

基于分段复合速度匹配的点目标检测算法

刘志刚, 卢焕章, 陈辉煌

(国防科学技术大学 ATR 实验室, 湖南 长沙 410073)

摘要:针对低信噪比下运动点目标检测这一难题,提出了一种新算法——分段复合速度匹配法,该方法通过递推使用速度匹配滤波器有效地实现了目标的能量累加。与其他检测算法相比,该方法具有运算简单、检测时间快等特点。对新算法的原理和性能进行了详细分析,并通过实验证明了该算法的有效性。

关键词:检测前跟踪; 点目标检测; 低信噪比; 分段复合速度匹配法

中图分类号: TP391.4 **文献标识码:** A **文章编号:** 1007-2276(2004)04-0366-05

Point target detection algorithm based on Multistage Velocity Matching

LIU Zhi-gang, LU Huan-zhang, CHEN Hui-huang

(ATR Laboratory, National University of Defense Technology, Changsha 410073, China)

Abstract: To detect moving point targets in low SNR, a new detection method——Multistage Velocity Matching Algorithm is presented. In the algorithm, target energy is accumulated by use of Velocity Matching Filter recursively. Compared with other detection methods, the new method's computation is simple, and detection velocity is fast. Theory and performance of the method are analyzed, and simulation experiment proves that it is an effective method.

Key words: Track-Before-Detect; Point target detection; Low SNR; Multistage Velocity Matching Algorithm

0 引言

低信噪比下红外点目标的检测一直是红外搜索和跟踪(IRST)系统的重要研究方向。低信噪比下,目标淹没在噪声中,仅依靠亮度信息无法从单帧图像中分割出目标;点目标没有形状特征可以利用,使那些基于目标形状的检测算法也不能直接使用。解决此

问题的一个方法是利用检测前跟踪(TBD),即沿着目标的运动轨迹对多帧图像的目标信息进行累加,提高信噪比,然后再进行检测。

目前主要的检测前跟踪方法包括:速度滤波器^[1]、3D 匹配滤波器^[2~4]、序贯假设检验法^[5,6](SHT)、动态规划法^[7~9](DP)。前三者属于穷尽搜索方式,要对图像序列中所有可能的轨迹进行计算,当序列帧数短时,信噪比提高不够;而序列帧数长时,轨迹的数目巨大,

收稿日期:2003-10-19; 修订日期:2003-12-05

作者简介:刘志刚(1971-),男,湖南湘潭人,博士生,研究方向为红外目标检测与识别。

没有实际利用价值。Yair Barniv 在参考文献[7]中通过 DP 算法实现了目标检测,该方法计算简单,然而在低信噪比下,要经过长时间的处理,才能得到高的目标检测概率。如对于 SNR 为 1.8,虚警率为 0.01,大小为 128×128 的图像序列,要运算到 120 帧时,目标的检测概率才能达到 0.9,检测时间过长,留给系统的后续处理时间减少,难以完成跟踪、识别等任务。

为了缩短目标的检测时间,同时使运算量增长不大,提出了分段复合速度匹配算法(MVMA),该算法通过重复使用速度匹配滤波器,能有效地实现目标的能量累加,速度匹配滤波器处理的序列长度由系统的运算能力和目标的检测要求决定。

1 速度匹配滤波器

对于低信噪比的点目标,提高其信噪比的方法是沿着目标的运动轨迹进行能量累加。当目标的速度未知时,需要对所有的运动轨迹进行能量累加,然后进行判断。当信噪比大于 1 时,目标轨迹的能量均值要大于噪声轨迹的能量均值,因此把最大的轨迹能量作为目标轨迹能量,并定义这种方法为速度匹配滤波器。

