

二维时空模糊熵运动检测中的自适应门限新算法

刘 洋, 初秀琴, 李玉山

(西安电子科技大学电路 CAD 研究所, 陕西西安 710071)

摘 要: 提出一种基于二维时空模糊熵准则自适应确定运动检测门限的新算法. 通过推导出二维模糊熵门限的快速实现形式, 利用积分和迭代操作避免了传统二维模糊熵门限求解过程中点的重复计算, 将灰度级为 N 的图像的各种运算操作次数从 $O(N^4)$ 降低到小于或等于 $O(N^3)$. 将运动检测归结为两个二值划分问题, 无需已知背景分布的具体形式和参数, 利用二维模糊熵准则自适应确定门限 T . 实验结果表明, 该方法在目标和背景对比度偏低的情况下也可以提取出完整的运动信息, 易于实现实时处理.

关键词: 二维时空模糊熵准则; 运动检测门限; 运动目标提取; 积分操作; 迭代操作

中图分类号: TN911.73 **文献标识码:** A **文章编号:** 0372-2112(2008)06-1092-06

A New Adaptive Thresholding Algorithm for Motion Detection Based on Two-Dimensional Spatio-Temporal Fuzzy Entropy Principle

LIU Yang, CHU Xiur qin, LI Yu shan

(Institute of Electronic CAD, Xidian University, Xi'an, Shaanxi 710071, China)

Abstract: A adaptive thresholding algorithm for motion detection based on two-dimensional spatio-temporal fuzzy entropy principle is proposed. A fast solution for calculating the two-dimensional fuzzy entropy threshold is deduced, in which repeated calculation is avoided by using integral and iterative operation, and the various calculation operation is reduced from $O(N^4)$ to less than $O(N^3)$ for N gray level image. The motion detection is reduced to two binary partition problems, and two-dimensional spatio-temporal fuzzy entropy principle is used to determine the threshold T , where the explicit function form or parameters of background distribution are not needed to be known. The experimental results show that the information of moving objects which have low contrast to background can also be extracted completely by this method in real time.

Key words: two-dimensional fuzzy entropy principle; motion detection threshold; moving object extraction; integral operation; iterative operation

1 引言

运动区域的有效分割是目标跟踪、行为理解以及 MPEG4 视频编码等计算机视觉应用领域中基本而又重要的一步, 也是运动目标检测研究领域中的难点问题^[1]. 常用的运动检测方法是对连续的图像序列中相邻两帧或多帧之间的帧差绝对值进行阈值化来确定灰度变化较大的像素点, 从而估算出运动区域. 那么, 决定运动检测性能的一个重要因素是如何选取合适的门限来有效区分目标和背景. 由于受到应用环境和噪声的影响, 不同的图像序列包含不同的图像分量, 而且它们的变量特性也明显不同, 所以对于运动目标检测, 任何预设的固定门限值都是不合适的. 从提取运动目标的角度

而言, 应该选取较小的门限以提取目标的完整信息. 而从抑制噪声的角度而言, 应该选取较大的门限以防止虚警. 相关文献中提出了两种解决方法: (1) 采用学习和训练的方法来估计背景分布, 根据像素点是否服从背景分布来进行运动检测^[2,3]. 这些方法在一定程度上实现了门限的自适应选取, 但是通常需要假设背景像素点服从某一特定的分布函数, 并且计算量偏大. (2) 灰度直方图熵门限方法^[4-6]. 该类方法对帧差绝对值建立一维或二维直方图, 并运用熵准则求解全局门限, 但是根据研究发现, 如果目标和背景的对比度偏低, 采用该类方法得到的门限进行运动目标提取会产生很大的误差.

针对上述问题, 在考虑到图像序列中目标和背景之间存在模糊性的基础上, 针对帧差值大于 0 和小于 0 的

像素点分别利用二维模糊熵准则确定门限矢量(T^+ , T^-), 并选取 $T = \min(T^+, T^-)$ 作为最终的运动检测门限, 其本质也就是将帧差值大于 0 和小于 0 的像素点分别看作是一个二值划分问题. 通过理论分析和推导, 给出二维模糊熵门限的快速实现形式, 将灰度级为 N 的图像的各种运算操作次数从 $O(N^4)$ 降低到小于或等于 $O(N^3)$, 便于实现运动检测的实时处理.

