

参赛密码 \_\_\_\_\_

(由组委会填写)



# “华为杯”第十四届中国研究生 数学建模竞赛

题 目                      多种场景下的视频前景目标提取和行为识别

## 摘                      要：

本文主要研究的是监控视频中前景目标的提取和行为识别问题。对静态背景、动态背景、运动摄像机、多角度摄像机的情景都给出了鲁棒性较强的前景检测模型。在检验视频集上表现良好。最后通过检测到的前景建立了对三种人群行为的识别模型。

针对问题一，视频不包含动态背景，摄像头稳定拍摄。考虑到背景像素的频率分布，本文给出了概率模型来建立背景模型，使用高斯分布表示背景像素的变化。通过“ $3\sigma$ 原则”判别单个像素点是否属于背景，从而对整个图像进行划分。为了提取前景的完整性，进一步使用形态学方法处理，得到前景目标。模型计算简单，判断效率较高。

针对问题二，视频为动态背景，摄像头稳定拍摄。由于背景为动态，单通道高斯模型已经不能刻画背景像素，因此使用混合高斯模型，赋予不同高斯分布不同的权重，像素点按权重大小与高斯分布进行比较，给出前景判断结果，并对背景模型进行更新。本文对传统混合高斯模型进行了改进，给出了新的背景学习速率自适应计算方法，背景建立速度更快，对光线变化等突变行为有更强的鲁棒性。

针对问题三，视频存在摄像机运动。为了消除摄像机运动带来的影响，基于摄像原理进行运动补偿。使用角点检测方法，检测对图像运动敏感的像素点，并

计算图像的光流场，根据速度矢量过滤掉变化过快的像素点，选取剩下的像素点作为特征点。使用特征点计算不同帧图像变换的单应性矩阵，利用单应性矩阵恢复图像背景，再利用高斯分布检测图像前景。所建立的模型能够一定程度上恢复背景，识别出明显的前景目标。

针对问题四，要求使用建立的模型检验不同视频。检验结果表明模型在动态背景下能够有效检测出前景目标，对于水波、窗帘、树枝等扰动有很好的抗干扰能力，而且能够应对光线的突变。总体来说具有较好的鲁棒性。当背景运动过于剧烈时仍会被误检为前景，但是误检的频率较小，不影响模型整体上对前景的识别。

针对问题五，对多角度摄像机拍摄同一地点的视频进行前景提取。本文在单应性矩阵进行空间坐标转换的基础上，结合目标之间距离、比例等相互关系在变换中存在不变性的特点，给出了目标判断模型。

针对问题六，对前景目标的行为进行识别。由于目标种类的多样性，目标形态的多样性以及目标行为的多样性，本文仅选取了具有代表性的三种人群行为：人群行走、人群奔跑，人群聚集进行识别。使用前景提取方法得到视频中的前景，通过计算前景目标的光流场，对前景目标光流场进行统计，提取出特征向量，进而利用一对多的逻辑回归或支持向量机对特征进行分类和识别。达到了识别前景行为的目的。

**关键词：**改进混合高斯模型、单应性矩阵、角点检测、光流场、逻辑回归

## 目录

一 问题背景与重述.....	1
1.1 问题的背景.....	1
1.2 问题的提出.....	1
1.3 问题的重述.....	1
二 问题的假设及符号说明.....	3
2.1 模型假设.....	3
2.2 符号说明.....	3
三 问题分析.....	4
3.1 对问题一的分析: .....	4
3.2 对问题二的分析: .....	4
3.3 对问题三的分析: .....	4
3.4 对问题四的分析: .....	4
3.5 对问题五的分析: .....	4
3.5 对问题六的分析: .....	5
四 问题一的解答.....	5
4.1 模型的准备.....	5
4.2 模型的建立.....	5
4.3 模型的求解与分析.....	6
五 问题二的解答.....	7
5.1 模型准备.....	7
5.2 模型的建立.....	7
5.3 模型的求解与分析.....	9
六 问题三的解答.....	10
6.1 模型准备.....	10
6.2 模型建立.....	11
6.3 模型的求解与分析.....	13
七 问题四的解答.....	15
八 问题五的解答.....	16
8.1 模型准备.....	16
8.2 模型的建立.....	17
九 问题六的解答.....	19
9.1 模型准备.....	19
9.2 模型建立.....	19
9.3 模型的求解与分析.....	20
十 模型的评价与改进.....	22
十一 参考文献.....	23
十二 附件.....	25

# 一 问题背景与重述

## 1.1 问题的背景

视频监控是中国安防产业中最为重要的信息获取手段。随着“平安城市”建设的顺利开展，各地普遍安装监控摄像头，利用大范围监控视频的信息，应对安防等领域存在的问题。目前，监控视频信息的自动处理与预测在信息科学、计算机视觉、机器学习、模式识别等多个领域中受到极大的关注。这一技术往往能够对一般的视频处理任务提供有效的辅助。

例如筛选与跟踪夜晚时罪犯：若能够预先提取视频前景目标，判断出哪些视频并未包含移动前景目标，并事先从公安人员的辨识范围中排除；而对于剩下包含了移动目标的视频，只需辨识排除了背景干扰的纯粹前景，对比度显著，肉眼更易辨识。因此，这一技术已被广泛应用于视频目标追踪，城市交通检测，长时场景监测，视频动作捕捉，视频压缩等应用中。

## 1.2 问题的提出

如何有效、快速抽取出监控视频中的前景目标信息是非常重要而基础的问题。这一问题的难度在于，需要有效分离出移动前景目标的视频往往具有复杂、多变、动态的背景。这一技术往往能够对一般的视频处理任务提供有效的辅助。