要得到速度匹配滤波器,首先要构造轨迹树,而轨迹树的构造由目标经过的图像帧数决定,这里定义为匹配级数。在由 N 帧图像组成的序列中,假设目标在图像序列中作直线运动,而且在 X 轴和 Y 轴两个方向上的帧间速度小于每帧一个像素,则目标点的轨迹树可以预先构造出来。以 $N=3$ 为例,设第 3 帧的像素 (i, j) 是目标点,目标在第 2 帧通过的区域是以 (i, j) 为中心,长、宽为 3 的一个正方形;目标在第 1 帧通过的区域是以 (i, j) 为中心,长、宽为 5 的一个正方形。为了表示方便,各像素点用一个标志号表示,如图 1 所示。经分析,过第 3 帧像素点 1 的轨迹有 49

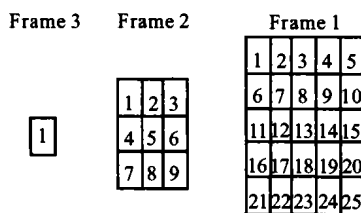


图 1 像素的标志图

Fig. 1 Index figure of pixels

条,表 1 列出了这些轨迹。这些轨迹可以由一棵轨迹树来表示。树的根是第 3 帧像素点,树的二级节点是第 2 帧像素点,树的三级节点是第 1 帧像素点。而轨迹就是从根节点到最下层节点的一条路径,由表 1 得到的轨迹树如图 2 所示。

表 1 轨迹表

Tab. 1 Track table

Track	Node
1	1-1-1
2	1-1-2
3	1-1-6
4	1-1-7
⋮	⋮
46	1-9-19
47	1-9-20
48	1-9-24
49	1-9-25

求过根结点的轨迹的最大能量为:

$$I = z_1(N) + \max\{z_i(N-1) + \max[\dots + \max(z_k(1))]\} \quad (1)$$

式中 $z_i(k)$ 表示 k 帧时像素点 i 的灰度值; $z_i(k) + \max[\dots]$ 表示 k 帧像素点 i 的灰度值加上该点在 $k-1$ 帧的各子节点的最大累积能量。

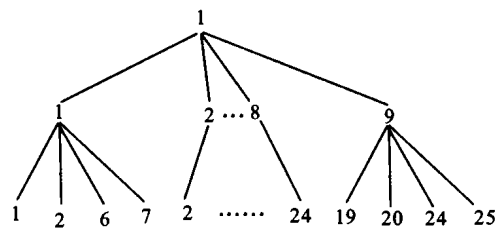


图 2 轨迹树

Fig. 2 Track tree

2 分段复合速度匹配算法

对于长图像序列,如果直接使用匹配级数等于序列长的速度匹配滤波器,由于构造的轨迹树过于庞大,而实际系统的资源有限,无法得到实时结果,此时可以采用分段复合速度匹配算法。该算法把原始序列分割成多个小段,每小段的长度由速度匹配滤波器的匹配级数决定。当匹配级数为 N 时,每小段的长度为 $N-1$ 。匹配滤波器处理的图像序列由本段的 $N-1$ 帧原始图像与上一段的输出图像组成,算法的

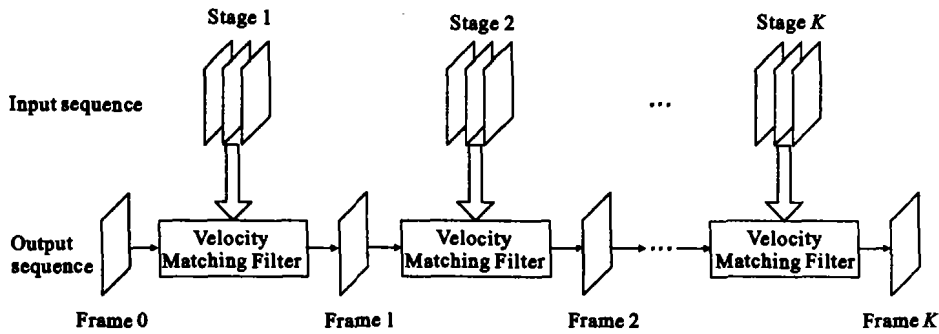


图 3 分段复合速度匹配算法原理图

Fig. 3 Sketch of MVMA

原理如图 3 所示。通过修改公式(1)可以得到分段复合速度匹配算法的表达式:

$$I_m(k) = z_m(kN - k) + \max_i \{ z_i(kN - k - 1) + \dots + \max_j [z_j(kN - k - N + 1) + \max_l (I_l(k - 1))] \} \quad (2)$$

