、部件、数据结构等。此外，熟悉该技术领域的人员可以理解，本发明可以用于其他计算机系统配置，包括手持设备、多处理器系统、基于微处理器的或可编程用户电子设备、网络PC、微型计算机、主计算机等。

参见图1，本发明的典型环境包括呈现为传统的个人计算机101形式的通用计算设备，它包括处理单元103、系统存储器105和将包括系统存储器105在内的各个系统部件连接到处理单元103上的系统总线107。系统总线107可以是包括存储器总线或存储控制器、外围总线和

使用任何总线体系结构的局部总线几种类型总线结构中任何一种总线结构。系统存储器105可以包括只读存储器(ROM)、闪速存储器或随机存取存储器(RAM)。含有有助于诸如在启动期间之类的在个人计算机101内一些部件之间传送信息的基本例行程序的基本输入输出109(BIOS)存储在存储器105内。

此外,个人计算机101还包括盘驱动器111,诸如软盘驱动器或CD盘驱动器。盘驱动器111接至系统总线107。盘驱动器111和关联的计算机可读媒体提供了计算机可读指令、数据结构、程序模块和个人计算机101的其他数据的非易失存储。

虽然在这里所说明的典型环境用了盘驱动器111,但熟悉该技术领域的人员应该理解,在这典型操作环境内也可以使用其他类型的可以存储计算机可接入的数据的计算机可读媒体,诸如盒式磁带、闪速存储卡、数字视盘之类。

在存储器105内可以存储许多程序模块,包括操作系统、一个或多个应用程序、其他程序模块和程序数据。用户可以通过诸如键盘113和光鼠标115之类的输入装置将命令和信息输入个人计算机101。这些和其他一些输入装置通常通过接到系统总线107上的串行端口接口与处理单元103连接。监视器117或其他类型的显示设备通过诸如显示卡之类的接口也接到系统总线107上。

个人计算机101可以在一个网络环境内进行操作,与一个或多个远程计算机逻辑连接。远程计算机可以是另一个个人计算机、服务器、路由器、网络PC、对等设备或其他网络节点,通常包括上面对个人计算机101所说明的许多或者所有部件。这样的联网环境在广泛用于办公室、企业的计算机内部网和互联网中是随处可见的。

光鼠标115可以基本上与在'882专利或者美国专利No. 6, 256, 016或美国专利No. 6, 172, 354中所揭示的类似。确实,本发明的方法和装置(用一个集成电路实现时)可以并入在上述专利中所看到的结构。与现有技术相比,本发明的独特之处是确定运动向量的方法。在'139专利中,运动向量是通过将后一帧在各个方向位移、对上一帧图像内每个像素计算相关性再用最高相关性确定方向来确定的,然而,正如下面将看到的那样,本发明通过一些块匹配步骤进行。

再来看图2,按照本发明形成的集成电路207包括传感器阵列301、

信号处理器303、输入输出(I/O)305、存储器309和总线307。I/O 305通过电缆209与计算机101连接。最好,这些器件都在单个硅衬底上形成,用标准的CMOS工艺集成在单个芯片上。

传感器阵列301例如可以是实质上类似于由本发明的受让方(OmniVision Technologies, Inc. of Sunnyvale, CA)制造的图像传感器的传感器阵列部分,如型号OV7630、OV7920、OV7930、OV9620、OV9630、OV6910或OV7640。然而,本发明的传感器阵列301通常具有比以上型号低得多的分辨率。因此,传感器阵列301可以只包括几百个或上千个像素而不是几十万个像素。例如,传感器阵列301可以是一个20×20的阵列。

因此,传感器阵列301包括配置成一个二维阵列的多个独立像素。在工作中,在将光鼠标指向任何任意图像时,由一个透镜系统将图像聚焦到传感器阵列301上,使得传感器阵列301可以得到原始图像数据。

信号处理器303通过总线307接收到原始图像数据后,开始进行信号处理。信号处理器303能执行一组实现光鼠标操作所需的预编程指令(这些预编程指令可以是存储在存储器309内的)。作为可应用的,信号处理器303的方案可以从许多提供嵌入式微处理器或DSP内核的公司得到。

在所例示的这个实施例的情况下,信号处理器303被编程成可以获取原始图像数据和将它存储在存储器309内。具体地说,在工作中,信号处理器303将第一图像存入存储器309。随后的第二图像也存入存储器309。随着接收到后来的图像,将这些图像以先进先出(FIFO)方式存储在存储器309内。存储器309应该大到足以存储至少两个相继的图像。在一个实施例中,传感器阵列可以以每秒1500帧的速度进行工作,当然速度慢一些或再快一些也是可以的。因此,两个相继的图像在时间上相隔0.67毫秒。