2 二维模糊熵准则

信息熵的概念是由 Shannon 率先提出, 其观点是一个系统事件产生的信息增益是与其产生的概率成反比的^[7]. 但是, 在描述现实生活中很多问题的时候存在一定的模糊性, 鉴于此, Zadeh 提出模糊熵的定义^[8]

$$H_{fuzzy}(A) = - \sum_{i=1}^N \mu_A(x_i) p(x_i) \log(p(x_i)) \quad (1)$$

其中, $\mu_A(x_i)$ 表示系统事件集合 A 的隶属函数, $p(x_i)$ 表示系统事件 $x_i (i = 1, 2, \dots, N)$ 产生的概率. Cheng 等利用 Zadeh 的模糊熵概念, 提出了利用一维灰度直方图和二维灰度直方图分割静态图像中目标和背景的模糊熵门限算法^[9, 10], 并都取得了比熵门限算法更为优越的分割效果.

假设图像具有 N 个灰度等级, (x, y) 位置的像素点灰度值及其邻域平均值分别用 $I(x, y)$ 和 $\bar{I}(x, y)$ 表示, 二维直方图用 $C(N \times N)$ 来进行表示, 其元素 $c(i, j)$ 表示 $I(x, y)$ 的灰度值为 i , $\bar{I}(x, y)$ 的灰度值为 j 的参数对 $(I(x, y), \bar{I}(x, y))$ 的共生次数. 二维直方图可以看作是集合 X 和集合 Y 的笛卡儿乘积, 其中集合 X 表示灰度级, 集合 Y 表示局部平均灰度级: $X = Y = \{0, 1, 2, \dots, N - 1\}$. 在图像 I 中存在目标分布 Object X 和背景分布 Background X ; 在邻域平均图像 \bar{I} 中也存在着目标分布 Object Y 和背景分布 Background Y , 具体函数形式表示如下

$$\begin{cases} \text{Object}X = \bigcup_{i \in X} \frac{\mu_{OX}(i)}{i} = \bigcup_{i \in X} \frac{S(i, a, b)}{i} \\ \text{Background}X = \bigcup_{i \in X} \frac{\mu_{BX}(i)}{i} = \bigcup_{i \in X} \frac{Z(i, a, b)}{i} \\ \text{Object}Y = \bigcup_{i \in Y} \frac{\mu_{OY}(i)}{i} = \bigcup_{i \in Y} \frac{S(i, a, b)}{i} \\ \text{Background}Y = \bigcup_{i \in Y} \frac{\mu_{BY}(i)}{i} = \bigcup_{i \in Y} \frac{Z(i, a, b)}{i} \end{cases} \quad (2)$$

其中, $\mu_{OX}(i)$ 和 $\mu_{BX}(i)$ 分别表示图像 I 中目标和背景的隶属函数, $\mu_{OY}(i)$ 和 $\mu_{BY}(i)$ 分别表示邻域平均图像 \bar{I} 中目标和背景的隶属函数, $S(\cdot)$ 和 $Z(\cdot)$ 是表征图像模糊性的隶属函数, 且 $Z(\cdot) = 1 - S(\cdot)$. $S(x, a, b)$ 隶属函数的形状是由 $0 \leq a < b \leq N - 1$ 确定的, 定义如下

$$S(i, a, b) = \begin{cases} 0, & \text{if } i \leq a \\ 2(i - a)^2 / (b - a)^2, & \text{if } a < i \leq (a + b) / 2 \\ 1 - 2(i - b)^2 / (a - b)^2, & \text{if } (a + b) / 2 < i < b \\ 1, & \text{if } i \geq b \end{cases} \quad (3)$$

在这里, 我们假设目标比背景具有更高的灰度值. 根据模糊集合的性质

$$\begin{cases} \mu_O(i, j) = \mu_{OX \times OY}(i, j) = \min(\mu_{OX}(i), \mu_{OY}(j)) \\ \mu_B(i, j) = \mu_{BX \times BY}(i, j) = \min(\mu_{BX}(i), \mu_{BY}(j)) \end{cases} \quad (4)$$

可以分别得到目标和背景在二维直方图中的划分区域

$$\begin{cases} R_O = \{(i, j) | \mu_O(i, j) \leq 1, 0 \leq i, j \leq N - 1\} \\ R_B = \{(i, j) | \mu_B(i, j) \leq 1, 0 \leq i, j \leq N - 1\} \end{cases} \quad (5)$$