需解决的问题：

1. 从固定摄像头、静态背景，只有前景目标移动的理想情况出发，建立能够识别出前景和背景的数学模型，求解图像的前景和背景。
2. 建立鲁棒性更强的判别模型，在背景有轻微扰动（如水纹，树枝晃动等）时，避免将其识别为前景。
3. 当摄像头不固定，比如发生晃动或平移时，图像整体发生改变，该类视频变换在短时间内可近似视为一种线性仿射变换。针对这个情况，建立判别模型。
4. 根据不同角度同时拍摄的近似同一地点的多个监控视频，建立前景判别模型。
5. 利用获取的前景目标，识别视频中前景目标的行为。

## 1.3 问题的重述

问题 1：对一个不包含动态背景、摄像头稳定拍摄的监控视频，构造提取前景目标（如人、车、动物等）的数学模型，并对该模型设计有效的求解方法。

问题 2：对包含动态背景信息的监控视频，设计有效的前景目标提取方案。

- 问题 3: 在监控视频中, 当监控摄像头发生晃动或偏移时, 视频也会发生短暂的抖动现象。对这种类型的视频设计有效的有效的提取前景目标提取方案。
- 问题 4: 利用所构造的建模方法, 从附件 3 中提供了 8 组视频的各组视频中选出包含显著前景目标的视频帧标号。
- 问题 5: 从不同角度同时拍摄的近似同一地点的多个监控视频中有效检测和提取视频前景目标。
- 问题 6: 利用所获取前景目标信息, 自动判断监控视频中无人群众短时聚集、人群惊慌逃散、群体规律性变化、物体爆炸、建筑物倒塌等异常事件。并尝试对更多的异常事件类型, 设计相应的事件检测方案加以验证。

## 二 问题的假设及符号说明

### 2.1 模型假设

假设 1: 前景目标为运动目标。

假设 2: 摄像机拍摄的视频没有经过人为处理。

假设 3: 运动摄像头的运动速度是较缓慢且连续的。

假设 4: 多角度摄像头检测的是同一个空间。

### 2.2 符号说明

$G(x)$	高斯概率密度函数
$\mu$	高斯分布的期望值
$\sigma$	标准差
$a_k$	第 $k$ 帧视频时像素的灰度值
$T$	高斯分布阈值
$(x_0, y_0)$	图像中的像素点
$P(X_t)$	像素点 $(x_0, y_0)$ 在 $t$ 时刻观察值 $X_t$ 属于背景的概率
$K$	模型中的高斯分布个数
$\omega_{i,t}$	为 $t$ 时刻第 $i$ 个高斯分布的权值
$\mu_{i,t}$	$t$ 时刻第 $i$ 个高斯分布的均值
$\Sigma_{i,t}$	$t$ 时刻第 $i$ 个高斯分布的协方差矩阵
$\alpha$	权值的学习速率
$\beta$	均值和方差的学习速率
$V$	运动量的函数
$V_x$	像素点在 $x$ 轴速度
$V_y$	像素点在 $y$ 轴速度
$\partial I / \partial x$	图像 $I$ 相对于 $x$ 轴的光强变化
$\partial I / \partial y$	图像 $I$ 相对于 $y$ 轴的光强变化
$V_t$	光流场运动矢量阈值
$H$	单应矩阵
$I$	像素点的灰度值
$Age$	像素作为背景的帧数

### 三 问题分析

此题主要的目标是从视频中提取出前景。提取过程可分为三部分：1. 通过视频前几帧的图像得到背景模型；2. 利用得到的背景模型，在之后的图像中计算前景；3. 选择合适的更新速率更新背景模型，当前景静止时将会融入背景，更新背景模型后继续识别前景。

#### 3.1 对问题一的分析：

针对问题一，摄像头固定，背景几乎不变。考虑每个像素点的变化。所有帧内，单个像素点的变化是很小的，仅在部分帧像素点的灰度值有较大改变。因此可以对背景建立概率模型来拟合背景像素点的这种变化，通过计算一个像素点为背景的概率是否满足阈值，来判别单个像素点属于背景还是前景，从而对整个图像进行划分。

#### 3.2 对问题二的分析：

针对问题二，摄像头固定，背景有扰动。还是考虑单个像素点的变化，所有帧内，单个像素点的灰度值并不是固定不变，而是随着扰动（如水波等）出现往复变化，因此背景像素会周期性出现多个变化规律。接着问题一的思路，用概率模型来拟合背景像素的这种变化，由于背景像素具有多个变化规律，故使用多个概率模型来拟合。同样，通过计算一个像素点为背景的概率是否满足阈值，来判别单个像素点属于背景还是前景。最后由于背景为动态，还要选取合适的更新速率更新背景。

#### 3.3 对问题三的分析：

针对问题三，摄像头不固定，由于摄像头的运动，视频中的图像会发生整体移动。如果直接对前景目标进行检测，将会把大面积的背景识别为前景。可以采取运动补偿的措施，恢复相机运动带来的影响，然后对恢复后的图像进行前景检测。

#### 3.4 对问题四的分析：

利用构建的识别模型，在多个视频上进行检验，判断其效果。

#### 3.5 对问题五的分析：

多角度摄像机协同拍摄同一地点，需要对多个视频中相同的目标进行判定，提取出相应目标。可以利用摄像机摄像的原理，进行空间坐标转换，从而确定空间中同一位置处的前景目标。

### 3.5 对问题六的分析：

对识别到的前景目标的行为进行判断。可以使用前景提取模型提取目标，考虑前景目标的特征，利用常用的逻辑回归，支持向量机等分类方法，对有标签的视频进行训练，得到识别模型。其中目标的特征选取可选取范围较广，一般来说目标的速度，包括速度大小和速度方向，可以提供大量信息，比如人群排队中，目标的速度方向较为一致，而人群奔跑中，速度大小会普遍较大。因此可以利用光流法计算目标速度信息，作为识别模型的输入特征。

