

交叉运动多目标电视跟踪算法

朱 进

(中国科学院长春光学精密机械研究所, 长春 130021)

摘要 本文提出了一个对交叉运动多目标的电视跟踪算法。包括跟踪策略, 多目标相互重合和离分时刻判定、以最小二乘法为基础, 对目标运动方向的计算、实际目标与视场中所见目标的对应关系的确认。

关键词: 交叉运动; 多目标; 电视跟踪; 最小二乘法

1 引 言

经纬仪加装电视跟踪可以实时输出视场中目标信息, 包括重心、前沿、后沿、像元数、灰度等等, 并对运动目标进行自动跟踪。以往的电视经纬仪都是对单一目标进行跟踪, 即视场中只有一个目标。多个目标在视场中以不同方向, 不同速度运动, 并相互之间有交叉的情况下, 要对每一个目标进行准确地跟踪和测量, 是电视跟踪测量领域中一个新的尝试, 我们结合跟瞄系统的研制, 对这一问题进行了深入的研究。

2 基本跟踪策略以及目标交叉和分离时刻的判定

以往单目标或多目标但目标间无交叉运动的跟踪策略, 可以简单地通过计算相邻两场之间重心距离的方法来进行。设定一个阈值, 若相邻两场间目标重心距离小于该阈值则判定此两场中两目标具有对应跟踪关系。在多目标运动交叉的情况下, 由于目标的大小不定, 目标交叉和分离时重心变化很大。阈值太小会丢失目标, 太大则会产生混乱。我们采用目标间相对最小边缘距离来代替重心距离, 在实验中取得

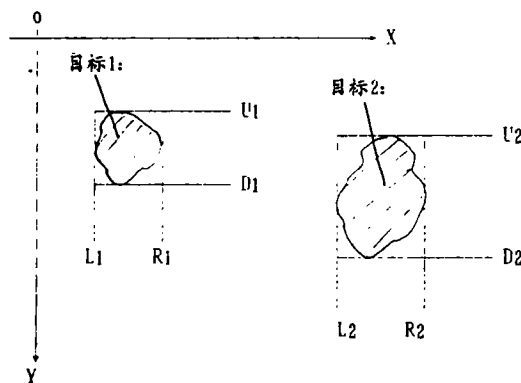


图 1

了较好的跟踪效果。

所谓相对最小边缘距离 B_d ，由以下过程得出：

设： U_1, U_2 分别为目标 1、目标 2 的上沿；

D_1, D_2下沿；

L_1, L_2前沿；

R_1, R_2后沿

如图 1 所示。

- (1) 若 $U_2 \leq U_1 \leq D_2$ 或 $U_2 \leq D_1 \leq D_2$
或 $U_1 \leq U_2 \leq D_1$ 或 $U_1 \leq D_2 \leq D_1$

则 B_d 定义为 $\min \{ |L_2 - R_1|, |L_1 - R_2| \}$

- (2) 若 $L_2 \leq L_1 \leq R_1$ 或 $L_2 \leq R_1 \leq R_2$
或 $L_1 \leq L_2 \leq R_2$ 或 $L_1 \leq R_2 \leq R_1$

则 B_d 定义为 $\min \{ |U_1 - D_2|, |U_2 - D_1| \}$

- (3) 若①和②的前提同成立则 B_d 定义为 0。

- (4) 若①和②的前提都不成立则：

B_d 定义为 $\min \{ |L_2 - R_1|, |L_1 - R_2|, |U_1 - D_2|, |U_2 - D_1| \}$

确定了最小边缘距离后,就可以确定一个阈值 T_n (根据目标运动速度确定),以和重心跟踪类似的方法判断两场间目标对应关系,并判断目标交叉和分离的时刻。若在 S 场中有目标 O_1, O_2, \dots, O_n , 在 $S + 1$ 场中目标 J , 使得, $O_k = (K = 1 \dots n)$ 与 J 都存在前面提到的对应关系,就判定目标 O_1, \dots, O_n 在 $S + 1$ 场时刻合并为目标 J 。若在 S 场中有目标 J , 在 $S + 1$ 场中有目标 O_1, \dots, O_n , 使得, J 与 $O_k (K = 1 \dots n)$ 都存在对应关系,就判定 J 目标在 $S + 1$ 场时刻分离为 O_1, \dots, O_n 。

3 目标运动方向的计算, 以及合并前与分离后目标的编批

在目标相互交叉合并又相互分离的过程中, 如何确定合并和分离后目标的对应关系, 是另一个要解决的问题。如果我们限定视场中最多只能有两个目标, 这个问题, 就容易得多。只须将分离后的两个目标以先上后下, 先左后右做一位置排序, 然后与合并前两个目标的位置顺序做一颠倒, 即: 合并前相对位置在上方的目标, 对应分离后在下方的目标, 合并前相对位置在左方的目标, 对应分离后右方的目标。但如果视场中有多于二个目标的情况, 并且三个以上目标同时交叉运动时这种方法就不适用。