根据上述定义, 整个图像的模糊熵可以定义为

$$H(\text{image}) = H(R_B) + H(R_O) \quad (6)$$

$$H(R_B) = - \sum_{i=0}^{b-1} \sum_{j=0}^{b-1} \mu_B(i, j) \frac{c(i, j)}{\sum_{i=0}^{b-1} \sum_{j=0}^{b-1} c(i, j)} \log \left(\frac{c(i, j)}{\sum_{i=0}^{b-1} \sum_{j=0}^{b-1} c(i, j)} \right) \quad (7)$$

$$H(R_O) = - \sum_{i=a+1}^{N-1} \sum_{j=a+1}^{N-1} \mu_O(i, j) \frac{c(i, j)}{\sum_{i=a+1}^{N-1} \sum_{j=a+1}^{N-1} c(i, j)} \log \left(\frac{c(i, j)}{\sum_{i=a+1}^{N-1} \sum_{j=a+1}^{N-1} c(i, j)} \right) \quad (8)$$

将定义的二维模糊熵函数作为目标函数, 通过设定二维模糊熵最大时参数 a 和 b 的值来选择最优门限 $(a + b) / 2$, 也就是模糊区域的交叉点, 以实现目标和背景的分类.

3 二维模糊熵门限快速算法

为了求解全局最优参数 (a, b) , 必须采用搜索算法对每个 (a, b) 都要计算 $O(N^2)$ 次来求解目标和背景的模糊熵, 由于有 $N(N - 1) / 2$ 种 (a, b) 组合, 所以求得全局门限矢量的运算次数是 $O(N^4)$. 对于实时性要求比较高的系统而言, 该方法的计算量是无法满足要求的. 根据式(6)~(8)可以看出, 在计算每 (a, b) 个模糊熵的时候很多的点都被重复进行运算, 造成很多不必要的资源浪费.

我们推导出一种实现二维模糊熵门限的新形式, 利用积分和迭代操作避免了点的重复计算, 从而将各种运算操作次数从 $O(N^4)$ 降低到小于或等于 $O(N^3)$. 为了使下面的叙述更加简明易懂, 将由 $c(i, j) \log(c(i, j))$ 元素组成的矩阵用 Q 表示, 也就是 Q 矩阵中的每个元素 $q(i, j)$ 等于 $c(i, j) \log(c(i, j))$, 并定义下述表示符号

$$P_{LU1}(x) = \sum_{i=0}^x \sum_{j=0}^x c(i, j) \quad (9)$$

$$P_{RL1}(x) = \sum_{i=x}^{N-1} \sum_{j=x}^{N-1} c(i, j) \quad (10)$$

$$P_{LU2}(x) = \begin{cases} c(0, 0), & \text{if } x = 0 \\ c(x, x) + \sum_{i=0}^{x-1} (c(i, x) + c(x, i)), & \text{otherwise} \end{cases} \quad (11)$$

$$P_{RL2}(x) = \begin{cases} c(N-1, N-1), & \text{if } x = N-1 \\ c(x, x) + \sum_{i=x+1}^{N-1} (c(i, x) + c(x, i)), & \text{otherwise} \end{cases} \quad (12)$$

$$Q_{LU1}(x) = \sum_{i=0}^x \sum_{j=0}^x c(i, j) \log(c(i, j)) \quad (13)$$

$$Q_{RL1}(x) = \sum_{i=x}^{N-1} \sum_{j=x}^{N-1} c(i, j) \log(c(i, j)) \quad (14)$$

$$Q_{LU2}(x) = \begin{cases} q(0, 0), & \text{if } x = 0 \\ q(x, x) + \sum_{i=0}^{x-1} (q(i, x) + q(x, i)), & \text{otherwise} \end{cases} \quad (15)$$

$$Q_{RL2}(x) = \begin{cases} q(N-1, N-1), & \text{if } x = N-1 \\ q(x, x) + \sum_{i=0}^{x-1} (q(i, x) + q(x, i)), & \text{otherwise} \end{cases} \quad (16)$$

即, $P_{LU1}(x)$ 和 $P_{RL1}(x)$ 分别表示矩阵 C 中坐标 (x, x) 左上方以及右下方元素的级数和; $P_{LU2}(x)$ 和 $P_{RL2}(x)$ 分别表示矩阵 C 中坐标 (x, x) 对应行左边和对应列上边元素的级数和以及对应行右边和对应列下边元素的级数和; $Q_{LU1}(x)$ 和 $Q_{RL1}(x)$ 分别表示矩阵 Q 中坐标 (x, x) 左上方和右下方元素的级数和; $Q_{LU2}(x)$ 和 $Q_{RL2}(x)$ 分别表示矩阵 Q 中坐标 (x, x) 对应行左边和对应列上边元素的级数和以及对应行右边和对应列下边元素的级数和. 通过分析, 可以推导出