## 四 问题一的解答

### 4.1 模型的准备

拟采用概率模型对背景像素的分布进行拟合。取附件 2 静态背景中 office 场景视频的单个像素点，以帧数为计数，做出频率分布直方图，其背景像素点典型的分布直方图如下图所示。可近似用高斯分布来拟合。

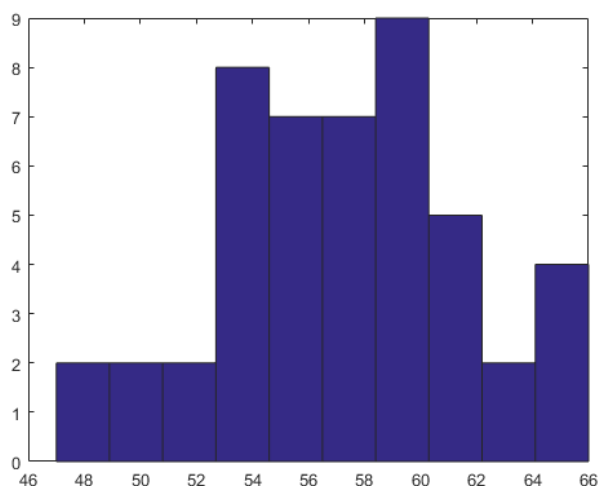


图 4-1 背景像素点典型的分布直方图

#### 高斯概率分布模型

若随机变量 $X$ 的数学期望为 $\mu$ 、方差为 $\sigma^2$ ，概率密度函数为

$$G(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

则称其服从高斯分布，记为 $x \sim N(\mu, \sigma^2)$ 。高斯分布的期望值 $\mu$ 决定了其位置，标准差 $\sigma$ 决定了分布的幅度。

### 4.2 模型的建立

由以上分析，单个背景像素点取值的概率可近似使用高斯分布拟合。



### (1) 背景建立

设  $a_k$  为第  $k$  帧视频时像素的灰度值，视频第  $m$  帧时的背景模型为前  $m-1$  帧图片得到。

$$G = \frac{1}{\sqrt{2\pi}\sigma_{m-1}} e^{-\frac{(x-\mu_{m-1})^2}{2\sigma_{m-1}^2}}$$

其中

$$\mu_{m-1} = \sum_{k=1}^{m-1} a_k$$

$$\sigma_{m-1} = \frac{1}{m-1} \sum_{k=1}^{m-1} (\mu_{m-1} - a_k)^2$$

### (2) 前景检测

利用高斯函数的  $3\sigma$  原则来判断第  $m$  帧中灰度为  $a_m$  的像素是否为背景。判别函数为

$$M_m = \frac{|a_m - \mu_{m-1}|}{\sigma_{m-1}}$$

设定阈值为  $T$ ，则像素点是否属于前景可通过下式来判断。

$$L = \begin{cases} 1, & M_m > T \quad \text{前景} \\ 0, & M_m \leq T \quad \text{背景} \end{cases}$$

### (3) 背景更新

检测完前景后，利用历史模型和当前图片对背景模型进行更新。这样当帧数  $m$  足够大时，背景模型将趋于稳定。新的背景模型为

$$G = \frac{1}{\sqrt{2\pi}\sigma_m} e^{-\frac{(x-\mu_m)^2}{2\sigma_m^2}}$$

## 4.3 模型的求解与分析

取附件 2 中的 office 视频，该视频共 48 帧，每一帧图片为  $120 \times 160$  像素的灰度图。视频中背景为几乎不变的办公室，前景为两个面向镜头走来的人。将视频转化为矩阵形式，编写 matlab 程序判别前景，在第 13 帧时开始识别出前景目标，直到第 48 帧（最后一帧）时前景目标的识别度越来越高。经过形态学处理后，识别出的前景目标如下图所示。识别效果较好。



图 4-1 原始图像



图 4-2 提取前景图像

## 五 问题二的解答

### 5.1 模型准备

在问题一高斯概率模型的基础上，对动态背景中的像素点进行分析。做出背景像素点典型的频率分布如下。

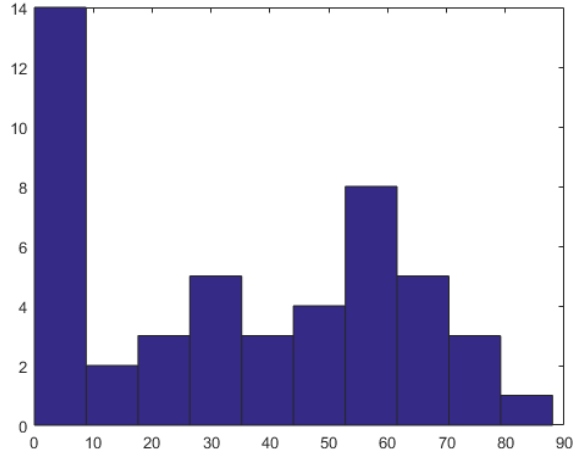


图 5-1 背景像素点典型的频率分布图

### 5.2 模型的建立

动态背景中，单个背景像素点取值的概率可近似使用多个高斯分布拟合，即传统的混合高斯模型。

#### (1) 背景建立

单个背景像素点取值的概率可近似使用混合高斯分布拟合，所以图像中的像素点  $(x_0, y_0)$  在  $t$  时刻的观察值  $X_t$  属于背景的概率为：

$$P(X_t) = \sum_{i=1}^K \omega_{i,t} * \eta(X_t, \mu_{i,t}, \Sigma_{i,t}) \quad (5.1)$$