我们以目标在视场中相对运动方向为主要判据, 进行合并前与分离后目标的编批。对视场中每个目标增加一项特征描述叫运动方向, 如果目标不动就设其方向为 -1, 如果目标有相对运动, 则其运动方向由下面的方法计算出。

从理论上讲, 通过相邻两场的目标重心变化, 就可以得到目标运动方向。如: 在 S 场时, 目标重心在 X_1, Y_1 , 在 $S + 1$ 场时目标重心在 X_2, Y_2 , 则目标运动方向就可以用 $(Y_2 - Y_1) / (X_2 - X_1)$ 确认。但是在实践中这种方法误差很大。经常出现目标编批错误, 难以满足要求。我们采用最小二乘法, 通过 $S - 4, S - 3, S - 2, S - 1$, 和 S 场 5 场重心位置来确定在 S 场中的目标的运动方向。

最小二乘法是一个解决如何通过处理观测结果来确定变量之间函数关系的方法,即要求选取函数 $f(X_i)$ 中的参数,使得观测值 Y_i 与对应的函数值 $f(X_i)$ 的偏差的平方和为最小。

现在我们从概率论的观点来说明最小二乘法的理论根据。假定变量 X 与 Y 之间的函数关系可以精确地表为解析式

$$Y = f(X) \quad (1)$$

实验数据 (X_i, Y_i) 由于不可避免的随机误差,而与这个解析式发生一些偏差。该偏差 ζ_i ($i=1, 2, \dots, n$) 通常是服从正态分布的随机变量,并且具有数学期望为 0, 均方差为 σ_i , 如果各次试验中测量的精确度是相同的, 则:

$$\sigma_1 = \sigma_2 = \dots = \sigma \quad (2)$$

于是 ζ_i 的分布密度可以表示为:

$$\Phi_i(Z) = \frac{1}{\sqrt{2\pi} \cdot \sigma} \exp\left(-\frac{Z^2}{2 \cdot \sigma^2}\right) \quad i = 1, 2, 3, \dots, n \quad (3)$$

在 n 次独立实验的结果中发生下列事件:

$$\zeta_i = Z_i = Y_i - f(X_i), \quad i = 1, 2, \dots, n, \quad (4)$$

现在的问题是选择 $f(X)$ 才能使上述事件的积的概率最大?

严格的说,因为偏差 ζ_i 是连续随机变量,所以上述任一事件的概率等于 0, 因此代替事件 $\zeta_i = Z_i$, 我们考虑事件 A_i :

$$Z_i \leq A_i \leq Z_i + \Delta Z_i, \quad i = 1, 2, \dots, n \quad (5)$$

我们有 $P(A_i) \approx \Phi_i(Z_i) \Delta Z_i$, $i=1, 2, \dots, n$

因为事件 A_1, A_2, \dots, A_n 是独立的, 所以有:

$$\begin{aligned} P(A_1, A_2, \dots, A_n) &= \prod_{i=1}^n P(A_i) \approx \prod_{i=1}^n \Phi_i(Z_i) \Delta Z_i \\ &= \left(\frac{1}{\sqrt{2\pi} \cdot \sigma}\right)^n \exp\left[-\frac{1}{2 \cdot \sigma^2} \cdot \sum_{i=1}^n \Delta Z_i^2\right] \\ &= C \cdot \exp\left[-\frac{1}{2 \cdot \sigma^2} \sum_{i=1}^n (Y_i - f(X_i))^2\right] \end{aligned} \quad (6)$$

其中 C 是不依赖于 $f(X_i)$ 的常数,

显然,为了使概率 $P(A_1, A_2, \dots, A_n)$ 为最大,应使和式

$$S = \sum_{i=1}^n [Y_i - f(X_i)]^2 \text{ 为最小 (上述推导详见}^{[3]} \text{)}$$

用最小二乘法来确定函数 $Y=f(X)$ 中的未知参数。我们只用到 Y 为线性函数的情况。因此 $Y=a+bX$ 。这时偏差的平方和 $S = \sum (Y_i - a - bX_i)^2$ 。为了使 S 取最小值,我们把上式分别对 a 及 b , 求偏导数。并让它们等于 0。得到:

$$\left. \begin{aligned} \sum_i (Y_i - a - b \cdot X_i) &= 0 \\ \sum_i (Y_i - a - b \cdot X_i) \cdot X_i &= 0 \end{aligned} \right\} \quad (7)$$

把 5 场中目标重心位置看做观测值 (X_i, Y_i) , $i=1, \dots, 5$ 。代入上式解方程组得到 a, b 的值。根据 a, b 的值不难得到目标运动方向的角度。

我们把视场正右方向定义为角度 0, 正上方向定义为角度 $\pi/2 \approx 1.57$, 正左方向定义为

角度 $\pi \approx 3.14$ ，正下方向定义为角度 $3/2\pi \approx 4.71$ 。如果目标不动它的方向规定为 -1 。对视场中三个目标进行跟踪实验得到以下数据和图像，图 2 是三个目标合并前的情景，01, 02, 03 为三个目标，其中 01 是主跟目标。虚线表示目标运动方向，图 3 显示三个目标合并在一起的情景，图 4 表示目标分离的情况。