假设图像序列分成 K 个小段, 每段由 $N - 1$ 帧图像组成, 完整的分段复合速度匹配算法如下:

(1) 初始化 当 $k = 0$, 输出图像的所有像素值置为 0

$$I_m(0) = 0 \quad (3)$$

(2) 递推 当 $1 \leq k \leq K$, 用匹配级数为 N 的速度匹配滤波器得到滤波图像

$$I_m(k) = z_m(kN - k) + \max_i \{ z_i(kN - k - 1) + \max_j [\dots + \max_l (I_l(k - 1))] \} \quad (4)$$

(3) 终止 当第 K 段图像处理完毕, 用门限 V_T 对输出图像 $I(K)$ 进行分割, 过门限的点是候选目标点。

3 性能分析

由公式(3)、(4)可以看出, 算法的主要操作是求最大值。在进行性能分析之前, 首先引入参考文献[7]对两个独立高斯变量求最大值的分析过程。设两个互不相关的高斯随机变量 X, Y , 概率分布分别为 $N(u_x, \sigma_x^2), N(u_y, \sigma_y^2)$, 定义:

$$\left. \begin{aligned} \Delta &= u_x - u_y \\ \theta &= (\sigma_x^2 + \sigma_y^2)^{1/2} \\ \phi &= 1/\sqrt{2\pi} \exp(-\Delta^2/2\theta^2) \\ \Phi &= \int_{-\infty}^{\Delta/\theta} 1/\sqrt{2\pi} \exp(-x^2/2) \\ \bar{\Phi} &= 1 - \Phi \end{aligned} \right\} \quad (5)$$

则均值 $u_{\max(X,Y)}$ 、方差 $\sigma_{\max(X,Y)}^2$ 为:

$$\left. \begin{aligned} u_{\max(X,Y)} &= u_x \Phi + u_y \bar{\Phi} + \theta \phi \\ \sigma_{\max(X,Y)}^2 &= \sigma_x^2 \Phi + \sigma_y^2 \bar{\Phi} + (\Delta \Phi + \theta \phi)(\Delta \bar{\Phi} - \theta \phi) \end{aligned} \right\} \quad (6)$$

对于有 n 个独立变量, 概率分布为 $N(u_i, \sigma_i^2), i = 1, 2, \dots, n$, 它的最大值可通过重复求两变量最大值的方法求出。

$$\max(X_1, X_2, \dots, X_n) = \max\{X_1, \max[X_2, \max(\dots, \max(X_{n-1}, X_n))]\} \quad (7)$$

输出图像中噪声点和目标点的能量, 可根据轨迹树的结构用公式(7)由下至上递推求得。现在以匹配级数 $N = 3$ 的分段复合帧速度匹配器为例, 简要说明输出噪声点与目标点的均值与方差的求解, 其他不同匹配级数的分段复合速度匹配器的分析与此类似。速度匹配滤波器处理的 3 帧图像由第 $2k, 2k - 1$ 帧原始图像和第 $k - 1$ 帧输出图像组成。轨迹树如图 2 所示, 树的第一层到第三层分别是第 $2k$ 帧原始图像、第 $2k - 1$ 帧原始图像、第 $k - 1$ 帧输出图像。

