

片段-自适应的监控视频浓缩

张云佐, 朱鹏飞, 郑婷婷, 李文博, 张天

(石家庄铁道大学信息科学与技术学院 石家庄 050043)
(zyz2016@stdu.edu.cn)

摘 要: 为解决现有视频浓缩在复杂场景下存在的轨迹不完整、交互行为难以保留等问题, 提出一种片段-自适应的监控视频浓缩方法. 首先, 提出一种视频划分模块, 对输入视频逐帧进行拥挤度检测, 采用自适应阈值将视频划分为稀疏和拥挤片段, 并将被中断的轨迹进行链接形成延展拥挤片段; 其次, 设计交互行为判别模块, 结合空间距离和视频自适应阈值判别并保留目标之间的交互行为; 最后, 提出片段-自适应重排模块, 综合碰撞约束、空间占比约束、交互约束和时序约束生成最优时间标签, 进而融合背景生成浓缩视频. 在数据集 VISOR, BEHAVE 和 CAVIAR 上的实验结果表明, 该方法比当前主流方法在帧压缩率和碰撞率上分别降低了 0.136 和 0.011, 计算时间减少了 120.03s.

关键词: 视频浓缩; 监控视频; 片段; 自适应; 交互行为

中图法分类号: TP391.4 **DOI:** 10.3724/SP.J.1089.2023.19520

Segmentation-Adaptive Surveillance Video Synopsis

Zhang Yunzuo, Zhu Pengfei, Zheng Tingting, Li Wenbo, and Zhang Tian

(School of Information Science and Technology, Shijiazhuang Tiedao University, Shijiazhuang 050043)

Abstract: To solve the problems of existing video synopsis methods such as incomplete tracks and difficulty to retain interactive behaviors in complex scenes, a segmentation-adaptive video synopsis method is proposed. Firstly, a video segmentation module is proposed, which detects the crowding degree of each frame of the input video, and divides the video into sparse and crowded segmentation using self-adaption threshold, and links the interrupted track to form extending crowded segmentation. Secondly, an interactive behavior judgment module is designed, which combines spatial distance and video self-adaption threshold to comprehensively judge and retain interactive behavior between objects. Finally, a segmentation-adaptive rearrangement module is proposed, which combines collision constraints, space proportion constraint, interactive constraints and temporal constraints to generate the optimal time tag, and fuses the background to generate synopsis video. Experimental results on public dataset VISOR, BEHAVE and CAVIAR show that compared with the current mainstream methods, the proposed method reduces 0.136 and 0.011 frame compression rate and collision rate respectively, and the time cost is reduced by 120.03s.

Key words: video synopsis; surveillance video; segmentation; self-adaption; interactive behavior

收稿日期: 2021-12-08; 修回日期: 2022-10-10. 基金项目: 国家自然科学基金(61702347, 62027801); 河北省自然科学基金(F2022210007, F2017210161); 河北省高等学校科学技术研究项目(ZD2022100, QN2017132); 中央引导地方科技发展资金(226Z0501G); 石家庄铁道大学研究生创新资助项目(YC2023081). 张云佐(1984—), 男, 博士, 副教授, 博士生导师, CCF 高级会员, 主要研究方向为计算机视觉、人工智能、大数据; 朱鹏飞(2000—), 男, 硕士研究生, 主要研究方向为视频浓缩、计算机视觉; 郑婷婷(1997—), 女, 硕士研究生, 主要研究方向为图像与视频处理、视频浓缩; 李文博(1996—), 男, 硕士研究生, 主要研究方向为计算机视觉、数字图像处理; 张天(1996—), 男, 硕士研究生, 主要研究方向为计算机视觉、数字图像处理.

随着数字视频技术的广泛应用和智慧城市快速发展,人们对监控视频的应用需求逐渐增加,如何有效地管理监控视频信息成为当前研究的难点和热点.以简短表达监控视频内容为目的的视频浓缩技术,受到了人们的广泛关注^[1-4].

视频浓缩,又被称为基于目标的视频摘要^[5-7]技术.它从输入视频中提取背景图像,并通过目标检测和跟踪技术提取目标轨迹^[8],将其优化重组后得到新的时间标签;之后,按照此标签将目标轨迹与背景进行融合,进而生成浓缩视频.目前相关工作基本是通过构建处理单元和添加约束条件等方法来实现的.

处理单元可以分为单目标和多目标^[9].其中,以单目标作为处理单元的方法^[8,10-14]起源较早,它们大多是将目标的轨迹用“管”来表示;而视频浓缩一般采用求解能量函数的方法来生成浓缩视频.为了加快能量函数的求解速度,Moussa等^[15]提出粒子群算法求解能量函数.Feng等^[16]受俄罗斯方块游戏的启发,通过逐个填充目标管得到最后的浓缩视频.虽然以单目标为处理单元的方法已经发展得较为成熟,但是在优化过程中会丢失目标之间的交互行为.为了保留交互行为,Fu等^[17]将运动结构加入能量函数,但是能量函数受系数影响会难以精准地保留交互行为.Lu等^[18]综合碰撞约束、交互约束和时序约束,利用视频编辑技术得到一个能够保留交互行为的浓缩视频,但是该方法在复杂场景下的效果不够好,并且没有摆脱对单目标处理的限制.

而以多目标作为处理单元的方法^[19-22]在保留目标之间的交互行为上能够得到更好的结果,但是现有方法不能准确地提取多目标单元.Li等^[20]结合空间距离和固定阈值来判断目标之间的交互行为,但因未考虑目标的移动方向以及使用固定阈值,当目标的移动方向有较大的变化时,判断结果较差.在此基础上,Namitha等^[22]采用最小空间距离的方法来判断交互行为,其未考虑交互时长,对一些接触时间短的目标可能会判断错误.因此,保留交互行为的视频浓缩方法仍有较大发展空间.

