

# 安全录像系统中动态感知算法的研究与实现

彭 波 胡治国

(中国农业大学电气信息学院)

**摘 要** 在安全录像系统中设计了动态感知算法。采用该算法,系统只记录那些发生了变化的有意义的图像,从而节约了大量存储资源。实际应用结果表明,算法效率较高,可以满足实时性要求;算法具有良好的交互性和通用性,可以应用于其他类似系统中。

**关键词** 动态感知; 算法; 安保系统

**中图分类号** TP 301.6; TP 311

## Study and Implementation of Dynamic Perception Algorithm in Security Recorder System

Peng Bo, Hu Zhiguo

(College of Electricity and Information, China Agricultural University, Beijing 100083, China)

**Abstract** The dynamic perception algorithm was designed for a security recorder system. Including the algorithm, the system can record only those pictures that are quite different from the previous ones. In this way, a great deal of storage space is saved. At the same time, any pictures that make sense are not missed. The algorithm is based upon a lot of scientific experiments, which makes it credible. It is efficient enough to give a real time result. With its friendly interactivity, it can also be used to many other applications where only motion pictures are noticed.

**Key words** dynamic perception; algorithm; security recorder system

近年来,安全录像系统已经广泛应用于银行、酒店、仓储和智能小区等对安全保障要求较高的场所。目前广泛使用的录像系统一个很大的缺点就是,每天录像资料的数据量太大,而且大部分时间的画面是静态的,造成了非常大的数据冗余。因此,构造一种算法,使系统能够快速判断画面是否发生了变化,并录下那些具有明显运动的画面,具有现实意义。

### 1 算法的总体思路

算法的指导思想是,比较前后 2 帧图像之间的差异,如果差异达到了一定程度,就认为图像有了变化,从而启动系统开始录像。要实现算法,必须明确 2 方面的问题:如何定义两幅图像的差异;差别达到了什么程度才认为有明显变化。下面分别说明。

1) 将 2 幅图像的差异定义为图像中变化了的像素个数占参加比较的像素总数的比例。数字图像是由一个个像素组成的,表征各个像素的量的集合构成一幅图像的数据。YUV<sup>[1]</sup>格式

收稿日期: 2002-03-07

彭 波,北京清华东路 17 号 中国农业大学(东校区)142 信箱, 100083

是一种通用的数字图像表示形式,所设计的录像系统中图像采用 YUV 4 2 2 中的 YUV 2 格式表示(对于其他非 YUV 格式的系统,可以先转化为该格式后再进行处理)。在这种格式下,像素用亮度分量 Y 和色度分量 U, V 来表示(图 1)。在每个扫描行中,相邻的 2 个像素共用 1 组色度分量 U 和 V,同时每个像素单独有 1 个亮度分量 Y。每个分量占用 1 个字节,因此每 2 个像素占用 4 个字节。

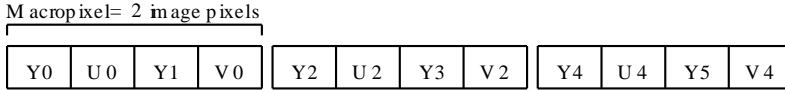


图 1 YUV2 格式下像素的表示

由于人眼对色度的分辨能力比对亮度要低得多<sup>[2]</sup>,因此图像的大部分信息都集中在亮度分量 Y 上,基于此,算法只选取了亮度分量来表达像素信息。当 2 幅图像中相同位置像素的亮度差异达到一定程度时,就表示这个像素发生了变化。用户通常只对画面的一部分(安全敏感区,如门、窗所在的位置)感兴趣,可将这些区域设定为动态感知区,1 个画面上可以设置多个动态感知区。只有当这些区域的画面有显著变化时,才会启动录像,其他区域的数据并不参与比较。统计出设定区域中变化了的像素个数,然后计算出它们占整个设定区域中像素总数的比例,就得到了 2 幅图像的差异。

2) 只有在 2 幅图像的差异达到阈值时,才认为有了明显的运动变化。由于各种因素的影响,2 幅在人眼看来没有变化的图像,数字化后数据也会有一定的差异,把这样 2 幅“没有变化”的图像在数据上的差异,定义为图像的噪声。必须排除噪声的影响,才能做出正确判断。根据各种情况下图像的最大噪声,结合用户设定的灵敏度,获得判断的阈值。

## 2 试验及数据整理

为保证系统的通用性,使其在各种录像环境下都不至于过于灵敏,导致录下不必要的画面,必须得到各种情况下图像噪声的最大值,将它屏蔽掉。设计试验的目的正是给出各种情况下噪声的最大值,作为选择阈值的根据。

### 2.1 影响图像噪声的因素

分析表明,可能对噪声造成影响的因素主要有如下 5 个:

1) 环境的亮度; 2) 被摄物体的属性; 3) 摄像头(包括镜头)的质量; 4) 采集卡的质量; 5) 录像速度, 帧·s<sup>-1</sup>。

### 2.2 试验方案的设计

鉴于噪声是上述 5 个因素综合作用的结果,对每个因素分别设计了如下实验方案。

1) 用亮度可调的灯光作为照明物,用照度计对亮度进行精确控制。由于系统运行在可见环境中,因此只需要对可见的环境照度进行测量。一般来说,环境照度最暗时对应的照度值为 60 lx,最亮时为 600 lx。为了将这一亮度范围分成合适的等级,正式试验前,针对亮度变化对噪声的影响做了一系列的预试验。试验结果表明将这一亮度范围分成 10 个等级,每级相隔 60 lx,既可以避免由于分级太细导致亮度变化对噪声的影响不明显,也可以避免由于分级太粗而无法准确反映噪声的变化规律。

2) 取 3 张纸,分别为白、灰、黑色作为典型的被摄物体。

3) 用 3 种常用的典型摄像头分别摄取画面。

4) 用 2 个采集卡同时采集画面。

5) 录像速度按由小到大顺序, 在  $1 \sim 25 \text{ 帧} \cdot \text{s}^{-1}$  之间选取  $1, 2, \dots, 25 \text{ 帧} \cdot \text{s}^{-1}$  共 25 种速度, 分别进行测试。

上述 5 个因素可以看做 5 个变量, 每个变量的取值个数依次为 10, 3, 3, 2 和 25, 这样就有 4 500 种 ( $10 \times 3 \times 3 \times 2 \times 25$ ) 组合情况。对每一种情况, 间隔一定帧数录下 2 幅“相同”图像, 这样就可以计算得到各种情况下的噪声数据。这里的“一定帧数”在数值上等于系统设定的录像速度, 这样每秒有且仅有 1 帧图像参加比较, 从而避免了由于比较太频繁而导致的计算量过大的问题, 同时抓住了每秒画面的主要信息, 不致漏掉有用数据。

### 2.3 试验结果处理

图像的亮度用 1 个字节 (8 bit) 进行编码, 其表示的灰度值范围是  $[0, 255]$ 。考虑到问题的实质以及保证尽量小的计算量, 将像素的亮度值映射为“0”或“1”, 如式 (1)<sup>[3]</sup>:

$$Y = \begin{cases} 0 & 0 \leq Y < 127 \\ 1 & 128 \leq Y < 255 \end{cases} \quad (1)$$

如果 2 幅图像中某对应位置的像素由“0”变成了“1”, 或由“1”变成了“0”, 则表示像素发生了变化。将前后 2 幅“相同”图像中发生了变化的像素的个数统计出来, 再除以图像的像素总数, 就得到了图像的噪声。

将所有计算得到的噪声数据制成表格, 可以发现如下规律:

- 1) 在其他因素不变的情况下, 亮度的改变会引起噪声规律性的变化;
- 2) 在其他因素不变的情况下, 画面属性不同, 噪声有较大不同;
- 3) 在其他因素不变的情况下, 摄像头的改变对噪声影响很小;
- 4) 在其他因素不变的情况下, 采集卡的改变对噪声几乎没有影响;
- 5) 在其他因素不变的情况下, 录像速度的改变对噪声影响也很小。

由此可见, 亮度对噪声具有决定作用, 图像的属性变化对噪声有较大影响, 摄像头、采集卡和录像速度对噪声的影响很小, 相对于前 2 个因素可以忽略不计。根据以上试验结论, 结合系统的应用实际, 采用如下方法选取噪声的最大值: 对于每种亮度 (环境照度, lx), 在 3 种典型图像的噪声中, 选取 1 个最大值, 作为各种图像在该亮度下可能产生的最大噪声。由于所选取的 3 幅图像代表了算法所关心的灰度属性, 具有代表性, 所有其他图像产生的噪声都不会超过它们得到的最大噪声, 因此这样处理是科学的, 实际的测试也证实了这一点。经过多次试验, 并进行综合分析之后, 得到环境照度与最大噪声的对应关系 (试验值) 见表 1。

表 1 环境照度与最大噪声对应表

照度/lx	60	120	180	240	300	360	420	480	540	600
最大噪声/%	2	2	13	9	16	19	21	8	10	9

在系统运行过程中, 无法自动得到环境的客观照度值, 因此要使用试验结论还必须做一下转化。当环境亮度较高时, 对应画面较亮, 表现在数据上就是像素的亮度分量  $Y$  较大, 所以可以得到亮度与像素灰度值的对应关系。在 10 个环境照度等级下分别摄取 1 幅白纸的图像 (目的是排除前景物体对灰度的影响), 求出所有像素点的平均灰度值, 就得到了环境照度与像素