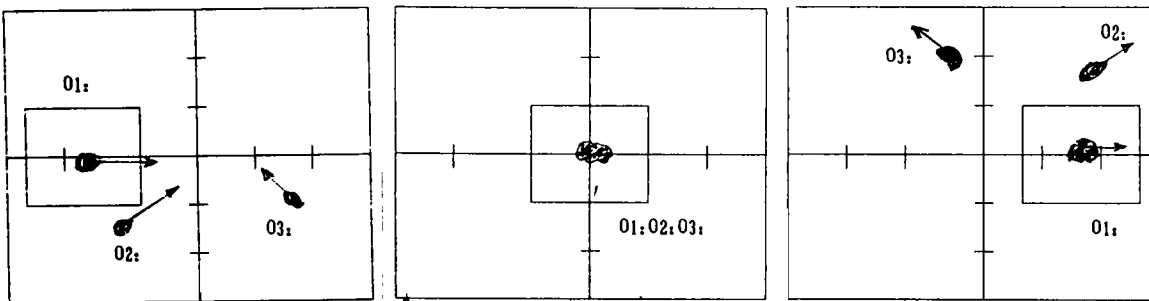


图 2 三个目标合并前的图像 图 3 三个目标在运动中合并在一起的图像 图 4 三个目标合并后又离开的图像

用 (1) 式计算得到运动方向的数据：

目标\时刻	1	2	3	4	5	6	7	8	9	10
0 ₁	0.2	0.1	0.8	1.32	1.34	0.7	0.3	0.2	0.1	0.2
0 ₂	1.01	1.02	X	X	X	1.04	1.05	1.04	1.03	1.02
0 ₃	2.44	2.46	2.47	X	X	X	2.42	2.43	2.41	2.42

上表可以看出：在时刻 1 和时刻 2，我们在视场中看见三个目标 01, 02, 03（如图 2 的情形），在时刻 3，我们在视场中只看见两个目标，跟据第一节中给出的跟踪策略可以判断出其中一个目标是由 01, 02 合并而来，另一个是 03。我们把 01, 02 合并前各自的运动方向数据 0.1 和 1.02 加以保存，在时刻 4，我们在视场中只看见一个目标，（如图 3 的情形），同样跟据第一节中给出的跟踪策略可以判断出是 01, 02, 03 三个目标合在一起。我们把 03 合并前的方向数据 2.47 加以保存。在时刻 6，目标一分为二，此时比较一下合并前与分离后的目标数目，发现还有一个目标没有分离出来。将等待一段时间，当然这个等待是有限度的。实际应用时我们确定一个阈值 T_n 。等待时间超过 T_n ，我们就可认为 03 目标丢失，继续下一步。在时刻 7，视场中有三个目标，根据跟踪策略可以判断出这三个目标都是从时刻 6 中的目标分离出来，我们把此时三个目标的运动方向数据 0.3, 1.05, 2.42，与以前保存的三个数据 0.1, 1.02, 2.47 比较，得出编批结果。

实验证明，这种方法计算目标运动方向是相当精确的。对于合并交叉前与分离后目标相对跟踪关系的准确编批提供了良好的判据。如果把目标重心位置观测值从 5 场提高到 7 场或更高，就会提高计算精度，当然提高精度的同时，也使得计算量加大。

4 结 论

以上方法,在跟瞄系统工程上应用取得了较好的跟踪效果。跟踪视场内最多四个运动目标,并且有四个目标同时交叉的情况时效果很好,合并前与分离后目标编批准确,跟踪稳定。但也存在一些问题,如目标合并之后分离之前,其运动方向突然发生变化超过 90° 时,目标跟踪编号将发生错误。但由于这种情况极少发生,所以不会有较大影响,本算法还有一些细节问题,如新目标出现的判断,目标短暂丢失情况的处理,目标一分为二或一分为多,以及目标丢失等情况的处理。实施本算法之前,还要进行信号二值化处理,目标捕获等问题。由于篇幅所限不能详细叙述。

参 考 文 献

- [1] Theo Pavlidis, Algorithms for graphics and image processing. Maryland Computer Science Press, Inc. 1982
- [2] 罗耀光, 盛立东, 模式识别. 北京人民邮电出版社, 1989
- [3] 沈恒范, 概率与统计. 上海人民教育出版社, 1966

An Algorithm of TV Tracking for Cross-Moving Multiple Objects

Zhu Jin

(Changchun Institute of Optics and Fine Mechanics,
Chinese Academy of Sciences, Changchun 130021)

Abstract

An algorithm for automatically tracking moving multiple objects that cross in the field of view of a tracking and measuring television system is presented in this paper. This algorithm includes tracking tactics, a judgement on the overlapped and separated moment of these objects, and the affirmation of the correspondence relationship between these objects's images. The least square method is used for calculating the direction of each object's movement.

Key words: Cross-moving, Multiple Objects, TV Tracking, Least square.