为了提升浓缩视频的视觉效果,研究人员会在优化过程中添加各种各样的约束条件.常用的有碰撞约束^[10,12-14]、交互约束^[17]和时序约束^[23-27]等.其中,碰撞约束用于避免平移目标管所产生的假碰撞,以获得更加流畅的浓缩视频;交互约束用于保留目标之间的交互行为;时序约束用于保持目标出现的时间顺序.在现实中,为了兼顾用户的

观感体验,需要控制视频的目标数量,即空间占比约束.无论是着重考虑某个约束条件还是综合考虑多个约束条件,通常在场景稀疏下会获得不错的浓缩效果.然而,在真实场景中,部分监控视频既有拥挤活动,也有稀疏活动,且拥挤程度是实时变化的.Li等^[20]提出了群划分算法和基于群的贪心算法用来解决复杂场景的监控视频浓缩,其重点在于交互行为的判定,而忽略了对视频拥挤程度的判断.当面对复杂场景的监控视频时,目标遮挡以及现有预处理方法的局限,会导致出现目标轨迹断裂或者丢失的情况.

针对现有视频浓缩方法存在的问题,本文提出了片段-自适应的监控视频浓缩方法.不同于传统方法,它将复杂场景和稀疏场景分开处理,并且保留稀疏场景目标之间的交互行为,通过对片段进行重排来生成浓缩视频.

1 本文方法

为了准确地提取出目标的运动轨迹,本文首先采用YOLOv4算法^[28]对输入视频进行目标检测;其次,利用卡尔曼滤波器^[29]和匈牙利算法^[30]进行目标跟踪和轨迹匹配,获得移动目标的基本信息;最后,提出视频划分模块、交互行为判断模块和重排模块来生成浓缩视频.本文方法框架如图1所示.

1.1 视频划分模块

视频划分模块分为2个部分:对视频逐帧地进行拥挤度检测,结合自适应阈值划分为拥挤片段和稀疏片段;将被中断的目标轨迹进行连接处理来延展拥挤片段.

1.1.1 拥挤和稀疏片段划分

用 I 表示长度为 L_I 的输入视频,用 $I(x,y,t)$ 表示在视频第 t 帧图像中坐标为 (x,y) 处的像素值, $t \in \{1,2,\dots,L_I\}$,目标管集合表示为 $T = \{T_1, T_2, \dots, T_o\}$,用 o 表示所有目标管的个数.本文对输入视频的每一帧进行拥挤检测,得到稀疏标签 t_s 和拥挤标签 t_c ,进而得到稀疏片段和拥挤片段.稀疏片段和拥挤片段分别是指检测结果被连续标记为稀疏或拥挤的连续帧.为此,本文提出一种逐帧检测拥挤度,并且采用不同输入视频选择不同阈值设置标签的方法.

本文依据2个因素设置视频帧标签,分别是前景像素占帧像素的比例和视频帧中目标的个数.由于目标距离监控摄像头的远近不同,因此不能



图 1 本文方法框架

采用统一的阈值与前景像素占帧像素的比例和视频帧中目标的个数进行比较. 本文根据输入的视频生成自适应阈值 θ_1 , θ_2 . 将监控区域划分为上、中上、中、中下和下 5 部分, 考虑每一个目标管所占区域的总数, 最后自适应地选择合适的阈值, 定义为

$$H_{\text{all}} = a(h(T_1), \dots, h(T_o)) \quad (1)$$

$$S_{\text{all}} = a(s(T_1), \dots, s(T_o)) \quad (2)$$

$$\theta_1, \theta_2 = \begin{cases} \frac{20 \times S_{\text{all}}}{W \times H}, 9, & \text{if } H_{\text{all}} \leq \frac{H}{5} \\ \frac{15 \times S_{\text{all}}}{W \times H}, 7, & \text{if } \frac{H}{5} < H_{\text{all}} \leq \frac{2H}{5} \\ \frac{10 \times S_{\text{all}}}{W \times H}, 6, & \text{if } \frac{2H}{5} < H_{\text{all}} \leq \frac{3H}{5} \\ \frac{7 \times S_{\text{all}}}{W \times H}, 5, & \text{if } \frac{3H}{5} < H_{\text{all}} \leq \frac{4H}{5} \\ \frac{4 \times S_{\text{all}}}{W \times H}, 4, & \text{if } \frac{4H}{5} < H_{\text{all}} \leq H \end{cases} \quad (3)$$

其中, $a(\dots, \dots, \dots)$ 表示所有元素的平均值; $h(T_i)$ 表示第 i 个目标管边界框的平均高度; H_{all} 表示所有目标的平均高度; $s(T_i)$ 表示第 i 个目标管边界框的平均面积; S_{all} 表示所有目标管边界框的平均面积; W 和 H 分别表示视频帧的宽和高.

视频帧的标签定义为

$$J(f) = \begin{cases} C_c, & \text{if } R_f > \theta_1 \text{ or } F_n > \theta_2 \\ S_s, & \text{otherwise} \end{cases} \quad (4)$$

其中, $f \in I$ 表示视频帧; C_c 表示属于拥挤帧; S_s 表示稀疏帧; R_f 表示当前帧前景占帧像素的比例; F_n 表示当前帧的目标个数.

假设对输入视频检测得到稀疏片段集合 $S = \{S_1, S_2, \dots, S_p\}$ 和拥挤片段 $C = \{C_1, C_2, \dots, C_q\}$. 其中, S_i 和 C_j 分别表示第 i 个稀疏片段和第 j 个拥挤片段, $i \in \{1, 2, \dots, p\}$, $j \in \{1, 2, \dots, q\}$. 下文将分别介绍拥挤片段和稀疏片段处理过程.

1.1.2 延展拥挤片段

若对延展拥挤片段直接采用拥挤片段进行浓缩, 则在浓缩视频中同一个目标的轨迹将会多次不连续出现, 如图 2 所示. 相同颜色的椭圆对应同一目标的多条轨迹, 不利于观察者浏览.