其中，

$K$ ——为模型中的高斯分布个数；

$\omega_{i,t} (\sum_{i=1}^K \omega_{i,t} = 1)$ ——为  $t$  时刻第  $i$  个高斯分布的权值；

$\mu_{i,t}$ ——为  $t$  时刻第  $i$  个高斯分布的均值；

$\Sigma_{i,t}$ ——为  $t$  时刻第  $i$  个高斯分布的协方差矩阵，定义如下： $\Sigma_{i,t} = \sigma_i^2 I$

$\sigma_i^2$  为第  $i$  个高斯分布的方差， $I$  为  $i$  行  $i$  列的单位阵；

$\eta$  为高斯分布的概率密度函数：

$$\eta(X_t, \mu_{i,t}, \Sigma_{i,t}) = \frac{1}{(2\pi)^2 |\Sigma_{i,t}|^{\frac{1}{2}}} * e^{-\frac{1}{2}((X_t - \mu_{i,t})^T \Sigma_{i,t}^{-1} (X_t - \mu_{i,t}))} \quad (5.2)$$

K根据计算机的存储容量和计算能力决定取值，通常取 3~5。

## (2) 前景检测

K个高斯分布按照优先级 $\rho_{i,t} = \frac{\omega_{i,t}}{\sigma_i}$ 从高到低排列。每个新观测值 $X_t$ 与K个高斯分布按下式进行匹配：

$$|X_t - \mu_{i,t-1}| < 2.5\sigma_i \quad (5.3)$$

如果 $X_t$ 与任何一个高斯分布都不匹配，则优先级最低的高斯分布将被一个新的高斯分布替换。该分布的均值设为 $X_t$ ，并且初始化一个较大的方差和一个较低的优先级。在匹配过程中，高斯分布的权值按下式进行更新：

$$\omega_{i,t} = (1 - \alpha)\omega_{i,t-1} + \alpha(L_{i,t}) \quad (5.4)$$

$\alpha$ 即学习速率，由(5.4)可知该值越小，当前权值受历史权值的影响越大，则背景越稳定，反之背景受当前帧的影响越大；当新观察值与第 $i$ 个高斯分布匹配时， $L_{i,t}$ ，取值为1，若不匹配，则取 0。

$$L_{i,t} \begin{cases} = 0, & \text{前景} \\ = 1, & \text{背景} \end{cases}$$

## (3) 背景更新

与问题一的背景更新不同，混合高斯分布的背景更新按下式进行。对于匹配的高斯分布，其均值和方差按式(5.5)、(5.6)更新，没有匹配的高斯分布维持不变。

$$\mu_{i,t} = (1 - \beta)\mu_{i,t-1} + \beta X_t \quad (5.5)$$

$$\sigma_{i,t}^2 = (1 - \beta)\sigma_{i,t-1}^2 + \beta(X_t - \mu_{i,t})^T (X_t - \mu_{i,t}) \quad (5.6)$$

$$\beta = \alpha \eta(X_t | \mu_{i,t}, \sigma_{i,t}) \quad (5.7)$$

式(5.4)~(5.7)中的 $\alpha$ 和 $\beta$ 是模型更新过程中两个重要的参数。 $\alpha$ 是权值的学习速率， $\beta$ 为均值和方差的学习速率。 $\alpha$ 通过式(5.7)决定 $\beta$ 的取值，两者共同决定了背景模型的学习速率。

## (4) 模型改进

传统的混合高斯模型使用帧数的倒数作为学习速率 $\alpha$ ，即 $\alpha = \frac{1}{k}$ ，这样在前几帧背景能够迅速形成，而且帧数越多，背景的更新越缓慢，意味着背景也越稳定。对于背景

变化不大的场景，这样的取值是有很有效的。但是一旦背景发生较大变化，比如灯光的突然变化，背景模型将很难适应新的背景。考虑到这一点，本文使用新的自适应学习速率。

以  $a$  为计数值，初始时  $a=1$ ，每经过一帧  $a$  增加一。

$$\alpha = \frac{1}{a^2}$$

当检测到背景发生大面积改变，即前景占比过大时

$$\frac{M(\text{fore})}{M(\text{back})} > \text{Thresh}$$

令  $a=1$

此时背景将重新学习。并且能快速学习到新的背景。

### 5.3 模型的求解与分析

使用上述模型，对动态背景下的视频进行检测，matlab 程序见附件。首先测试了以水波为动态背景的视频。结果显示提取效果很好，人物形态完整。背景建立速度快，从第 8 帧开始就识别出显著的前景。

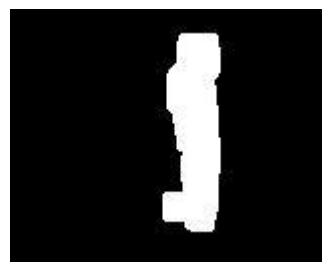


图 5-2 实际图像第 48 帧

图 5-3 提取后第 8 帧

图 5-4 提取后第 48 帧

对发生灯光变化的视频进行测试，在第 449 帧时光线突然变暗（如图 5-5），出现了大面积误检，经历了 11 帧后，模型学习到了新的背景。第 1062 帧时光线突然变暗，再次出现大面积误检，经历 16 帧后模型学习到新的背景，并趋于稳定。