$$\begin{aligned} H(R_B) &= -\frac{1}{P_{LU1}(b-1)} \sum_{x=0}^{b-1} \sum_{y=0}^{b-1} (\mu_B(x, y) q(x, y)) \\ &+ \frac{\log(P_{LU1}(b-1))}{P_{LU1}(b-1)} \sum_{x=0}^{b-1} \sum_{y=0}^{b-1} (\mu_B(x, y) c(x, y)) \\ &= -\frac{1}{P_{LU1}(b-1)} \sum_{x=0}^{b-1} \sum_{y=0}^{b-1} (\min(\mu_B(x), \mu_B(y)) q(x, y)) \\ &+ \frac{\log P_{LU1}(b-1)}{P_{LU1}(b-1)} \sum_{x=0}^{b-1} \sum_{y=0}^{b-1} (\min(\mu_B(x), \mu_B(y)) c(x, y)) \\ &= \frac{\log(P_{LU1}(b-1))}{P_{LU1}(b-1)} \left[\sum_{x=0}^{b-1} (\mu_B(x) P_{LU2}(x)) \right] \\ &- \frac{1}{P_{LU1}(b-1)} \left[\sum_{x=0}^{b-1} (\mu_B(x) P_{LU2}(x)) \right] \\ &= \frac{\log(P_{LU1}(b-1))}{P_{LU1}(b-1)} \left[\sum_{x=0}^a P_{LU2}(x) + \sum_{x=a+1}^{b-1} \mu_B(x) P_{LU2}(x) \right] \\ &- \frac{1}{P_{LU1}(b-1)} \left[\sum_{x=0}^a Q_{LU2}(x) + \sum_{x=a+1}^{b-1} \mu_B(x) Q_{LU2}(x) \right] \\ &= \frac{\log(P_{LU1}(b-1))}{P_{LU1}(b-1)} \left[P_{LU1}(a) + \sum_{x=a+1}^{b-1} \mu_B(x) P_{LU2}(x) \right] \end{aligned}$$

$$- \frac{1}{P_{LU1}(b-1)} \left[Q_{LU1}(a) + \sum_{x=a+1}^{b-1} \mu_B(x) Q_{LU2}(x) \right] \quad (17)$$

同理, 并利用 $\mu_O(\bullet) = 1 - \mu_B(\bullet)$ 这一性质, $H(R_O)$ 可以推导为

$$\begin{aligned} H(R_O) &= \frac{1}{P_{RL1}(a+1)} \left[\sum_{x=a+1}^{b-1} (\mu_B(x) Q_{RL2}(x)) \right] \\ &- \frac{\log(P_{RL1}(a+1))}{P_{RL1}(a+1)} \left[\sum_{x=a+1}^{b-1} \mu_B(x) P_{RL2}(x) \right] \\ &+ \log(P_{RL1}(a+1)) - \frac{Q_{RL1}(a+1)}{P_{RL1}(a+1)} \end{aligned} \quad (18)$$

式(17)和式(18)分别是式(7)和式(8)的一种新的实现方式. 通过分析式(17)和式(18)可以看出, 在全局最优门限搜索的过程中, 对每一个坐标点 (x, x) 都要反复用到 $P_{LU1}(x)$, $P_{RL1}(x)$, $P_{LU2}(x)$, $P_{RL2}(x)$, $Q_{LU1}(x)$, $Q_{RL1}(x)$, $Q_{LU2}(x)$, $Q_{RL2}(x)$, 其中 $0 \leq x \leq N-1$. 在进行全局搜索之前, 将上述这些值全部计算出来, 并分别保存到一维数组 P_{LU1} , P_{RL1} , P_{LU2} , P_{RL2} , Q_{LU1} , Q_{RL1} , Q_{LU2} , Q_{RL2} 中, 以避免数据点的重复计算. 为了快速计算这八个数组, 对矩阵 C 和 Q 分别进行积分操作

$$\begin{cases} r_0(x, y) = r_0(x-1, y) + c(x, y) \\ r_1(x, y) = r_1(x, y-1) + r_0(x, y) \end{cases} \quad (19)$$

$$\begin{cases} l_0(x, y) = l_0(x-1, y) + q(x, y) \\ l_1(x, y) = l_1(x, y-1) + l_0(x, y) \end{cases} \quad (20)$$