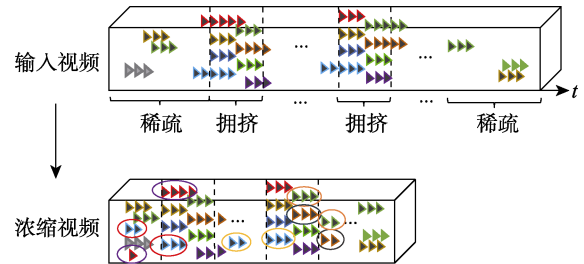


图 2 采用拥挤片段生成浓缩视频示意图

本文通过连接被中断的轨迹, 将拥挤片段扩展为延展拥挤片段. 其中, 被中断的轨迹是指同时出现在稀疏和拥挤片段中的目标轨迹. 延展拥挤片段是指正确连接拥挤片段和被中断轨迹后形成的片段, 如图 3 所示, 其中绿色椭圆区域表明轨迹被中断. 本文将每一个拥挤片段 C_j 扩展为延展拥挤片段 C_j , 得到了包含所有延展拥挤片段的集合 $C = \{C_1, C_2, \dots, C_q\}$, $j \in \{1, 2, \dots, q\}$.

1.2 交互行为判别模块

为了能够在浓缩视频中保留移动目标之间的交互行为, 采用该模块处理稀疏片段中的所有目标管 T^s_i , $i \in \{1, 2, \dots, p\}$. 受文献[20]的启发, 本文

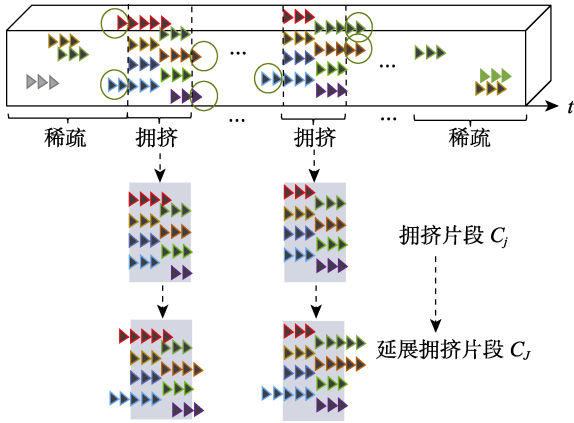


图 3 延展拥挤片段示意图

提出交互行为判别方法改进了群划分方法中空间距离的度量方式, 并采用自适应阈值而非固定阈值 37 进行判断, 同时加入了方向交互函数, 将具有交互行为的目标管精准地划分到同一个目标管集合, 作为多目标统一处理。

本文通过 $I_t(T_i, T_j)$ 来度量目标管 T_i 和 T_j 之间的交互情况, 即

$$I_t(T_i, T_j) = \begin{cases} 1, & \text{if } D(T_i, T_j) < T_v(T) \text{ and } D_F(T_i, T_j) = 1 \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

其中,

$$D(T_i, T_j) = \exp(-a(d(T_i, T_j, f))) \times \sum_{f \in T_i \cap T_j} g(d(T_i, T_j, f)) \quad (6)$$

表示 T_i 和 T_j 的时空距离; $d(T_i, T_j, f)$ 表示在第 f 帧目标管 T_i 和 T_j 的下边界框中心点之间的欧几里得距离, 其通过考虑目标边界框的高来减轻由目标距离监控摄像头远近不同所产生的影响;

$$g(d(T_i, T_j, f)) = \begin{cases} 1, & \text{if } \frac{d(T_i, T_j, f)}{a(h(T_i, f), h(T_j, f))} < w \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

$h(T_i, f)$ 和 $h(T_j, f)$ 分别表示在第 f 帧目标管 T_i 和 T_j 的边界框的高. 本文根据文献[20]设置 $w = 1.17$.

根据输入视频计算满足交互条件的自适应阈值定义为

$$T_v(T) = \exp\left(-\omega \times \frac{\sum_{i=1}^M t_i}{M} \times \text{FPS}\right) \quad (8)$$

其中, t_i 表示第 i 个管的持续时间; 每秒传输帧数 (frames per second, FPS) 表明视频每秒提供的信息

量; M 表示 T^s 中目标管的数量。

$D_F(T_i, T_j)$ 是一个方向交互函数, 用于判断 T_i 和 T_j 在方向上是否满足交互条件. 本文利用边界框的质心坐标 (x, y) 计算每个目标管的运动方向, 通过计算连续帧质心差, 可以得到物体的方向. 定义质心差计算公式为

$$\begin{cases} \Delta x = x^i - x^k \\ \Delta y = y^i - y^k \\ \text{if } |\Delta x| > 0, \text{ 方向} = \text{“东”}; \text{ else 方向} = \text{“西”} \\ \text{if } |\Delta y| > 0, \text{ 方向} = \text{“南”}; \text{ else 方向} = \text{“北”} \end{cases} \quad (9)$$

当目标运动方向保持一致时, 则 $D_F(T_i, T_j) = 1$; 否则, 考虑目标之间是否在运动发生沟通交流产生停留. 在本文中, 若目标移动方向相反但交互时长超过 1s, 则设定 $D_F(T_i, T_j) = 1$; 否则, $D_F(T_i, T_j) = 0$.