收集到两个相继的图像后,就可以由信号处理器(在存储器309的配合下)实现本发明的方法,确定这两个图像之间的运动向量。

因此,用可以是一个微处理器或数字信号处理器(DSP)的信号处理器303根据传感器阵列301截获的相继图像来计算光鼠标115的运动。一旦计算出运动情况,信号处理器303就将X和Y运动坐标(称为运

动向量)提供给I/O 305,再由I/O 305通过电缆209将这坐标提供给计算机101。这信息于是通过I/O 305提供给计算机101。计算机101因此能用这信息操纵监视器117上的光标。这样,本发明的光鼠标115就可以用来控制光标。

再来看图3,其中示出了一个 12×12 的原始图像401的例子。这个原始图像401的每个像素都有一个关联的信号幅度。图中例示了原始图像401一部分的像素的信号幅度。

如上所述,本发明采用了块平均匹配方法,从而大大减少了处理器303计算运动向量的计算量。这种块匹配方法使用可调的块规模。假设原始图像数据的规模为 M (水平) $\times N$ (垂直)。在图3所示的原始图像的情况下, $M=12$, $N=12$ 。

对于第一帧的情况,原始图像数据可以在数学上表示为 $F_s(x, y)$,其中: S 为表示块规模的整数, x 从1改变到 M ($1 \leq x \leq M$),而 y 从1改变到 N ($1 \leq y \leq N$)。

类似,对于第二帧的情况,原始图像数据可以在数学上表示为 $C_s(x, y)$,其中: S 为表示块规模的整数, x 从1改变到 M ($1 \leq x \leq M$),而 y 从1改变到 N ($1 \leq y \leq N$)。

正如下面将详细说明的那样,按照本发明,块规模 S 从1开始(向为 N 或 M 的极限)递增1,直到发现一个成功的匹配。

如果块规模 S 增加到2,所得到的数据阵列的规模就为 $M/2 \times N/2$ 。对于第一帧来说,原始图像数据可以在数学上表示为 $F_2(x, y)$,其中: x 从1改变到 $M/2$ ($1 \leq x \leq M/2$),而 y 从1改变到 $N/2$ ($1 \leq y \leq N/2$)。类似,对于第二帧来说,原始图像数据可以在数学上表示为 $C_2(x, y)$,其中 x 从1改变到 $M/2$ ($1 \leq x \leq M/2$),而 y 从1改变到 $N/2$ ($1 \leq y \leq N/2$)。

于是,对图3的原始图像的每个 2×2 的块进行平均,形成一个规模较小的 6×6 的像素矩阵。图4示出了这样一个经平均的像素矩阵,其中块规模为 2×2 。这个矩阵的一些元用从图3得出的数据填入。

如果块规模 S 增加到4,所得到的数据阵列规模就为 $M/4 \times N/4$ 。对于第一帧来说,原始图像数据可以在数学上表示为 $F_4(x, y)$,其中: x 从1改变到 $M/4$ ($1 \leq x \leq M/4$),而 y 从1改变到 $N/4$ ($1 \leq y \leq N/4$)。类似,对于第二帧来说,原始图像数据可以在数学上表示为 $C_4(x, y)$,其中 x 从1改变到 $M/4$ ($1 \leq x \leq M/4$),而 y 从1改变到 $N/4$ ($1 \leq y \leq N/4$)。

通常，本发明的方法有助于确定鼠标的快速运动而不需要大量计算，对于运动大于一个像素距离的情况尤其有用。特别是，用块规模S为1执行运动向量计算。

在一个实施例中，通过对于每个可能方向的位移执行两个相继帧的像素数据之间的相关计算来确定运动向量。通常，有9个可能方向位移：左、右、上、下、上右、上左、下右、下左和没有位移。在一个实施例中，相关计算基于两个所比较的帧内相应像素的信号幅度的简单相减。然后，对于每个方向的位移得出各对像素的信号幅度差之和。因此，对于每组两个相继帧，要进行9个相关计算。

下面将以本发明的一个实施例为例具体说明怎样执行称为帧差和“FDS”的相关计算。概括地说，如下面的伪码所示，对于9个方向中每个方向(表示为 $S(d)$ ，其中d在0-8之间)得出一个相关计算(标为“和(Sum)”)。这个和是在第一帧F与第二帧C内像素之间的差的绝对值之和。然而，对被减的像素在各个方向上作了位移(或者Sum(4)的完全不位移)。