那么, 可以通过下式分别快速计算 $P_{LU1}(x)$, $P_{RL1}(x)$, $P_{LU2}(x)$, $P_{RL2}(x)$, $Q_{LU1}(x)$, $Q_{RL1}(x)$, $Q_{LU2}(x)$, $Q_{RL2}(x)$

$$\begin{cases} P_{LU1}(x) = r_1(x, x) \\ P_{RL1}(x) = r_1(N-1, N-1) - r_1(x, N-1) \\ \quad - r_1(N-1, x) + r_1(x, x) \end{cases} \quad (21)$$

$$P_{LU2}(x) = \begin{cases} r_1(0, 0), & \text{if } x = 0 \\ r_1(x, x) - r_1(x-1, x-1), & \text{otherwise} \end{cases}$$

$$P_{RL2}(x) = \begin{cases} r_1(x, N-1) + r_1(N-1, x) - r_1(x, x) \\ \quad - (r_1(x-1, N-1) + r_1(N-1, x-1)) \\ \quad - r_1(x-1, x-1), & \text{otherwise} \end{cases} \quad (22)$$

$$\begin{cases} Q_{LU1}(x) = l_1(x, x) \\ Q_{RL1}(x) = l_1(N-1, N-1) - l_1(x, N-1) \\ \quad - l_1(N-1, x) + l_1(x, x) \end{cases} \quad (23)$$

$$Q_{LU2}(x) = \begin{cases} l_1(0, 0), & \text{if } x = 0 \\ l_1(x, x) - l_1(x-1, x-1), & \text{otherwise} \end{cases}$$

$$Q_{RL2}(x) = \begin{cases} l_1(x, N-1) + l_1(N-1, x) - l_1(x, x) \\ \quad - (l_1(x-1, N-1) + l_1(N-1, x-1)) \\ \quad - l_1(x-1, x-1), & \text{otherwise} \end{cases} \quad (24)$$

综上所述, 本文提出的二维模糊熵门限快速算法分为以下两个步骤:

(1) 在进行全局搜索之前, 根据式 (19) 和式 (20) 分别计算矩阵 C 和 Q 的积分图像, 并根据式 (21)~(24) 分别计算 $P_{LU1}(x), P_{RL1}(x), P_{LU2}(x), P_{RL2}(x), Q_{LU1}(x), Q_{RL1}(x), Q_{LU2}(x), Q_{RL2}(x)$, 将它们保存在一维数组 $P_{LU1}, P_{RL1}, P_{LU2}, P_{RL2}, Q_{LU1}, Q_{RL1}, Q_{LU2}, Q_{RL2}$ 中.

(2) 在计算每一个点 (a, b) 的时候, 从一维数组 $P_{LU1}, P_{RL1}, P_{LU2}, P_{RL2}, Q_{LU1}, Q_{RL1}, Q_{LU2}, Q_{RL2}$ 中获取 $P_{LU1}(x), P_{RL1}(x), P_{LU2}(x), P_{RL2}(x), Q_{LU1}(x), Q_{RL1}(x), Q_{LU2}(x), Q_{RL2}(x)$, 根据式 (17) 和 (18) 计算 $H(R_B)$ 和 $H(R_O)$.

从理论上分析, 传统二维模糊熵门限算法需要 $N^2(N-1)^2/2$ 次加法、减法、除法、对数运算, 以及 $N^2(N-1)^2$ 次乘法运算, 而本文算法的第一步需要 $N(2N+1)$ 次加法, $6N$ 次减法, N^2 次乘法, 0 次除法和 N^2 次对数运算, 第二步需要 $N(N-1)(N-2)/3$ 次加法和减法, $2N(N-1)(N-2)/3$ 次乘法, $N(N-1)$ 次除法和对数运算. 从而将传统二维模糊熵门限算法所需各种运算操作次数从 $O(N^4)$ 降低到小于或等于 $O(N^3)$. 在奔腾 4 3. 2G 微处理器上, 用 VC++ 语言验证了二维模糊熵门限快速算法的性能, 与传统二维模糊熵门限算法相比, 该快速算法可以节省大约 99% 的计算时间. 比较结果 (毫秒级) 见表 1.