1.3 片段-自适应重排模块

通过视频划分模块, 可以将视频分为稀疏片段和拥挤片段, 再利用交互行为判别模块将稀疏片段中的目标提取成目标管集合统一处理. 在片段-自适应重排模块中, 需要对拥挤片段和交互行为保留后的稀疏片段安排新的时间标签, 本文将每一个需要重排的部分称作片段。

假定每一个片段在监控视频场景内的活动时间集合为 $T_t = \{t_1, t_2, \dots, t_{q+n}\}$, $t_i = [t_i^s, t_i^e]$. 其中, t_i^s, t_i^e 分别表示集合中所有目标在原始视频中出现时间的最小值和消失时间的最大值。

图 4 所示为片段-自适应重排模块的流程. 首先设置片段 P_1 的开始时间标签 $\hat{t}_1^s = 1$, 输入 $P = \{P_2, P_3, \dots, P_{q+n}\}$, 估计 P_1 初始可能的开始时间标签; 其次, 利用标记计算模型 (tag calculation model, TCM) 计算每一帧的标记 $d(f_{\psi_i}^k)$, 其值为 1 表示可以加入新的片段, 其值为 0 表示不可以加入新的片段; 然后, 寻找可能的时间标签 t_i , 在综合考虑碰撞约束、交互约束和时序约束的情况下, 得到 P_1 最优的开始时间标签, 并优化 TCM; 最后, 循环输入所有片段。

1.3.1 估计初始可能的时间标签

传统视频浓缩方法的基本步骤包括: 目标管提取、管优化重排和背景生成与缝合. 其中, 管优化重排是指为目标管安排新的时间标签. 本文将视频浓缩的管优化重组问题转化为求解片段在浓

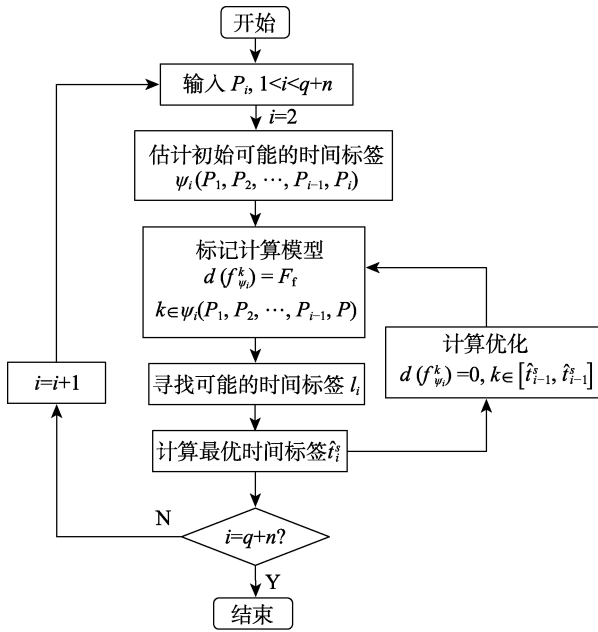


图 4 片段-自适应重排模块流程

缩视频中的时间标签集合最大后验(maximum a posteriori, MAP)概率估计问题。

已知浓缩视频的长度为 L_s , 对应浓缩视频中片段的时间标签集合定义为

$$I = \{\hat{t}_1^s, \hat{t}_1^e, \dots, \hat{t}_M^s, \hat{t}_M^e \mid \hat{t}_i = [\hat{t}_i^s, \hat{t}_i^e]\} \quad (10)$$

其中, $\hat{t}_i^s \in l_i$, $\hat{t}_i^e \in [1, L_s]$, $1 \leq i \leq q+n$. l_i 表示第 i 个片段可能的时间标签取值, 其定义为

$$l_i = \chi_i(\psi_i(P_1, P_2, \dots, P_{i-1}, P_i)) \quad (11)$$

$\psi_i(P_1, P_2, \dots, P_{i-1}, P_i)$ 是指已安排好开始时间的片段, $P_1, P_2, \dots, P_{i-1}$ 为 P_i 初始所有可能时间标签的取值. 当 $i=1$ 时, $\psi_1(P_1)=1$; 当 $i>1$ 时, 对每一个片段有

$$\psi_i(P_1, P_2, \dots, P_{i-1}, P_i) = [1, M_m = (\hat{t}_1^e, \dots, \hat{t}_{i-1}^e) + 1] \quad (12)$$

其中, $M_m(\hat{t}_1^e, \hat{t}_2^e, \dots, \hat{t}_{i-1}^e)$ 表示片段 $P_1, P_2, \dots, P_{i-1}$ 中结束帧的最大值。

1.3.2 TCM

如果直接采用 $\psi_i(P_1, P_2, \dots, P_{i-1}, P_i)$ 作为可能的时间标签集合, 那么随着片段的增多, 计算量会成倍增加. 因此, 本文在获得片段 P_i 的开始时间标签时, 会估计片段 $P_1, P_2, \dots, P_{i-1}$ 产生浓缩视频 S_{i-1} 每一帧的情况, 衡量对于 S_{i-1} 中的每一帧是否可以添加新的片段. 为了更好地满足人们的观看需求且估计帧的情况, 受文献[26]的启发, 本文提出适用于片段的 TCM, 由碰撞概率模型(collision probability model, CPM)和密度概率模型(density

probability model, DPM)组成, 其定义为

$$\text{TCM}(f_{\psi_i}^k) = \text{CPM}(f_{\psi_i}^k) \times \text{DPM}(f_{\psi_i}^k) \quad (13)$$

其中, f_{ψ_i} 表示 S_{i-1} 的所有帧; $f_{\psi_i}^k$ 表示 S_{i-1} 中的第 k 帧, 并且 $k \in \psi_i(P_1, P_2, \dots, P_i)$.