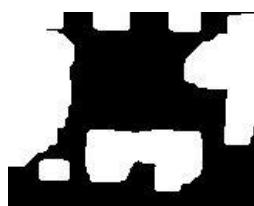


图 5-5 提取后第 449 帧

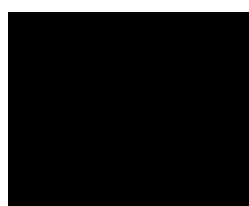


图 5-6 提取后第 461 帧



图 5-7 提取后第 1062 帧

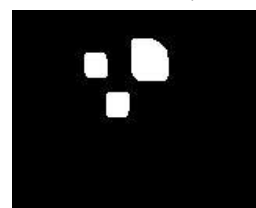


图 5-8 提取后第 1079 帧

## 结果分析

改进后的混合高斯模型鲁棒性强，能够适应各种不同的动态背景。具有背景建立速度快，建立之后稳定，能迅速应对光线变化等情况的特点。

## 六 问题三的解答

### 6.1 模型准备

当拍摄视频的摄像机发生运动时，前面的模型已经不能有效判别前景目标。因为图像发生全局运动，模型会将大面积的背景识别为前景。一般认为，图像的这种运动可以简化为仿射变换，通过对背景图像的全局运动进行估计，使用单应性矩阵对原始图像进行修改，可在一定程度上消除影响。最后对运动补偿后的图像可采用单通道高斯分布检测其前景。

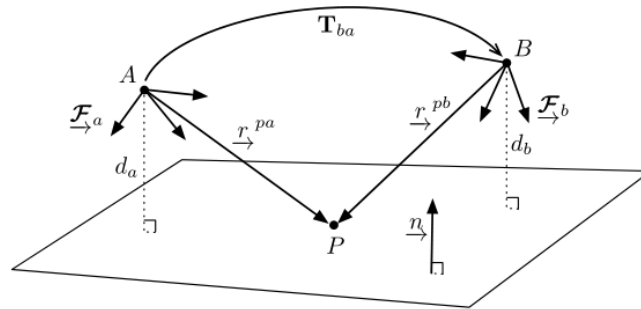
#### 摄像相关原理——单应矩阵

单应矩阵是一个数学概念，它定义了两幅图像之间的相互关系，一张图像上的任意一点可以在另一张图像上找到对应的点，且对应点唯一，反之亦然。

设  $a = (x, y, 1)^T$  为立体影像对上一点的齐次坐标， $b = (x', y', 1)^T$  为上一点的齐次坐标，则从  $a$  点变换到  $b$  点可以通过单应矩阵  $H$  得到  $b = Ha$ ， $H$  定义了两幅影像点之间的一一对应关系， $H$  是一个  $3 \times 3$  的矩阵，其定义式如下：

$$H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} = \begin{bmatrix} h_1^T \\ h_2^T \\ h_3^T \end{bmatrix} \text{ 其中 } h_i^T (i=1,2,3) \text{ 为向量 } (h_{i1}, h_{i2}, h_{i3})$$

如果已知观察的点在某个平面上，利用平面方程的先验信息可以估计出深度



已知相机中心到平面距离和法向量，可以得到平面的法线方程

$$n_i^T \rho_i + d_i = 0,$$

将空间坐标替换为像素坐标：

$$Z_i n_i^T K_i^{-1} q_i + d_i = 0,$$

或者

$$Z_i = -\frac{d_i}{n_i^T K_i^{-1} q_i},$$

替换掉，得到由像素齐次坐标到空间坐标的转换

$$\rho_i = -\frac{d_i}{n_i^T K_i^{-1} q_i} K_i^{-1} q_i$$

空间点在前后两帧坐标系下的三维坐标的约束关系

让我们假设我们知道  $T_{ba}$  从  $F_a$  变化到  $F_b$  因此

$$\begin{bmatrix} \rho_b \\ 1 \end{bmatrix} = \begin{bmatrix} C_{ba} & r_b^{ab} \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} \rho_a \\ 1 \end{bmatrix},$$

或者

$$\rho_b = C_{ba} \rho_a + r_b^{ab}.$$

带入第一个公式，得到两帧之间同一空间点的像素点坐标的约束关系

$$q_b = K_b H_{ba} K_a^{-1} q_a,$$

这里

$$H_{ab} = \frac{z_a}{z_b} C_{ba} \left( 1 + \frac{1}{d_a} r_a^{ab} n_a^T \right)$$

就称为单应矩阵<sup>[20]</sup>。

单应矩阵的二维形式即为仿射变换，本论文考虑摄像机**三维摄像**的形式。

### Harris 角点检测方法

如果在各个方向上移动这个特征的小窗口，窗口内区域的灰度发生了较大的变化，那么就认为在窗口内遇到了角点。如果这个特定的窗口在图像各个方向上移动时，窗口内图像的灰度没有发生变化，那么窗口内就不存在角点；如果窗口在某一个方向移动时，窗口内图像的灰度发生了较大的变化，而在另一些方向上没有发生变化，那么，窗口内的图像可能就是一条直线的线段。<sup>[19]</sup>

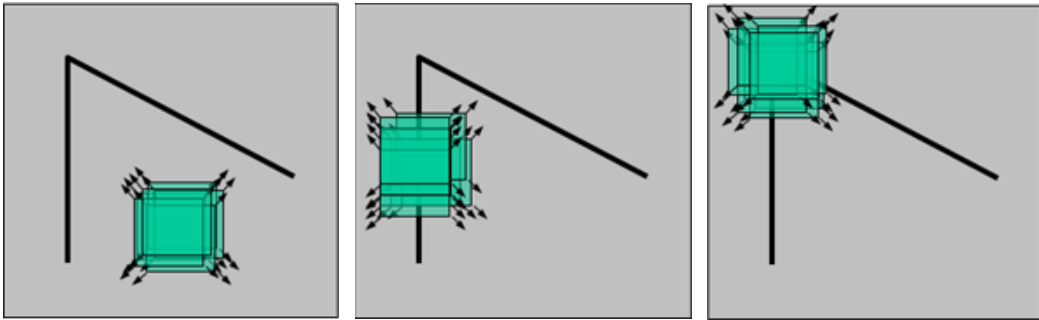


图 6-1 角点示意图

## 6.2 模型建立

### (1) 特征点提取和追踪

为了得到相邻两帧图像之间的变换矩阵，在图像上选取特征点，通过特征点的变化反映相机运动。特征点的选取使用 Harris 角点检测方法。假设相机的运动不大，且是连续的运动，所以相邻两帧图片的背景不会发生较大运动，只有前景像素会发生较大运

