

# 多目标航迹显示数据的动态实时压缩处理

汤建华

(中国科学院长春光学精密机械研究所, 长春 130022)

**摘要** 针对多目标航迹显示的特殊性, 提出了一种动态实时数据压缩处理方法。该方法实时性好, 节省内存, 目标航迹显示恢复简捷不失真。

**关键词:** 数据压缩; 多目标; 实时处理; 航迹显示

## 1 引言

在现代跟踪测量系统中, 控制中心需实时显示目标的航迹信息, 掌握目标的运动情况, 制定必要的控制策略。目标航迹显示数据主要包括目标的编号, 距离、方位角、俯仰角等。

对于简单的目标航迹显示, 只要将目标的运动数据实时转化成显示屏幕上的  $x, y$  坐标, 用绘图函数画出相应的航迹点即可, 无需记录航迹数据, 也自然不必进行数据压缩。如果屏幕要进行多画面的显示切换(如变距离的多目标态势图的显示切换)或事后分析目标运动情况, 就需要实时记录目标的航迹显示数据。假设目标航迹显示数据中编号占 1 字节内存, 方位角、俯仰角和距离各占两个字节, 其它信息占 1 字节(目标性质、显示颜色, ……), 则目标航迹一点显示数据需要 8 字节内存, 设测控系统的数据通讯频率为 20 Hz, 则相邻航迹点间的采样周期为 50 ms, 一个目标一秒钟需要 160 字节的内存空间用来记录航迹数据, 20 分钟需 19 k 字节 ( $160 \times 1200$ ), 这样 10 个目标 20 分钟 190 k 内存来记录航迹数据。显然随目标运动时间的增长, 系统通讯频率的提高, 用来记录目标航迹数据的内存也越来越大, 不对这样的数据进行实时动态压缩将使记录的内存会不够用, 同时记录的航迹点太多, 也不利于画面切换时的航迹快速恢复。为此本文结合目标航迹显示的特殊性, 提出了一种动态实时数据压缩处理方法, 即折半压缩法, 该方法不对数据进行变换处理, 实时性好, 航迹恢复简捷不失真, 记录内存固定。

## 2 折半压缩

### 2.1 多目标航迹显示数据压缩的可能性

多目标航迹显示数据相邻采样点间的数据是相似的, 数据间的相关性很大, 可进行压缩; 计算机显示屏幕的分辨率有限, 使得相邻的一些航迹显示数据可能转化成屏幕上的同一点, 航迹显示数据相对于屏幕的分辨率来说有很大的冗余性, 可对这样冗余的数据进行压缩, 而不影

响航迹显示效果;一般的航迹显示比较关心近期的运动轨迹,选择必要的控制策略,因此可按时间关系,多保留当前数据,而少保留以前的数据。

## 2.2 折半压缩法

根据显示屏分辨率选择一固定长度的数据结构来记录航迹数据,例如分辨率为  $640 \times 480$  时长度可选 256,即最多可记录连续的 256 个航迹点,当目标航迹点超过 256 时,采用等间隔抽取数据的方法进行折半压缩处理,即保留航迹中的奇数点,删除偶数点,这样使显示数据压缩一半,而采样周期变成原来的 2 倍。压缩后的数据存入记录结构的前 128 个连续单元里,后 128 个单元用来记录新的航迹点,新的航迹点依次存入第 129, 130, ..., 256 单元,当数据记满时,再进行折半压缩处理,这样反复折半压缩下去,直到目标消失,测控结束为止。当记录长度为 16 时折半压缩法如图 1 示意,其中连续数字表示航迹点的出现顺序,①,②,③,④,⑤为经四次折半压缩后形成的采样周期不等的五个区域。