(1) CPM. 由于片段 $P_1, P_2, \dots, P_{i-1}$ 在平移过程中会产生“伪碰撞”, 为了估计 S_{i-1} 中的碰撞程度提出 $\text{CPM}(f_{\psi_i}^k)$, 其定义为

$$\text{CPM}(f_{\psi_i}^k) = \begin{cases} 1, & \text{if } \text{IoU}^k \leq 0.3 \\ 0, & \text{otherwise} \end{cases} \quad (14)$$

交并比(intersection over union, IoU)是指“预测的边框”和“真实的边框”交集与并集的比值. IoU^k 指在第 k 帧中所有片段的交集与并集的比值, 定义为

$$\text{IoU}^k = \frac{\sum_{P_v, P_w \in U^k} (\chi_{P_v}(x, y, k) \times \chi_{P_w}(x, y, k))}{\sum_{P_v = \max U^k}^{\min U^k} \chi_{P_v}(x, y, k)} \quad (15)$$

其中, U^k 表示在第 k 帧出现的片段集合。

(2) DPM. 考虑帧中出现的片段, 衡量前景目标的占比程度, 避免在帧中出现过多目标, 造成观看困难, 定义为

$$\text{DPM}(f_{\psi_i}^k) = \begin{cases} 1, & \text{if } R_s^k \leq 0.5 \\ 0, & \text{otherwise} \end{cases} \quad (16)$$

其中, R_s^k 表示第 k 帧中所有片段所占像素点与背景的比值, 定义为

$$R_s^k = \frac{\sum_{P_v \in U^k} \chi_{P_v}(x, y, k)}{H \times W} \quad (17)$$

$\chi_{P_v}(x, y, k)$ 表示第 k 帧片段 P_v 的面积; $d(f_{\psi_i}^k)$ 表示是否可以在帧 $f_{\psi_i}^k$ 中加入新的片段, 其定义为

$$d(f_{\psi_i}^k) = \begin{cases} 1, & \text{if } \text{TCM}(f_{\psi_i}^k) = 1 \\ 0, & \text{if } \text{TCM}(f_{\psi_i}^k) = 0 \end{cases} \quad (18)$$

其中, 1 和 0 分别表示可以或不可以加入新的片段。

1.3.3 寻找可能的时间标签

通过 TCM, 已经得到每一个片段 P_i 可以插入的位置, 随后采用如下方案寻找 l_i .

首先, 从小到大对每一帧进行计算得到 $d(f_{\psi_i}^k)$ 值; 然后, 对所有片段从后往前找 l_i , 分 2 种情况: 找到第 1 个 $d(f_{\psi_i}^k)=0$, 其对应的 l_i 值为 $k+1$; 没有找到 $d(f_{\psi_i}^k)=0$, 即全部为 1 的情况, 则

$$l_i = l_{i-1}.$$

1.3.4 计算最优时间标签

本文考虑了交互约束、时序约束和碰撞约束计算片段的开始时间标签. 由于采用片段作为处理单元, 已经满足了交互约束; 同时, 将片段按照其出现时间进行排序, 已满足时序约束. TCM中也包含空间占比约束. 通过以下公式进一步优化碰撞约束, 从 l_i 中找到 P_i 最优的时间标签.

$$\hat{t}_i^s = \begin{cases} p, & \text{if } \frac{\sum_{p \in l_i} C_{\text{IOU}}^p(S_{i-1}, P_i)}{L(l_i)} \leq \varsigma \\ \max(l_i) + 1, & \text{otherwise} \end{cases} \quad (19)$$

其中, $C_{\text{IOU}}^p(S_{i-1}, P_i)$ 表示从 p 帧开始将 P_i 插入 S_{i-1} 中所产生的边界框重叠面积和所有边界框面积和的比值, 定义为

$$C_{\text{IOU}}^p(S_{i-1}, P_i) = \frac{s(\xi_{S_{i-1}}^p) \cap s(\xi_{P_i}^p)}{s(\xi_{S_{i-1}}^p) \cup s(\xi_{P_i}^p)} \quad (20)$$

$\xi_{S_{i-1}}^p$ 和 $\xi_{P_i}^p$ 分别表示 S_{i-1} 和片段 P_i 中的边界框信息, 如果片段为延展拥挤片段, 那么考虑被中断轨迹目标的边界框信息; $s(\xi_{S_{i-1}}^p) \cap s(\xi_{P_i}^p)$ 表示将 P_i 插入 S_{i-1} 时第 p 帧的所有片段边界框的交集; $s(\xi_{S_{i-1}}^p) \cup s(\xi_{P_i}^p)$ 表示将 P_i 插入 S_{i-1} 时第 p 帧的所有片段边界框的并集. 当每个 $p \in l_i$ 均不满足条件时, 则将 P_i 插入 S_{i-1} 的末尾. 为了保证生成的浓缩视频能够较好地平衡视频长度和产生的碰撞, 本文设定碰撞阈值 $\varsigma = 0.1$.

1.3.5 计算优化

为了避免重复计算, 提高计算速度, 本文对计算方法进行了优化. 通过得到的每一个片段 P_i 的 $\hat{t}_i^s$, 设置

$$d(f_{\psi_i}^k) = 0, k \in [\hat{t}_{i-1}^s, \hat{t}_i^s - 1] \quad (21)$$

考虑 P_{i+1} 的时间标签时, 将不再计算相应的 $d(f_{\psi_{i+1}}^k)$, $k \in [1, \hat{t}_i^s - 1]$.

最后, 将片段和背景进行拼接, 得到浓缩视频. 现有的视频浓缩方法在生成背景时通常使用时间中值法、高斯混合模型(Gaussian mixture model, GMM)和视觉背景提取器等方法. 对比各种方法的计算效率, 本文采用时间中值法来估计监控视频的背景, 利用泊松图像编辑技术将片段的最优时间标签依次拼接到生成的背景中, 得到

浓缩视频.

2 实验结果与分析

为了全面评估本文方法的有效性和普适性, 本文进行了大量实验. 实验测试的视频来自公开数据集 VISOR, BEHAVE 和 CAVIAR, 其中包括商场、交叉路口、餐厅进出口和道路等不同的场景. 对比方法选取将求解目标管最优标签当作求解能量函数最优解的优化方法^[15]、考虑目标交互性的方法^[20,22]和将事件重排问题归结为对轨迹相关性和事件紧凑性的方法^[26], 将这些方法分别从客观和主观2个方面与本文方法进行对比. 选取的方法是交互行为保留方面具有较大影响力、较新颖的方法, 同时客观视觉质量评价的对比方法也是近年来影响力较大的方法. 与这些方法进行对比, 能有效地检验本文方法的有效性.