动。为了反映相机运动，对检测到的角点使用 LKT 光流法追踪，删除运动过大的特征。光流法的基本原理是由于运动目标和观测器之间的相对运动，以图像亮度变化作为识别对象，在序列图像中产生瞬时位移，体现了图像亮度模式的表观运动。图像中所有像素点的亮度光流就构成了图像的光流场。而光流法的核心就是从连续的图像系列中计算光流场。简单地说，光流场就是类刚体物体的速度矢量场。

光流法的计算方法是：视频图像序列中的像素点 $(x, y, t)$ ，经过时间 $\Delta t$ ，像素点移动到 $(x + \Delta x, y + \Delta y, t + \Delta t)$ ，由于像素点的位移变化与其灰度值是独立的，所以 $I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t)$ ，对时间 $t$ 求导得：

$$\begin{aligned} \frac{\partial I}{\partial x} \frac{\Delta x}{\Delta t} + \frac{\partial I}{\partial y} \frac{\Delta y}{\Delta t} + \frac{\partial I}{\partial t} &= 0 \\ \frac{\partial I}{\partial x} V_x + \frac{\partial I}{\partial y} V_y + \frac{\partial I}{\partial t} &= 0 \end{aligned}$$

其中， $V_x$ 和 $V_y$ 为像素点在 $x$ 轴和 $y$ 轴的速度，也就是 $I(x, y, t)$ 的光流， $\partial I / \partial x$ 和 $\partial I / \partial y$ 为图像 $I$ 相对于 $x$ 轴和 $y$ 轴的光强变化， $\partial I / \partial t$ 为相邻时间的同一像素点的光强变化。

计算运动量的函数由下式给出：

$$V = \sqrt{V_x^2 + V_y^2}$$

舍弃 $V > V_t$  ( $V_t$ 为阈值)的特征，相机运动越快，应当设置的阈值越大。

## (2) 求取单应性矩阵

得到特征点后，设 $H$ 为单应矩阵，则第 $i$ 个特征点 $X_i$ 和追踪到的特征点 $X'_i$ 之间的关系可表示为：

$$[X'_1 X'_2 \dots] = H[X_1 X_2 \dots]$$

## (3) 前景检测

得到单应性矩阵 $H$ 之后，通过单应矩阵对背景进行变换，消除相机运动带来的影响。对于当前帧中的一个像素 $X_c$ ，其对应的背景像素可表示为：

$$X_b = H^{-1} X_c$$

为了节省计算量，使用单高斯分布来建立背景模型 $N(\mu, \sigma^2)$ 。利用像素点的灰度值 $I$ 和背景模型比较来判断其是否属于前景。

$$D(x_c) = \frac{|I(X_b) - \mu|}{\sigma}$$

设定阈值为 $T$ ，则像素点是否属于前景可通过下式来判断。

$$L(X_c) = \begin{cases} 1, & D(x_c) > T \quad \text{前景} \\ 0, & D(x_c) \leq T \quad \text{背景} \end{cases}$$

## (4) 背景更新

模型以学习速率 $\alpha$ 对 $\mu$ 值和 $\sigma$ 进行更新。其中 $\alpha$ 为自适应， $Age$ 表示像素作为背景的帧

数，以 $Age$ 的倒数作为学习速率可以使新的图像区域迅速建立为背景，而多次作为背景的区域只会缓慢更新，慢慢移出镜头。

$$\begin{aligned}\mu(X_b) &= (1 - \alpha)\mu_{t-1}(X_b) + \alpha I(X_c) \\ \sigma(X_b) &= \sqrt{(1 - \alpha) \cdot \sigma(X_b)^2 + \alpha (I(X_c) - \mu(X_b))^2} \\ \alpha &= \left( \frac{1}{Age(\hat{X}_b)} \right) \cdot (0.1)\end{aligned}$$

### 6.3 模型的求解与分析

模型的求解主要使用 Harris 算法进行角点检测，使用 LKT 算法计算光流场，使用最小二乘法求解单应性矩阵。计算程序均已包含在附件。

使用 Harris 角点检测算法<sup>[18]</sup>检测图像角点。检测结果如图所示(白点为检测的角点)，树枝区域和汽车外壳都被标记为角点，而天空和道路等不易发生变化的目标未被标记。



图 6-2 检测角点

通过光流法计算速度矢量，筛选掉运动过大的点。剩下的点作为特征点。通过比较相邻两帧的特征点计算单应性矩阵。



图 6-3 筛选前



图 6-4 筛选后

使用选取的特征点计算单应性矩阵，并对图像进行矫正，矫正后的图像补偿了摄像机的运动。





图 6-5 矫正前



图 6-6 矫正后

对矫正后的图像使用**单通道高斯背景模型**进行前景检测。进行摄像机运动补偿后的图像总体能够识别出前景，但是效果并不理想。



图 6-7 原始图像



图 6-8 提取前景图像

### 结果分析

上述模型能够在一定程度上识别出前景。影响其识别准确度的因素主要有两个方面，一是计算单应性矩阵鲁棒性不够强，单应性矩阵的误差会对背景恢复产生较大影响；二是单通道高斯分布拟合背景精度不够，可以考虑使用更多高斯分布来建立背景模型。