(1) k 帧输出噪声点的计算

对于 k 帧输出噪声点的分析, 是在轨迹树中的所有像素点都是噪声点的基础上进行的。其中第 $k - 1$ 帧输出图像中节点 1-25 的灰度分布为 $N(u_{k-1}, \sigma_{k-1}^2)$, u_{k-1} 和 σ_{k-1}^2 是第 $k - 1$ 帧输出噪声点的均值与方差; 第 $2k, 2k - 1$ 帧原始图像中噪声点的灰度分布为 $N(u_n, \sigma_n^2)$, u_n 和 σ_n^2 是原始输入图像中噪声点的均值与方差。

首先要计算轨迹树中从二级节点到三级节点的最大能量累加值 $I'_i, i = 1, 2, \dots, 9$, I'_i 的大小与二级节点 i 所关联的三级节点数有关。其中二级节点 1 的最大能量累加值 I'_1 的均值与方差是:

$$\left. \begin{aligned} u_1(2) &= u_n + u_{\max(I'_1(k-1), I'_2(k-1), I'_6(k-1), I'_7(k-1))} \\ \sigma_1^2(2) &= \sigma_n^2 + \sigma_{\max(I'_1(k-1), I'_2(k-1), I'_6(k-1), I'_7(k-1))}^2 \end{aligned} \right\} \quad (8)$$

类似地求得 $I'_i, i=2,3,\dots,9$ 的均值与方差。输出噪声点的均值与方差是:

$$\begin{aligned} u_k &= u_1(1) = u_n + u_{\max(I'_1, I'_2, \dots, I'_9)} \\ \sigma_k^2 &= \sigma_1^2(1) = \sigma_n^2 + \sigma_{\max(I'_1, I'_2, \dots, I'_9)}^2 \end{aligned} \quad (9)$$

(2) k 帧输出目标点的计算

含有不同目标轨迹的轨迹树,其最大能量轨迹的均值与方差互不相同,而三级轨迹树共有 49 条轨迹。在计算出含有不同目标轨迹的轨迹树的最大能量轨迹的均值与方差后,再求其平均值,就可得到 k 帧输出目标点的均值与方差。

如果轨迹 1-1-1 是目标轨迹,则第 $k-1$ 帧输出图像中节点 2-25 是噪声点,灰度分布为 $N(u_{k-1}, \sigma_{k-1}^2)$,节点 1 是目标点,灰度分布为 $N(\bar{u}_{k-1}, \bar{\sigma}_{k-1}^2)$, \bar{u}_{k-1} 和 $\bar{\sigma}_{k-1}^2$ 是第 $k-1$ 帧输出目标点的均值与方差; $2k-1$ 帧原始图像中节点 2-9 为噪声点,灰度分布为 $N(u_n, \sigma_n^2)$; $2k-1$ 帧原始图像的节点 1 和 $2k$ 帧原始图像的节点 1 是目标点,灰度分布为 $N(u, \sigma^2)$, u 和 σ^2 是原始输入图像中目标点的均值与方差。同计算噪声点一样,先求出轨迹树中二级节点到三级节点的最大能量累加值 I_i 的均值与方差,其中二级节点 1 的最大能量累加值 I_1 的均值与方差是:

$$\left. \begin{aligned} \bar{u}_1(2) &= u, + u_{\max(I_1(k-1), I_2(k-1), I_6(k-1), I_7(k-1))} \\ \bar{\sigma}_1^2(2) &= \sigma_n^2 + \sigma_{\max(I_1(k-1), I_2(k-1), I_6(k-1), I_7(k-1))}^2 \end{aligned} \right\} \quad (10)$$

其他二级节点是噪声点,计算方法同公式(8)。目标轨迹为 1-1-1 时,最大能量轨迹的输出为:

$$\left. \begin{aligned} \bar{u}_{1-1-1} &= \bar{u}_1(1) = u, + u_{\max(I_1, I_2, \dots, I_9)} \\ \bar{\sigma}_{1-1-1} &= \bar{\sigma}_1^2(1) = \sigma_n^2 + \sigma_{\max(I_1, I_2, \dots, I_9)}^2 \end{aligned} \right\} \quad (11)$$