2.1 客观评价

为了验证所提交交互行为判断方法的准确性, 本文与群划分方法^[20]、递归群方法^[22]进行对比, 并采用准确率 p 、召回率 r 和 F 分数作为评价指标, 分别定义为

$$\begin{cases} p = \frac{\text{TP}}{\text{TP} + \text{FP}} \\ r = \frac{\text{TP}}{\text{TP} + \text{FN}} \\ F = \frac{2 \times p \times r}{p + r} \end{cases} \quad (22)$$

其中, 真正例(true positive, TP)表示在原始视频中具有交互行为的目标对, 其检测结果是交互的; 假正例(false positive, FP)表示在原始视频中不具有交互行为的目标对, 其检测结果是交互的; 假负例(false negative, FN)表示在原始视频中具有交互行为的目标对, 其检测结果不是交互的.

表1所示为本文方法与文献[20]、文献[22]方法的对比实验结果, 表中实验数据为所有测试视频结果的平均值.

从表1可以看出, 本文方法能够获得更高的 p

表1 3种算法对比结果

方法	p	r	F
文献[20]	0.326	0.604	0.423
文献[22]	0.256	0.813	0.389
本文	0.782	0.562	0.653

注: 粗体表示最优值.

和 F , 分别是 0.782 和 0.653. 即在浓缩视频中, 本文方法能够更加准确地保留移动目标之间的交互行为.

目前, 针对监控视频的浓缩方法没有统一的评价指标. 但是现有方法通常是考虑浓缩视频的长度、目标之间的假碰撞、目标之间的时序关系和生成浓缩视频的速度. 因此, 本文采用帧压缩率 (frame compression ratio, FCR)、碰撞率 (overlap ratio, OR)、运行时间 (running time, RT) 和时间复杂度来全面、客观地评估本文方法, 上述 4 个参数均是越小越好.

图 5 所示为上述 3 种方法和本文方法的实验结果, 选取典型场景下的视频 ($V_1 \sim V_{10}$) 进行对比.

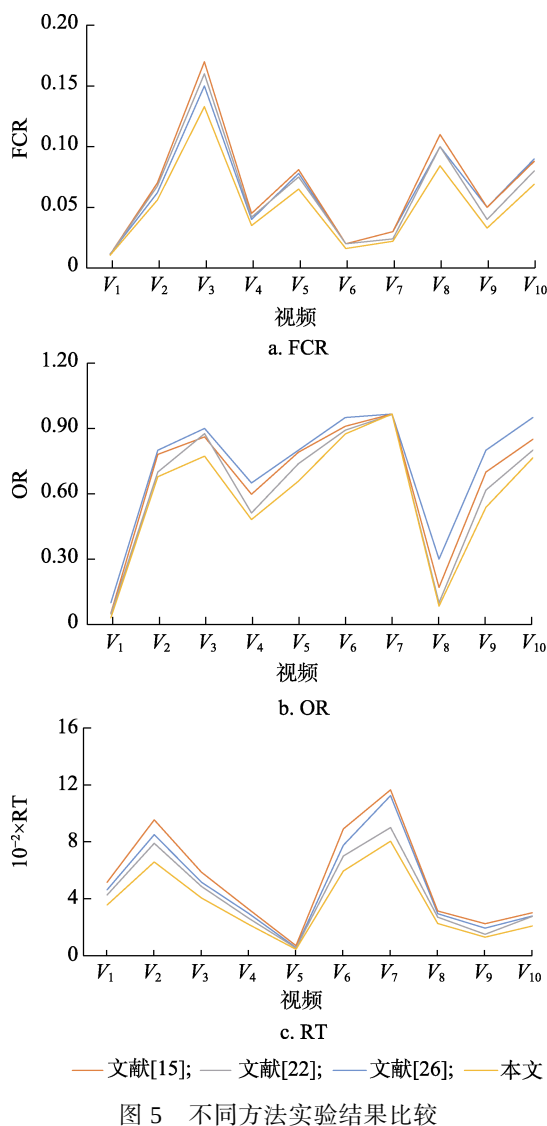


图 5 不同方法实验结果比较

V_4 是一段包含拥挤片段的监控视频, 虽然本文方法和对比方法碰撞率差别较小, 如图 5b 所示, 但是本文方法的帧压缩率小于对比方法, 如图 5a 所示, 并且运行时间明显降低. V_7 是一段比较拥挤

的视频, 从图 5a 可以看出, 本文方法和对比方法的帧压缩率差别较小, 但从图 5b 可以看出, 本文方法的碰撞率均小于对比方法, 且从图 5c 可以看出, 本文方法的运行时间明显少于对比方法. 从图 5a~图 5c 可以看出, 与其他方法相比, 本文方法的帧压缩率、碰撞率和运行时间均有所降低.

本文采用 3 个模块生成浓缩视频, 记原视频帧数为 F , 目标数量为 N , 片段数量为 R . 其中, $R < N$. 3 个模块是平级不存在嵌套的, 最优情况下的时间复杂度为 $O(\max(F, R, N))$, 最差情况下的时间复杂度为 $O(\max(F, R, N^2))$. 传统方法的时间复杂度一般为 $O(\max(F, N^2))$, 如文献[16,23]方法等. 采用立方体投票方法^[19]的时间复杂度为 $O(\max(F, N^3))$. 本文方法的计算效率更高.

综合以上实验结果可以看出, 本文方法在 FCR, OR 和 RT 上明显优于其他方法.

2.2 主观评价

图 6 和图 7 分别展示了一段稀疏的监控视频和一段包含拥挤片段的监控视频, 其中, 蓝色矩形框代表在原始视频中需要保留的交互行为或拥挤信息, 绿色矩形框代表成功保留. 图 6a 中 #6996 和图 6b 中 #7753 代表输入视频的帧号, 图 6c 中 #1004 代表浓缩视频的帧号. 如图 6c 所示, 浓缩视频中的交互行为被完全保留.