表 1 快速二维模糊熵门限算法与传统二维模糊熵门限算法计算时间的比较(ms)

图像 ($W \times H$)	快速二维模糊熵门限求解算法			原始二维模糊熵门限求解算法 法总体时间
	第一步 计算时间	第二步 计算时间	总体 时间	
Lena(256× 256)	16	125	141	22813
Miss(256× 256)	15	141	156	40140
Couple(256× 256)	16	156	172	31187
Desert(409× 259)	16	156	172	70344

4 基于二维模糊熵准则的运动检测

运动检测可以看作是一个分类问题, 也就是如何划分灰度值变化较大的和灰度值变化较小的像素点. 那么, 从理论上讲, 如果将相邻两帧帧差的绝对值看作是输入图像 (此时, 帧差绝对值 FD 相当于 $I(x, y)$), 可以采用二维模糊熵准则来确定划分灰度值变化较大和较小的像素点的门限值 $T^{[6]}$. 但经过研究发现, 当目标颜色值和背景颜色值非常接近的时候, 采用这种方式进行门限定位有时是不准确的, 从而造成无法提取完整的运动目标信息. 究其原因是因为这种方法只是简单地将运动检测看作是一个划分灰度值变化较大和较小的像素点问题, 而没有考虑到灰度值变化较大的像素点有可能包含目标灰度值大于背景灰度值的像素点和目标灰

度值小于背景灰度值的像素点两个部分. 本文将像素点细化分为帧差值大于 0 和小于 0 的像素点, 并分别采用二维模糊熵准则进行检测门限的精确定位, 其本质也就是将大于 0 和小于 0 的像素点分别看作是一个二值划分问题. 相邻两帧的帧差表示为

$$FD^+(x, y, t) = \begin{cases} I(x, y, t) - I(x, y, t-1), & \text{if } (I(x, y, t) - I(x, y, t-1)) \geq 0 \\ 0, & \text{otherwise} \end{cases}$$

$$FD^-(x, y, t) = \begin{cases} I(x, y, t-1) - I(x, y, t), & \text{if } (I(x, y, t) - I(x, y, t-1)) \leq 0 \\ 0, & \text{otherwise} \end{cases}$$

(25)

其中, $FD^+(x, y, t)$ 和 $FD^-(x, y, t)$ 分别表示帧差值大于 0 和小于 0 的帧差. 分别将 $FD^+(x, y, t)$ 和 $FD^-(x, y, t)$ 看作是输入图像, 采用二维模糊熵准则进行门限的选取, 可以分别得到 T^+ 和 T^- , 那么运动区域可以通过下式进行检测

$$FDM^+(x, y, t) = \begin{cases} 1, & \text{if } FD^+(x, y, t) > T \wedge \overline{FD^+}(x, y, t) > T \\ 0, & \text{otherwise} \end{cases}$$

$$FDM^-(x, y, t) = \begin{cases} 1, & \text{if } FD^-(x, y, t) > T \wedge \overline{FD^-}(x, y, t) > T \\ 0, & \text{otherwise} \end{cases}$$

(26)

$$FDM(x, y, t) = FDM^+(x, y, t) \vee FDM^-(x, y, t)$$
 (27)

其中, $T = \min(T^+, T^-)$, FDM^+ 和 FDM^- 分别表示采用帧差值大于 0 和小于 0 的帧差得到的运动检测结果, FDM 表示最终的运动检测结果. 如果是彩色图像, 则对各个颜色分量分别进行上述操作, 确定门限 $\bar{T} = (T_R, T_G, T_B)$, 并进行像素点的划分. 同时, 当某一空间位置像素点的帧差值大于一定门限 T_{max} 时, 可以认定其一定属于目标像素点, 二维模糊熵门限的搜索范围可以限定在 $[0, T_{max}]$ 区间.

5 实验结果与分析

为了验证本文算法, 在奔腾 4 3. 2G 微处理器上, 用 VC++ 语言对室内场景的“Laboratory”、“IntelligentRoom”和室外场景的“Highway”、“Campus”等标准彩色图像序列进行了测试, 并分析了算法的运算时间及门限定位精度, T_{max} 取为 100.

图 1 和图 2 是对“Laboratory”图像序列分别根据帧差绝对值 FD 进行门限选取的检测结果和采用本文方法确定门限得到的运动检测结果. 通过比较图 1 和图 2 的运动检测结果可以看出, 由于目标部分区域的颜色值和背景的颜色值非常接近, 根据帧差绝对值 FD 进行门限选取会将目标部分区域的像素点当作噪声滤除掉了. 而根据本文算法进行门限定位, 能够完整地提取运动信息, 包括目标投射在背景上的阴影. 相对于图 1 的运动

检测结果,图 2 的运动检测结果有少许的噪声点,这是由于对应于噪声像素点和背景像素点的分布存在严重的迭加而造成的,可通过形态学滤波等后处理消除这些噪声点.图 3 为“Laboratory”所有图像帧根据 FD 求解出的门限和根据本文算法求解出的门限.对于特定的图像序列而言,其图像分量和变量特性基本是恒定的,那么由于噪声的影响,门限应该在比较小的范围内波动.而通过图 3 可以看出,根据 FD 求解出的门限矢量非常不稳定,而根据本文算法求解出的门限矢量集中在一定区域范围内,门限定位比较稳定.