其他目标轨迹的计算与 1-1-1 的计算相似。在计算出 49 条目标轨迹的最大能量轨迹的均值与方差后, k 帧输出目标点的均值与方差为:

$$\left. \begin{aligned} \bar{u}_k &= (\bar{u}_{1-1-1} + \bar{u}_{1-1-2} + \dots + \bar{u}_{1-9-25})/49 \\ \bar{\sigma}_k^2 &= (\bar{\sigma}_{1-1-1}^2 + \bar{\sigma}_{1-1-2}^2 + \dots + \bar{\sigma}_{1-9-25}^2)/49 \end{aligned} \right\} \quad (12)$$

4 检测概率与虚警率

为了计算检测概率与虚警率,假设经过分段复合速度匹配法处理后的输出噪声点服从高斯分布 N

(u_k, σ_k^2) , 目标点服从高斯分布 $N(\bar{u}_k, \bar{\sigma}_k^2)$ 。第 k 帧时的虚警率为:

$$P_{FA} = 1 - \Phi^n \left(\frac{V_k - u_k}{\sigma_k} \right) \quad (13)$$

式中 V_k 是 k 帧时的门限; Φ 是标准高斯累积分布函数; n 是一帧图像中像素点的数目。

当与实际目标点距离小于 1 个像素的邻域内有像素点的亮度超过门限 V_k 时,认为目标被检测,故第 k 帧时的检测概率为:

$$P_D = 1 - \Phi \left(\frac{V_k - \bar{u}_k}{\sigma_k} \right) \Phi^n \left(\frac{V_k - u_k}{\sigma_k} \right) \quad (14)$$

分段复合速度匹配法的检测性能与匹配级数 N 直接相关,匹配级数 $N=2$ 的分段复合速度匹配法就是邻域 $q=9$ 的动态规划算法。为了更好地比较算法的性能,绘制了 $N=2,3,4,5,6$ 时,算法的检测概率示意图,见图 4。其条件是虚警率 $P_{FA}=0.01$, $SNR=1.8$, 图像尺寸为 128×128 。图中示出,相同条件下, N 越大,目标的检测概率越高,算法到达高检测

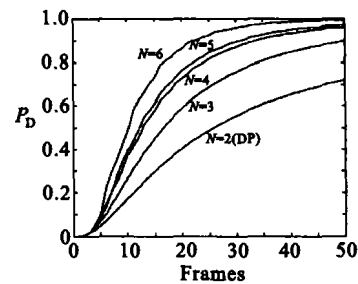


图 4 分段复合速度匹配法检测概率图 ($SNR=1.8, P_{FA}=0.01$)

Fig. 4 Detection probability of MVMA ($SNR=1.8, P_{FA}=0.01$)

概率的时间越短,表 2、表 3 列出了 $SNR=1.8, 2.0$ 时, N 取不同值时算法的处理时间(处理的帧数)。分段复合速度匹配法能够有效地提高目标的信噪比,但运算量随着 N 的增大而增加,表 4 列出了 N 取不同值时算法的复杂度。