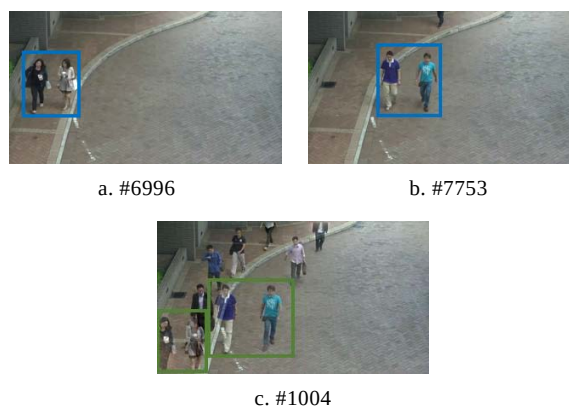


图 6 交互性分析

图 7 中, #152 和 #875 来自输入视频, #139 和 #456 来自浓缩视频, 经过自适应拥挤帧检测方法得到 #152 属于拥挤片段, #875 属于稀疏片段. 如图 7a 所示, 本文方法能够保留 #152 中的拥挤信息, 即 #139. 如图 7b 所示, 对稀疏片段进行浓缩, 图中 #875 中的目标出现在 #456 中.

从浓缩视频的可视化结果可以看出, 本文方

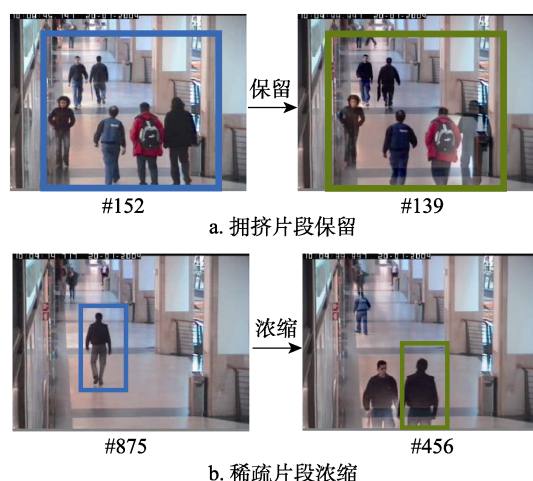


图 7 拥挤和稀疏片段处理

法生成的浓缩视频具有良好的视觉感知, 并且在浓缩视频中未发生目标重复出现的情况, 说明目标轨迹未发生中断。

浓缩视频的主观评价采用人工主观评估的方法, 连续观看原视频、对比方法生成的浓缩视频以及本文方法生成的浓缩视频, 采用平均意见得分 (mean opinion score, MOS) 作为主观评价指标。采用主观评价得到的评估结果相较于客观评价更加准确, 更接近用户的体验感知。

为了准确评估本文方法, 本实验邀请了 50 位来自不同专业的用户进行打分。要求用户按照表 2 所提的问题进行评估, 按照 1(情况最坏)~5(情况最好)的范围进行打分, 最后得到本文方法和对比方法的平均意见得分。本文方法和对比方法生成的浓缩视频的主观反馈平均意见得分如图 8 所示。

表 2 主观评估问题描述

问题编号	问题描述
1	你认为浓缩视频“保留了目标之间交互行为”吗
2	你认为浓缩视频目标之间“产生了不必要的碰撞”吗
3	你认为浓缩视频的目标“有重复出现”吗

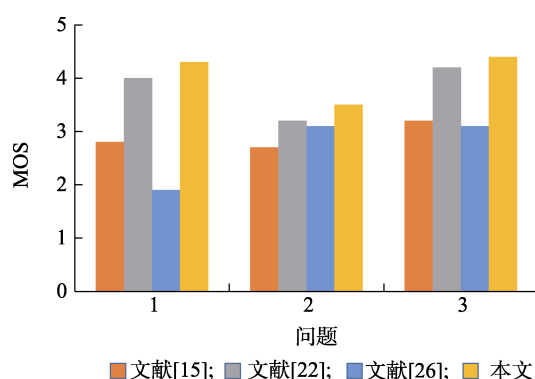


图 8 不同方法主观反馈的平均意见得分

从图 8 可以看出, 在测试视频上本文方法的 MOS 均明显高于对比方法。本文方法能够更好地保留目标之间的交互行为, 避免了目标轨迹断裂。因此, 本文方法具有更好的视觉效果。

3 结 语

本文提出了一种片段-自适应的监控视频浓缩方法。该方法首先对输入视频逐帧进行拥挤度检测, 利用自适应阈值将视频划分为稀疏和拥挤片段, 并连接被中断的轨迹; 其次, 对稀疏片段采用交互行为判断方法保留交互行为; 最后, 利用重排模块得到最优时间标签, 据此生成浓缩视频。综合客观的评价和主观的视觉质量评价, 本文方法能够应对各种复杂的视频场景, 满足人们对监控视频的浓缩需求, 通用性较强, 具有一定的实际意义。下一步将对跨摄像头场景下的监控视频浓缩方法进行探究。

参考文献(References):