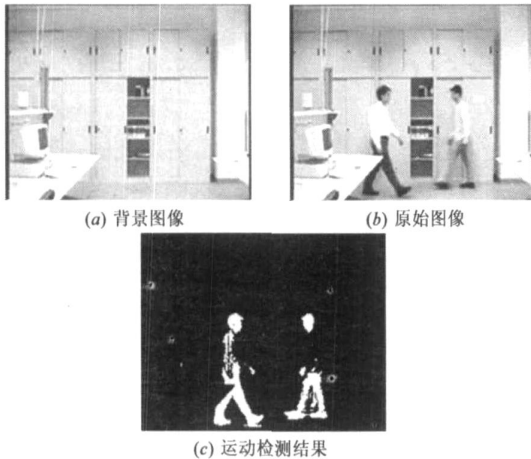


图 1 根据帧差绝对值 FD 进行二维模糊熵门限选取的检测结果

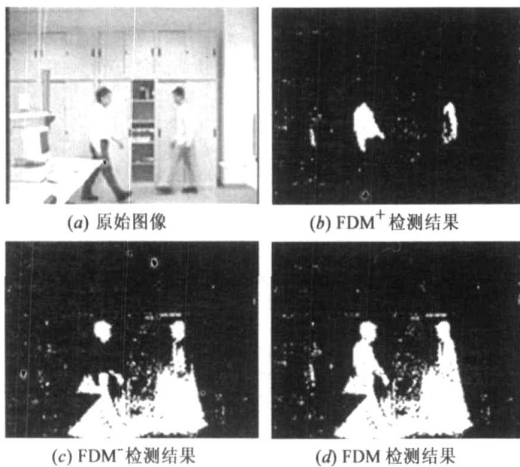


图 2 本文算法进行二维模糊熵门限选取的检测结果

表 2 列出了各种图像序列的实验结果,包括本文算法计算门限所需的平均时间以及根据 FD 求解出的门限矢量和根据本文算法求解出的门限矢量的稳定性比较结果.由于对于特定的图像序列而言,其图像分量和变量特性基本是恒定的,门限应该在比较小的范围内波动,可以根据计算出来的各个颜色分量门限矢量的方差进行稳定性判别.根据表 2 可以看出,根据 FD 进行二维模糊熵门限选取的方法与目标和背景的对比度密切相关,如果目标颜色值和背景颜色值非常接近,那么采用该方法确定门限会出现严重偏差,见表 2“Laboratory”实验结果.随着目标和背景对比度的增加,采用该方法对门限定位的精度会逐渐提高.而采用本文的方法对测试的各个序列都可以获得比较稳定的性能,并且计算时间可以实现实时处理.

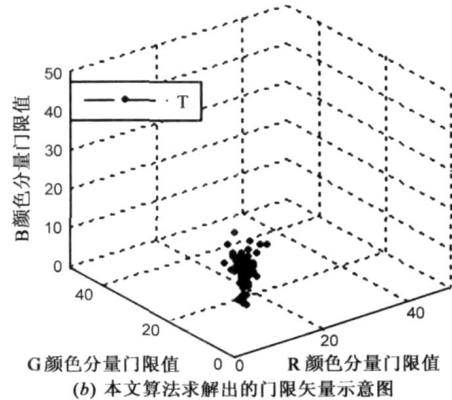
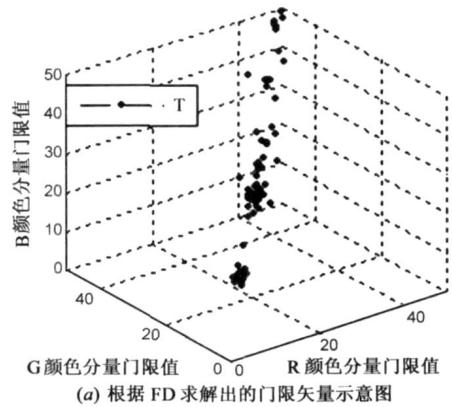