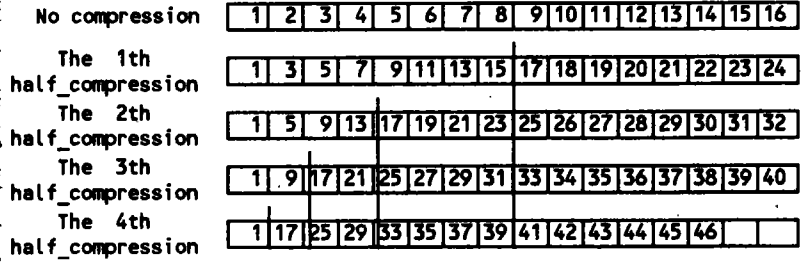


图 1 目标航迹显示数据折半压缩示意

Fig. 1 An example for object track video data half compression

## 2.3 折半压缩的特点

设航迹点记录长度  $N=2^n$ ;折半压缩次数计数器为  $b$ ;系统航迹点实时采样周期为  $t$ ,随着折半压缩次数的增加,相邻航迹点间的采样周期增大,采样周期相等的一组航迹点组成一个区域,设  $k$  为区域编号,  $T[k]$  为  $k$  区域的采样周期,则航迹显示数据经多次折半压缩有如下特点:

- 初始点永远保留,不被压缩;从初始点到当前点采样间隔由疏到密;航迹显示记录不受内存和时间的限制;

- 折半压缩后航迹区域划分如图 1 和图 2 示意,  $N$  点一次折半压缩后分裂成等长的①区和②区;  $N$  点二次折半压缩后②区分裂成等长的②区和③区;  $N$  点  $k$  次折半压缩后  $k$  区分裂成等长的  $k$  区和  $k+1$  区;  $n$  次压缩后区域划分结束;

- 区域采样周期  $T[k]$  的计算:

$$\text{当 } b \leq n \text{ 时, } T[k] = t \cdot 2^{k-1} \quad k \in [1, b+1]$$

$$\text{当 } b > n \text{ 时, } \begin{cases} T[k] = t \cdot 2^{k-1} & k \in [1, n] \\ T[n+1] = t \cdot 2^b & \end{cases}$$

相邻区域界面两侧航迹点间的采样周期等于两区域采样周期的极大值;

- 相邻区域的采样周期及区域所含的航迹点数均成等比数列,公比为 2,即:

$$T[1] : T[2] : \dots : T[k] : \dots : T[n] = 1 : 2 : \dots : 2^{k-1} : \dots : 2^{n-1}$$

$$\text{①} : \text{②} : \dots : k : \dots : h = 2^{n-1} : 2^{n-2} : \dots : 2^{n-k} : \dots : 1$$

• 第  $n+1$  区只记录初始航迹点,该点到压缩后的最近点间的采样周期随着折半压缩次数  $b$  的增加而成倍增长,为  $t \cdot 2^b$ ;

• 区域  $k$  相邻航迹点间压缩的点数为  $T[k]/t-1$

### 3 多目标航迹数据压缩与恢复算法

#### 3.1 航迹显示数据结构

设目标数量上限为  $M$ , 单点航迹显示数据需 8 字节内存记录, 记录数据长度为  $N$ , 采用  $N \times M \times 8$  的三维数组  $\text{line-buf}[N][M][8]$  作为航迹显示数据结构, 当  $M=10, N=256$  时, 需要  $20k$  内存空间。设多目标航迹单点数据存放在  $\text{obj}[M][8]$  中, 用数组  $I[M]$  记录  $M$  个目标当前航迹点在  $\text{line-buf}$  中的记录位置, 初始化时置为 0。

#### 3.2 多目标航迹显示数据折半压缩算法

```
for(j=1; j ≤ M 或 j ≤ 实际目标数; j++)
  if(I[j] < N) (第 j 个目标单点数据记录)
    { for(k=0; k < 8; k++)
      line-buf[I[j]][j][k] = obj[j][k];
      I[j] = I[j] + 1; }
  else (第 j 个目标数据折半压缩)
    { for(i=0; i < N; i=i+2)
      for(k=0; k < 8; k++)
        line-buf[i/2][j][k] = line-buf[i][j][k];
      I[j] = N/2; }
```

#### 3.3 多目标航迹显示恢复算法

```
for(j=1; j ≤ 实际目标数; j++)
  for(i=0; i ≤ I[j]; i++)
    { 利用第 j 个目标压缩后的第 i 个航迹点数据 line-buf[i][j][?] 计算该点的屏幕座标 (x,y)(算法略); 航迹显示恢复处理如下:
      方式 1: 画出第 j 个目标的第 i 个航迹点;
      方式 2: 连接第 j 个目标的第 i-1 点和第 i 点画直线;
      方式 3: 在第 j 个目标的第 i-1 点和第 i 点间进行插值显示恢复; }
    在第 I[j] 点的屏幕位置画出第 j 个目标的图符;
```