表 2 $SNR=1.8, P_{FA}=0.01, P_D=0.9$ 时,分段复合速度匹配法的帧数

Tab. 2 Frame numbers needed by MVMA, when $SNR=1.8, P_{FA}=0.01, P_D=0.9$

N	2(DP)	3	4	5	6
Frames	120	50	34	31	23

表 3 SNR=2.0, $P_{FA}=0.01$, $P_D=0.9$ 时,分段复合速度匹配法的帧数

Tab. 3 Frame numbers needed by MVMA, when SNR=2.0, $P_{FA}=0.01$, $P_D=0.9$

N	2(DP)	3	4	5	6
Frames	32	24	19	18	15

表 4 分段复合速度匹配法的运算复杂度

Tab. 4 Computation complexity of MVMA

N	2(DP)	3	4	5	6
Complexity	$O(9n)$	$O(49n/2)$	$O(81n/3)$	$O(121n/4)$	$O(169n/5)$

Note: N is the number of pixels in an image

5 实验结果

用 30 帧 128×128 的红外图像数据对算法进行检验。序列背景是夜空,在序列中人为加入一运动点目标,目标的起始位置是(20,32),X 轴方向的速度为每帧 0.3 像素,Y 轴方向的速度为每帧 0.8 像素。由于背景中有一些静止的恒星,且夜空背景是类似于非零均值的高斯噪声,因此在使用新算法之前,要对序列进行预处理。首先把每帧图像与前 10 帧图像的均值相减,去掉图像中静止的背景;然后对新图像进行归一化处理,用 7×7 窗口滑过整个图像,求取窗口内像素的均值与方差,将窗口内像素值与均值的差除以方差就得到归一化后的像素值。经过预处理后的图像序列就可以用分段复合速度匹配法进行处理,速度匹配器的匹配级数取 4。图 5 是检测过程中的处理结果,(a)是红外序列中的一帧原始图像,背景中有一些恒星;(b)是预处理之后的一帧图像;(c)是最后得到的目标轨迹。



(a) 第 30 帧图像 (b) 第 30 帧图像预处理后的结果 (c) 目标轨迹
(a) The 30th frame (b) Preprocessed result of the 30th frame (c) Target track of the 30th frame

图 5 检测过程处理结果图

Fig. 5 Sketch of processing result of detection

6 结论

为了解决低信噪比下点目标的快速检测问题,提出了分段复合速度匹配法。与已有的检测算法相比,新算法的运算复杂度、存储量和目标确认时间等综合性能有着明显的改善。对新算法进行了详细的性能分析,并通过与动态规划算法的对比,发现信噪比越低,新算法对检测性能的改善越明显,而且新算法的匹配级数 N 越大,性能越好。将新算法用于实际图像序列之前,需要去除背景杂波和归一化,通过对红外序列进行实验,成功地实现了目标检测,证明了新算法的有效性。

参考文献:

- [1] Mohanty N C. Computer tracking of moving point targets in space[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1981, 3(5): 606-611.
- [2] Reed I S, Gagliardi R M, Shao R M. Application of three-dimensional filtering to moving target detection[J]. IEEE Trans on Aerospace and Electronic Systems, 1983, 19(6): 898-905.
- [3] Reed I S, Gagliardi R M, Stotts L B. Optical moving target detection with 3-D matched filtering[J]. IEEE Trans on Aerospace and Electronic Systems, 1988, 24(4): 327-335.
- [4] Reed I S, Gagliardi R M, Stotts L B. A recursive moving-target-indication algorithm for optical image sequences[J]. IEEE Trans on Aerospace and Electronic Systems, 1990, 26(3): 434-440.
- [5] Steven D Blostein, Thomas S Huang. Detecting small moving objects in image sequences using sequential hypothesis testing [J]. IEEE Trans on Signal Processing, 1991, 39(7): 1611-1629.
- [6] Steven D Blostein, Haydn S Richardson. A sequential detection approach to target tracking[J]. IEEE Trans on Aerospace and Electronic Systems, 1994, 30(1): 197-212.
- [7] Yair Barniv. Dynamic programming solution for detecting dim moving targets[J]. IEEE Trans on Aerospace and Electronic Systems, 1985, 21(1): 144-156.
- [8] Yair Barniv, Offer Kella. Dynamic programming solution for detecting dim moving targets part II: analysis[J]. IEEE Trans on Aerospace and Electronic Systems, 1987, 23(6): 776-788.
- [9] Tonissen S M, Evans R J. Performance of dynamic programming techniques for track-before-detect [J]. IEEE Trans on Aerospace and Electronic Systems, 1996, 32(4): 1440-1451.