## 七 问题四的解答

利用所构造的建模方法，从待检测视频中识别出包含显著前景目标的视频帧标号如下表。

表 7-1 著前景目标的视频帧标号表

视频名称	识别出显著前景目标的视频帧标号
Campus	200-224, 350-440, 651-682, 692-712, 810-875, 1007-1035, 1331-1405。
Curtain	967, 1144-1147, 1347-1371, 1562-1572, 1590-1598, 1770-1893, 2126, 2174-2316, 2635-2646, 2662-2667, 2711-2779
Escalator	3-174, 213-2391, 2754, 2766-3417。
Fountain	141, 158-210, 323-326, 335, 408-517, 519-523。
Hall	3-3534
Lobby	154-197, 345-349, 373-393, 427-460, 623-655, 664-668, 664-668, 964-966, 982-998, 1008-1036, 1062-1081, 1085-1538。
Office	197, 327, 501, 582-2041, 2080。
Overpass	214-237, 374, 468-483, 487-489, 500-522, 526-532, 559-637, 645-646, 721-728, 823-848, 1041-1048, 1095-1129, 1401-1422, 1551, 1611-1619, 1622-1638, 1714-1740, 1872-1884, 1886-1890, 1919-1942, 1983-2007, 2012-2016, 2019-2033, 2062-2063, 2018-2132, 2158-2196, 2206-2207, 2302-2306, 2317-2835。

### 结果分析

应用本文所建立的改进混合高斯模型对不同视频进行测试。实验表明，模型在动态背景下能够有效检测出前景目标，对于水波，窗帘、树枝等扰动有很好的**抗干扰能力**，而且能够应对光线的突变。总体来说具有较好的鲁棒性。当背景运动过于剧烈时仍会被误检为前景，但是**误检的频率较小**，不影响模型整体上对前景的识别。

## 八 问题五的解答

### 8.1 模型准备

在问题三中，本文已经叙述了摄像以及单应性矩阵的原理。利用空间坐标位置的变换，可以确定目标的空间位置并进行跟踪。

在多摄像头的跟踪系统中，目标与目标之间存在着一些有用的特征关系，比如目标之间的距离、夹角和相对速度等，这些特征关系对多摄像头之间目标的确认提供了有效的参考信息，因为这些特征关系不受平面单应性约束的影响，比如目标之间的距离在脚部位置和在中位置或者头部位置是相同的，而且在不同的摄像头中，这些特征也是非常稳定。利用平面单应性确定多摄像头之间的目标时，一个必须满足的条件是所有摄像头监控的场景必须是平面，但是这个约束条件在实际的情形中一般都得不到绝对的保证，所以一定的误差总会出现目标的确认中。在利用平面单应性对目标进行确定中，加入目标之间的距离特征，能很好的提高目标之间的确认精度。<sup>[12]</sup>

为了增加投影中的区分信息信息量，现有的方法从多个平面来计算单应性，这些平面都与地平面平行。单应性能够通过垂直移动无影点，然后估计与平面俯视图相平行的投影平面来获取。

让  $H^{C_j \pi_j}$  表示在摄像头  $C$  中的第  $j$  个平面的点  $C_j$ ，投影到第  $j$  个公共视角平面的单应性矩阵。投影公式如下：

$$Z_k^{\pi_j}(u, v) = H^{C_j \pi_j} Z^{C_j \pi_j}(x, y)$$

多层平面的投影可以分开进行处理，以获取有关目标的形状信息，也可以通过串联各平行平面的特征向量，将其合并到上述的单层单应性中来处理。公共视野能够通过融合来自三个单应性平面的像素来产生，这三个平面分别是目标的脚平面，头平面和这两个平面之间的一个平面。通过使用等式：

$$Z_k^{\pi, C}(u, v) = \frac{1}{\sigma^2 \left( \left\{ Z_k^{\pi, C}(u, v) \right\}_{C=\{1, \dots, N\}} \right)}$$

其中：

$$Z_k^{\pi, C}(u, v) = \begin{cases} H^{C, \pi} Z_k^C(x, y), \bar{Z}_k^C(x, y) = 1 \\ 0 \end{cases}$$

来产生一个方差图，来进行像素值的融合。

在公共平面中每个位置的信号强度与投影到该位置的前景像素的数量成比例。在多层的单应性中，来自不同摄像头视野中表示同一个目标（比如人）不同部分的像素沿着垂直轴（比如脚、腿、躯干，颈部和头部）投影到公共视野的相同位置，因此，就会增加了信号的强度。

## 8.2 模型的建立

寻找和匹配摄像头  $C_A$  和  $C_B$  监控区域的特征点，计算  $C_A$  到  $C_B$  的单应性矩阵  $H_{AB}$ 。

### 前景检测

获取目标的前景图，对每个目标分配一个标识号，并提取目标的脚部位置  $(P_i^A(x, y), P_i^B(x, y))$ ，并计算目标之间的距离  $d_{ij}^A$  和  $d_{ij}^B$ ，计算方法如下：

$$d_{ij}^A = \min(d_{ijl}^A, d_{ijm}^A, d_{ijh}^A)$$

其中  $d_{ijl}^A$ ， $d_{ijm}^A$  和  $d_{ijh}^A$  分别表示摄像头  $C_A$  中目标  $T_i^A$  和  $T_j^A$  之间脚部，中部和腿部之间的距离。

利用平面单应性将  $C_A$  中检测到的目标  $T_i^A$  投影到  $C_B$  中，得到在  $C_B$  中的位置  $P_i^{AB}(x, y)$ 。

计算  $C_A$  中目标投影到  $C_B$  中后，目标相互之间的距离  $d_{ij}^{AB}$ 。

$T_i^{AB}$  ( $C_A$  中的  $T_i^A$  投影到  $C_B$  中后得到的目标) 与  $C_B$  中对应的目标进行确认。确认方法如下：