- [1] Baskurt K B, Samet R. Video synopsis: a survey[J]. Computer Vision and Image Understanding, 2019, 181: 26-38
- [2] Li Xuelong, Zhao Bin. Video distillation[J]. Scientia Sinica (Informationis), 2021, 51(5): 695-734(in Chinese)
(李学龙, 赵斌. 视频萃取[J]. 中国科学: 信息科学, 2021, 51(5): 695-734)
- [3] Ghatak S, Rup S. Single camera surveillance video synopsis: a review and taxonomy[C] //Proceedings of the International Conference on Information Technology. Los Alamitos: IEEE Computer Society Press, 2020: 483-488
- [4] Wang Feiyue, Xie Yujia, Mao Bo. Multi-video synopsis in geographic scene considering virtual viewpoint area of camera[J]. Geomatics and Information Science of Wuhan University, 2021, 46(4): 595-600(in Chinese)
(王飞跃, 解愉嘉, 毛波. 顾及相机可观测域的地理场景多相机视频浓缩[J]. 武汉大学学报: 信息科学版, 2021, 46(4): 595-600)
- [5] Li Yiyi, Wang Jilong. Self-attention based video summarization[J]. Journal of Computer-Aided Design & Computer Graphics, 2020, 32(4): 652-659(in Chinese)
(李依依, 王继龙. 自注意力机制的视频摘要模型[J]. 计算机辅助设计与图形学学报, 2020, 32(4): 652-659)
- [6] Rav-Acha A, Pritch Y, Peleg S. Making a long video short: dynamic video synopsis[C] //Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2006: 435-441
- [7] Liu Yujie, Tang Shunjing, Gao Yongbiao, et al. Label distribution learning for video summarization[J]. Journal of Computer-Aided Design & Computer Graphics, 2019, 31(1):

- 104-110(in Chinese)
(刘玉杰, 唐顺静, 高永标, 等. 基于标签分布学习的视频摘要算法[J]. 计算机辅助设计与图形学学报, 2019, 31(1): 104-110)
- [8] Pritch Y, Rav-Acha A, Gutman A, *et al.* Webcam synopsis: peeking around the world[C] //Proceedings of the 11th IEEE International Conference on Computer Vision. Los Alamitos: IEEE Computer Society Press, 2007: 1-8
- [9] Namitha K, Narayanan A. Preserving interactions among moving objects in surveillance video synopsis[J]. *Multimedia Tools and Applications*, 2020, 79(43/44): 32331-32360
- [10] Nie Y W, Xiao C X, Sun H Q, *et al.* Compact video synopsis via global spatiotemporal optimization[J]. *IEEE Transactions on Visualization and Computer Graphics*, 2013, 19(10): 1664-1676
- [11] Li X L, Wang Z G, Lu X Q, *et al.* Surveillance video synopsis via scaling down objects[J]. *IEEE Transactions on Image Processing*, 2016, 25(2): 740-755
- [12] He Y, Gao C X, Sang N, *et al.* Graph coloring based surveillance video synopsis[J]. *Neurocomputing*, 2017, 225: 64-79
- [13] He Y, Qu Z G, Gao C X, *et al.* Fast online video synopsis based on potential collision graph[J]. *IEEE Signal Processing Letters*, 2017, 24(1): 22-26
- [14] Nie Y W, Li Z K, Zhang Z S, *et al.* Collision-free video synopsis incorporating object speed and size changes[J]. *IEEE Transactions on Image Processing*, 2020, 29: 1465-1478
- [15] Moussa M M, Shoitani R. Object-based video synopsis approach using particle swarm optimization[J]. *Signal, Image and Video Processing*, 2021, 15(4): 761-768
- [16] Feng S K, Lei Z, Yi D, *et al.* Online content-aware video condensation[C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2012: 2082-2087
- [17] Fu W, Wang J Q, Gui L K, *et al.* Online video synopsis of structured motion[J]. *Neurocomputing*, 2014, 135: 155-162
- [18] Lu S P, Zhang S H, Wei J, *et al.* Timeline editing of objects in video[J]. *IEEE Transactions on Visualization and Computer Graphics*, 2013, 19(7): 1218-1227
- [19] Tian Y M, Zheng H H, Chen Q C, *et al.* Surveillance video synopsis generation method via keeping important relationship among objects[J]. *IET Computer Vision*, 2016, 10(8): 868-872
- [20] Li X L, Wang Z G, Lu X Q, *et al.* Video synopsis in complex situations[J]. *IEEE Transactions on Image Processing*, 2018, 27(8): 3798-3812
- [21] Ahmed A, Kar S, Dogra D P, *et al.* Video synopsis generation using spatio-temporal groups[C] //Proceedings of the IEEE International Conference on Signal and Image Processing Applications. Los Alamitos: IEEE Computer Society Press, 2017: 512-517
- [22] Namitha K, Narayanan A, Geetha M. Interactive visualization-based surveillance video synopsis[J]. *Applied Intelligence*, 2022, 52(4): 3954-3975
- [23] Ghatak S, Rup S, Majhi B, *et al.* An improved surveillance video synopsis framework: a HSATLBO optimization approach[J]. *Multimedia Tools and Applications*, 2020, 79(7): 4429-4461
- [24] Ghatak S, Rup S, Majhi B, *et al.* HSAJAYA: an improved optimization scheme for consumer surveillance video synopsis generation[J]. *IEEE Transactions on Consumer Electronics*, 2020, 66(2):144-152
- [25] Ghatak S, Rup S, Didwania H, *et al.* GAN based efficient foreground extraction and HGWOA based optimization for video synopsis generation[J]. *Digital Signal Processing*, 2021, 111: 102988
- [26] He Y, Han J, Sang N, *et al.* Chronological video synopsis via events rearrangement optimization[J]. *Chinese Journal of Electronics*, 2018, 27(2): 399-404
- [27] Moonsoo R, Whoi-Yul K. Parallelized tube rearrangement algorithm for online video synopsis[J]. *IEEE Signal Processing Letters*, 2018, 25(8):1186-1190
- [28] Bochkovskiy A, Wang C Y, Liao H Y M. YOLOv4: optimal speed and accuracy of object detection[OL]. [2021-12-08]. <https://arxiv.org/abs/2004.10934>
- [29] Kalman R E. A new approach to linear filtering and prediction problems[J]. *Journal of Basic Engineering*, 1960, 82(1): 35-45
- [30] Kuhn H W. The Hungarian method for the assignment problem[J]. *Naval Research Logistics Quarterly*, 1955, 2(1/2): 83-97