灰度值之间的对应关系(表2)。

表2 环境照度与像素灰度值对应关系

照度/lx	60	120	180	240	300	360	420	480	540	600
灰度值	210	213	216	218	222	225	228	230	234	238

在系统运行中,取每组参加比较的第1幅图像的白色背景区域,计算出区域中像素的平均灰度值,就可以查表1和2得到此时的最大噪声值。

由噪声的最大值,根据式(2),可计算出判断所需的阈值。式(2)为

$$L = B + N \times (100 - S) / 100 \quad (2)$$

其中: $L$ 为阈值。 $B$ 为阈值的基值,保证 $L > 0$ 恒成立,可以避免系统过于灵敏,以致录下没有实际意义的画面,该值通过试验值确定。 $N$ 为当前亮度下图像的最大噪声。 $S$ 为用户设定的灵敏度值,范围为0~100;灵敏度值越大,阈值越小,表示系统对画面变化越敏感;因为噪声最大值是在所有情况下的综合结果,为了适合每一种具体的应用环境,必须由用户根据应用要求和主观经验对该值进行修正,这就是灵敏度调节的原因。

### 3 算法的实现

以Visual C++ 6.0<sup>[4]</sup>为工具,开发出了一个静态库,给出了算法的程序实现。该静态库有4个类,其中2个用于与主界面程序进行交互,1个用于在图像画面中设置动态感知区域,另一个类是处理数据的核心类。这里仅对核心类进行介绍,其主要处理过程如下。

从数字图像的YUV分量中提取出亮度分量Y。为减少数据计算量,不对每个像素采样,而是每隔4个像素取1个像素点的Y值。将每个动态感知区域的长宽(以像素为单位)先扩展为8的倍数,然后将该区域划分成多个 $8 \times 8$ 像素的像素块<sup>[5]</sup>,每1个小块包括8个扫描行的数据,每行中包括8个像素点。

将这些按块排列的亮度Y值依式(1)映射为“0”或“1”。

经过以上的处理之后,对2幅图像感知区域中的数据进行比较,即将对应位置的数据,做“与”运算,得到发生了变化的像素个数。

根据下面描述的三级收敛策略进行判断,给出画面是否有明显变化的结论。

由于数字图像的数据量很大,要实时地得到结论,就必须提高比较速度。算法采用三级收敛策略,保证了算法的快捷性。

#### 1) 第一级:在像素块一层判断

如果2幅图像的某一小块里对应位置变化了的像素个数占这个块像素总数的百分数达到一定的值(这个值的取值很大,比如80%)时,就断定图像发生了运动变化,此后,就不必再对其他块的数据进行比较了,从而节约了计算时间。原理是,如果一个小块里的绝大部分像素都发生了变化,就不会是噪声引起的,而是实际上确实发生了变化。

#### 2) 第二级:在块一层判断

如果在已比较的块中,有若干块中像素的变化都超过了一定的值(比第一级取值小,如60%),并且占了已经比较的小块总数的一定比例(如40%)时,就判定发生了运动变化。如果在这一层能够断定发生了显著变化,也不必比较后面剩下的数据块。原理是,如果在比较完的

数据块中, 有比较多的数据块都发生了相当大的改变, 只能是由画面的实际变化造成的。

### 3) 第三级: 在所选区域一层判断

当前两级都没有做出“有明显变化”的判断时, 采用这一级来判定。这一级将所选区域中的所有像素块的数据都进行比较后, 再做出判断。当整个区域中发生了变化的像素个数占总像素个数的百分比达到了前面得到的阈值时, 就判定发生了运动变化。

对多个区域分别判断, 如果有 1 个发生了显著变化, 就录下该画面; 如果所有区域都没有发生显著变化, 则无需录像。

## 4 结束语

本算法已经在一个实际的安全录像系统中得到了运用。测试结果表明, 在系统的所有 16 个通道都启用动态感知录像方式的情况下, CPU 占用率只增加了 10%~ 15%, 其录像效果与客户期望十分接近。

以下 2 个改进措施可以进一步优化算法:

- 1) 在判断画面是否发生变化的过程中, 考虑色度因素, 使结果更精确;
- 2) 对感知区域先进行统计意义上的粗判断, 在没有变化的情况下, 不再进入三级判断过程, 从而进一步加快速度。

## 参 考 文 献

- 1 钟玉琢 多媒体技术 北京: 清华大学出版社, 1999 73~ 86
- 2 黄孝建 多媒体技术 北京: 北京邮电大学出版社, 2000 9~ 20
- 3 容观澳 计算机图像处理 北京: 清华大学出版社, 2000 114~ 116
- 4 Microsoft MSDN Library Visual Studio 6.0 版 <http://www.microsoft.com/msdn> 1999
- 5 马小虎 多媒体数据压缩标准及实现 北京: 清华大学出版社, 1996 34~ 43