在计算出所有的和后，确定哪个和是最小的。然后，对这个最小的和(MinimumSum)用一个门限值加以检验。这个最小和，或者说最小的相关计算值，提供了运动方向(运动向量)。

此外，按照本发明，只有在MinimumSum小于门限值时才将这个运动向量确定为有效，予以输出。如果MinimumSum不小于门限值，就认为导致得出这个运动向量的相关计算已经失败。此时，如下面详细示出的那样，通过对 $S \times S$ 数据块进行平均压缩原始图像数据，形成一个如图4所示的经压缩的矩阵。对这个经压缩的矩阵执行同样的相关计算。这个过程一直继续到MinimumSum低于门限。注意，对于像素不多的情况应该降低门限值，予以补偿。作为一个例子，对于 2×2 数据块平均，门限值可以除以4。

```
/*Nine - direction absolute magnitude frame difference sum
and get the minimum FDS (9个方向绝对值帧差和以及获取最小的
FDS) * / FDS ( )
(
  initial Sum(n)
  for (x=2; x<M; x++)
```

```

for (y=2; y<N; y++)
{
    Sum(0)=Sum(0)+|C(x,y)-F(x-1,y-1)|; //移至上左
    Sum(1)=Sum(1)+|C(x,y)-F(x,y-1)|; //上移
    Sum(2)=Sum(2)+|C(x,y)-F(x+1,y-1)|; //移至上右
    Sum(3)=Sum(3)+|C(x,y)-F(x-1,y)|; //左移
    Sum(4)=Sum(4)+|C(x,y)-F(x,y)|; //不移
    Sum(5)=Sum(5)+|C(x,y)-F(x+1,y)|; //右移
    Sum(6)=Sum(6)+|C(x,y)-F(x-1,y+1)|; //移至下左
    Sum(7)=Sum(7)+|C(x,y)-F(x,y+1)|; //下移
    Sum(8)=Sum(8)+|C(x,y)-F(x+1,y+1)|; //移至下右
}
}
(I, Minimum)=getMinimunSum(Sum); //获取最小和以及移动指数
If Miniumnsum <Threshold
    return(i); // i为0-8, 表示移动向量
else
    return(failed);
}

```

因此, 如果按块规模S相关计算得到一个无效的结果, 就递增块规模S(用各种可能的递增策略中的一种), 直到得到一个有效的相关计算。一旦认同了这个可接受的相关计算值, 就可以确定运动向量, 运动向量的幅度取决于得到可接受的相关计算值的块规模S。

下面是实现上面所说明的一般方法的伪码。注意, FDS为相关计算值。

```

GetcurrentFrame(C1);

if FDS(C1, F1, M, N)=failed then
    if FDS(C2, F2, W2, N/2)=failed then
        if FDS(C4, F4, M/4, N/4)=failed then
            ...

```

```
        Else
            Report movement vector-4 //以4x的速度移动
            Update F1 =C1
        endif
    else
        Report movement vector-2 //以2x的速度移动
        Update F1=C1
    endif
else
    Report movement vector-1 //以1x的速度移动
    Update F1=C1;
endif
```

业已发现上述方法对于鼠标的快速运动是有益的。需执行的计算量少于任何已知的现有技术的计算量。

虽然在这里以有限的实施例为背景对本发明进行说明，但本发明可以根据本发明的精神实质用许多方式实现。因此，所例示和说明的实施例就各种情况来说都是例示性的而不是限制性的。例如，在上述说明中，相关计算是简单的在像素的信号幅度之间的差的绝对值。在其他实施例中，相关计算可以是相继的帧之间的差别的许多数学度量中的任何一种度量。此外，虽然通过对像素取平均对像素数据块进行压缩，但是也可以用其他数学方法将原始图像数据压缩成经压缩的版本。

因此，虽然本发明是以在光鼠标中的应用为例进行说明的，但是本发明可以用于许多不同的应用，包括但并不局限于计算机视觉。本发明的专利保护范围由所附权利要求书而不是由上述说明给定，所有在与权利要求书给出的意义相当的范围内的改变都应视为属于本发明的专利保护范围。