航迹恢复处理可选上述三种方式之一, 其中方式 1 实时性最好, 显示的是真实的航迹点; 方式 2 次之, 在压缩的航迹点间进行了简单的平滑处理, 使航迹的显示效果更好; 方式 3 需根据压缩次数  $b$  确定第  $i-1$  点所在的区域  $k$  及相应的采样周期  $T[k]$ , 则第  $i-1$  点和第  $i$  点间压缩掉了  $T[k]/t-1$  个数据, 可在这两点间进行线性插值, 恢复显示出压缩掉的航迹点。

### 4 实时性处理

多目标航迹显示数据的动态实时处理中, 压缩算法只进行数据移动操作, 时间为微秒级,

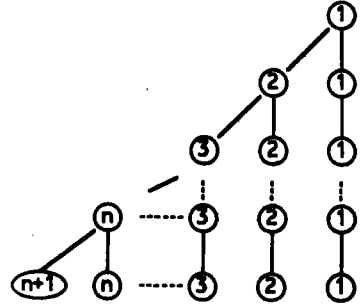


图 2 折半压缩区域分裂示意  
Fig. 2 An example for half compression area division

无需做进一步的实时性处理,而航迹恢复算法中,由于单目标航迹的绘制时间为毫秒级,当多个目标航迹同时恢复时,时间很长,为此需做相应的实时处理,用系统的处理频率 20 Hz 分时操作,每次仅恢复一个目标的航迹信息,实时性处理流程如图 3 所示,图中  $m$  为实际目标数,  $a$  为目标恢复计数器,  $j$  为目标处理序号,初始化时令  $a=1; j=1$ 。

### 5 结 论

实践表明上述方法是有效的。根据每次航迹显示数据压缩与保存数量的不同,折半压缩算法可做相应的改变,例如将记录长度等分成三份,每次将前两份的航迹点进行折半压缩处理,可使数据压缩三分之一,保存三分之二。

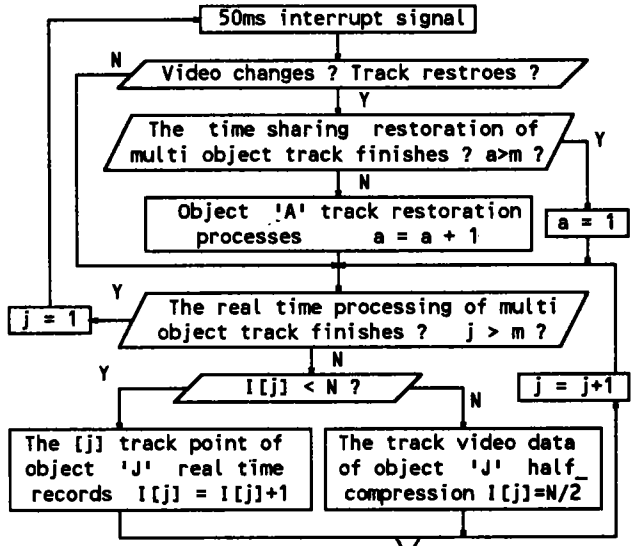


图 3 多目标航迹显示数据实时处理  
Fig. 3 Multi object track video real time processing

### 参 考 文 献

[1] [美]D. E. 克努特, 计算机程序设计技巧. 第三卷分类与检索, 北京: 国防工业出版社, 1984

## Dynamic Real-time Compression of Multi-object Track Videographic data

Tang Jianhua

(Changchun Institute of Optics and Fine Mechanics, Chinese Academy of Sciences, Changchun, 130022)

### Abstract

In this paper we introduce a dynamic real-time data compression method for multi-object track videographic display. This method processes fast with a few memory. It can simply restore object track videographic display without distortion.

**Key words:** Data compression, Multi-object, Real-time processing, Track videographic display

汤建华 助研, 工学硕士, 1989年毕业于哈尔滨工业大学模式识别与智能控制专业, 现从事计算机控制, 工程软件总体设计工作。