图 3 根据 FD 进行门限定位与本文算法进行门限定位的稳定性比较

表 2 各种图像序列的门限计算时间(ms)及门限定位精度比较

图像序列	Laboratory	IntelligentRoom	Highway	Campus
图像大小(宽×高)	320×240	320×240	320×240	352×288
图像帧数	887	300	500	1167
目标和背景对比度	低	高	中	高
门限平均计算时间	82.32	67.09	91.67	83.03
FD 方法的门限方差	$\sigma_T = (9.63, 9.35, 9.56)$	$\sigma_T = (3.39, 3.31, 3.20)$	$\sigma_T = (4.71, 5.02, 4.79)$	$\sigma_T = (1.97, 1.71, 1.35)$
本文方法的门限方差	$\sigma_T = (2.51, 3.03, 2.7)$	$\sigma_T = (1.13, 1.07, 1.03)$	$\sigma_T = (2.68, 2.84, 2.64)$	$\sigma_T = (1.52, 1.46, 1.20)$

6 结束语

本文将传统的运动检测看作是二个二值划分问题, 针对帧差值大于 0 和小于 0 的像素点分别采用二维时空模糊熵准则进行检测门限矢量的精确定位, 解决了目标和背景对比度偏低的条件下由于门限定位不准而造成的将目标像素点划分为背景像素点的问题. 根据推导出二维模糊熵门限的快速实现形式, 可以利用积分和迭代操作避免传统二维模糊熵门限求解过程中点的重复计算, 将灰度级为 N 的图像的各种运算操作次数从 $O(N^4)$ 降低到小于或等于 $O(N^3)$, 实现运动检测的实时处理. 通过实验验证了本文算法的有效性.

目前, 该算法还没有解决投射阴影对提取运动目标所造成的影响, 这是下一步要研究和解决的问题.

参考文献:

- [1] Weiming Hu, Xuejuan Xiao, Xie D, Tieniu Tan, Maybank S. A survey on visual surveillance of object motion and behaviors [J]. IEEE Trans on Systems, Man and Cybernetics, 2004, 33(3): 677- 694.
- [2] Stauffer C, Grimson W E L. Learning patterns of activity using real time tracking[J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2000, 22(8): 747- 757.
- [3] Paragios N, Deriche R. Geodesic active contours and level sets for the detection and tracking of moving objects [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2000, 22(3): 266- 280.
- [4] Rosin P L, Ioannidis E. Evaluation of global image thresholding for change detection [J]. Pattern Recognition Letters, 2003, 24(14): 2345- 2356.
- [5] 刘洋, 李玉山. 基于 2D 时空熵门限的运动目标检测 [J]. 电子与信息学报, 2005, 27(1): 39- 42.
Liu Yang, Li Yu Shan. The moving object detection based on 2D spatio-temporal entropic thresholding [J]. Journal of Electronics and Information Technology, 2005, 27(1): 39- 42. (in Chinese)
- [6] Velloso M L F, Carneiro T A, de Souza F J. Unsupervised change detection using fuzzy entropy principle [A]. Proceedings

of IEEE IGARSS' 2004 [C]. Anchorage: IEEE, 2004. 2550- 2553.

- [7] Pal N R, Pal S K. Object background segmentation using new definition of entropy [J]. IEE Proceedings E, 1989, 136(4): 284- 295.
- [8] Zadeh L A. Probability measures of fuzzy events [J]. Journal of Mathematical Analysis and Applications, 1968, 23(2): 421- 427.
- [9] Cheng H D, Chen J R, Li J. Threshold selection based on fuzzy c partition entropy approach [J]. Pattern Recognition, 1998, 31(7): 857- 870.
- [10] Cheng H D, Chen Y H, Jiang X H. Thresholding using two-dimensional histogram and fuzzy entropy principle [J]. IEEE Trans on Image Processing, 2000, 9(4): 732- 735.

作者简介:



刘 洋 男, 1980 年 6 月出生于安徽省灵璧县, 现为西安电子科技大学 CAD 研究所讲师. 主要从事计算机运动视觉算法与实现等方面的研究. 在国内外发表学术论文 10 余篇.
E-mail: liuyang@mail.xidian.edu.cn



初秀琴 女, 1972 年出生于山东莱州, 现为西安电子科技大学副教授. 主要从事计算机视觉、图像处理体系结构以及电路与系统设计等方面的研究工作.
E-mail: xqchu@mail.xidian.edu.cn



李玉山 男, 1945 年出生于河南省洛阳, 现为西安电子科技大学教授、国家级重点学科电路与系统学科带头人和博士生导师. 多年来获省部级奖励 10 项; 出版教材和专著 8 部; 主持制定标准 3 部; 发表论文 100 多篇, 其中 63 篇收入三大检索. 研究方向为: 高速互连设计及信号完整性分析、数字视觉技术与系统集成设计、电子 EDA 技术及软件等. E-mail: yshli@xidian.edu.cn