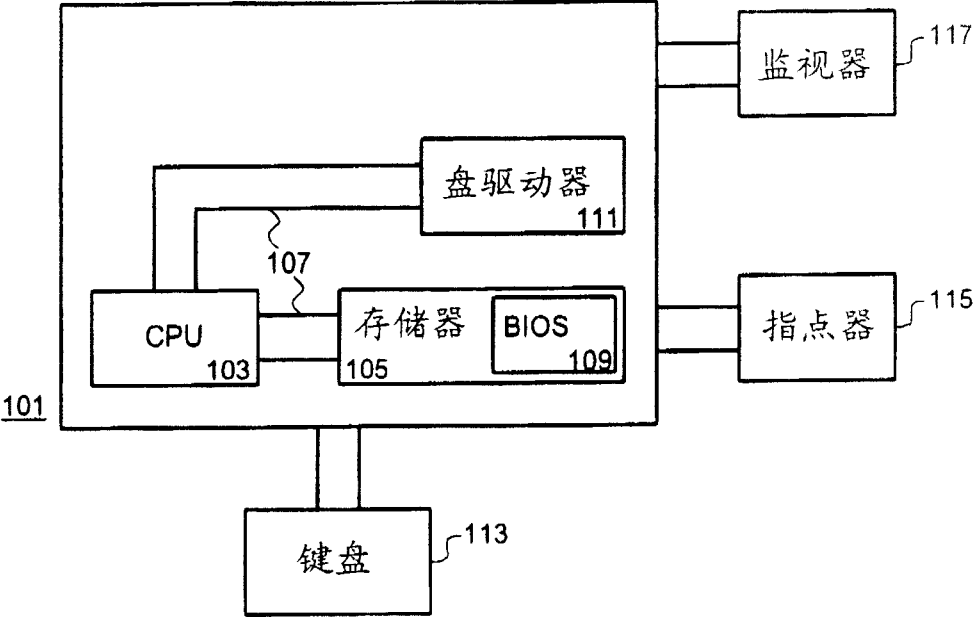


图 1

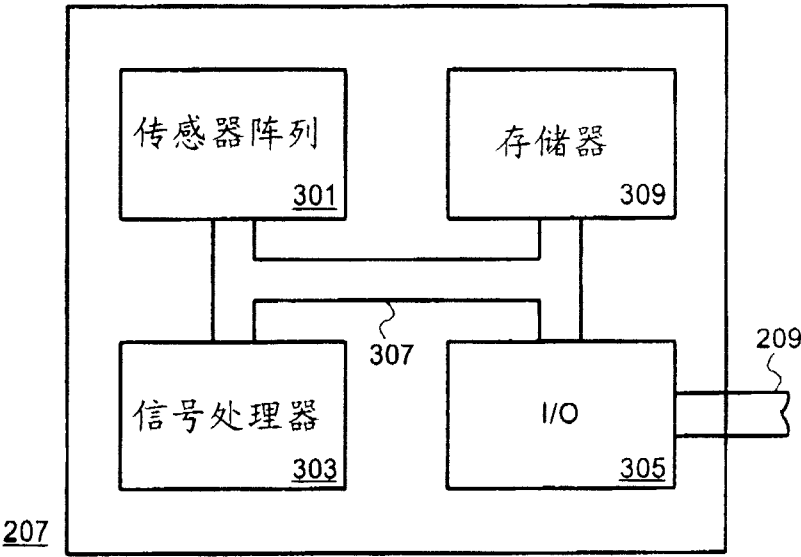


图 2

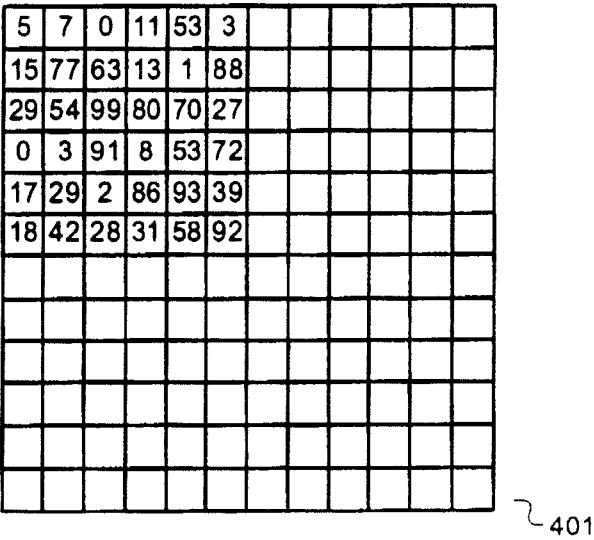


图 3

26	21.7	38.7			
21.5	69.5	55.5			
26.5	36.7	65.5			

图 4