(1) 如果  $T_i^{AB}$  的位置与  $C_A$  中的某个目标  $T_j^B$  重合，即  $P_i^{AB}(x, y) = P_j^B(x, y)$ ，则将  $C_A$  中的目标  $T_i^A$  与  $C_B$  中的目标  $T_j^B$  确认为同一个目标。

(2) 如果  $C_B$  中没有一个目标的位置与  $T_i^{AB}$  的位置重合，则确定以  $T_i^{AB}$  为中心， $r$  为半径的区域内  $C_B$  中的所有目标 (假定目标的个数为  $M$ )；对目标  $T_i^{AB}$  与从  $C_A$  投影到  $C_B$  中的其他目标之间的距离  $d_{ij}^{AB}$  ( $j=1, 2, \dots, N$  且  $j \neq i$ )，以及  $C_B$  中的这个  $M$  个目标分别与其他目标之间的距离  $d_{ik}^B$  ( $k=1, 2, \dots, N$  且  $k \neq i$ ) 进行从小到大的排序，得到集合  $S_i$  和  $S_t$  ( $t$  表示  $M$  个目标中的其中一个目标)；分别计算  $d_{ij}^{AB}$  与  $d_{ik}^B$  的和  $Sd_i^{AB}$ ， $Sd_t^{AB}$ ；从这  $M$  个目标中选择与  $Sd_i^{AB}$  值最接近的目标  $T_l^B$ ，如果集合  $S_i$  与  $S_t$  中每个位置对应的值之差的绝对小于某个阈值  $\theta$ ，即  $|S_{iq} - S_{tq}| \leq \theta$  ( $q=1, 2, \dots, N-1$ ) 其中  $S_{iq}$ ， $S_{tq}$  分别表示集合  $S_i$  和  $S_t$  中的第  $q$  个目标。则将  $C_A$  中的目标  $T_i^A$  与  $C_B$  中的目标  $T_l^B$  确认为同一个目标。如果集合  $S_i$  和  $S_t$  不能满足  $|S_{iq} - S_{tq}| \leq \theta$  ( $q=1, 2, \dots, N-1$ ) 式，则选择下一个与  $Sd_i^{AB}$  第二接近的目标，并检验该

目标是否满足 $|S_{i_q} - S_{l_q}| \leq (q = 1, 2, \dots, N-1)$ 式，若满足则确定 $T_j^A$ 与该目标为同一个目标，若不满足则选择与 $Sd_i^{AB}$ 第三接近的目标，依此类推<sup>[12]</sup>。

## 九 问题六的解答

### 9.1 模型准备

对前景对象的行为分析建立在前景检测的基础上。在问题二和问题三的解答中，本文已经提出了鲁棒性较强，准确率较高的前景检测方法。下面对目标的行为进行分析。

由于目标种类，目标形态的多样性，建立起统一的行为检测模型显得尤为困难。本文对针对人群的规律性行为进行分析和建模，但是同样的模型加以修改可以应用到不同的目标上。考虑到目标的行为特征很大程度上就是其运动的特征，对其运动的速度矢量进行分析，就能检测出不同的运动行为。目标运动的速度矢量可以由光流法获得。

建模过程主要包括以下几个步骤：视频图像预处理、检测前景目标、计算前景目标速度矢量、光流场特征分析、判断不同行为。

### 9.2 模型建立

#### 1. 视频图像预处理和前景目标检测

使用前文中提出的改进后的混合高斯分布模型检测前景。去除图像背景，使前景突出，得到的第*i*帧图像记为*F<sub>i</sub>*。

#### 2. 计算前景目标光流场

前景目标速度矢量可由相邻两帧*F<sub>i</sub>*和*F<sub>(i-1)</sub>*得到。相邻两帧视频图像序列中的像素点(*x, y, t*)，经过时间 $\Delta t$ ，像素点移动到(*x* +  $\Delta x$ , *y* +  $\Delta y$ , *t* +  $\Delta t$ )，由于像素点的位移变化与其灰度值是独立的，所以 $I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t)$ ，对时间*t*求导得：

$$\begin{aligned}\frac{\partial I}{\partial x} \frac{\Delta x}{\Delta t} + \frac{\partial I}{\partial y} \frac{\Delta y}{\Delta t} + \frac{\partial I}{\partial t} &= 0 \\ \frac{\partial I}{\partial x} V_x + \frac{\partial I}{\partial y} V_y + \frac{\partial I}{\partial t} &= 0\end{aligned}$$

其中， $V_x$ 和 $V_y$ 为像素点在*x*轴和*y*轴的速度，也就是 $I(x, y, t)$ 的光流， $\partial I / \partial x$ 和 $\partial I / \partial y$ 为图像*I*相对于*x*轴和*y*轴的光强变化， $\partial I / \partial t$ 为相邻时间的同一像素点的光强变化。

$V = (V_x, V_y)$ 即为速度矢量。

#### 3. 光流场特征分析

光流场的特征主要统计 4 个指标：

a.速度大小频率分布:均值和方差

$$\mu_1 = E(\|V\|)$$

$$\sigma_1 = Var(\|V\|)$$

b.速度方向的频率分布：均值和方差

$$\mu_2 = E(|\theta|)$$

$$\sigma_2 = Var(|\theta|)$$

使用一对多逻辑回归对目标行为进行判别，分别建立人群行走、人群奔跑、人群聚集的逻辑回归模型。需要使用已经判断过标签的训练集作为输入。输入特征值为  $X(1, \mu_1, \sigma_1, \mu_2, \sigma_2)$ 。

$$h_{\theta}(X) = \theta^T X$$

$$\sigma(X) = \frac{1}{1 + e^{-h_{\theta}(X)}}$$

针对三种行为分别建立三个逻辑回归模型  $\sigma_1, \sigma_2, \sigma_3$ 。

一组输入  $X(1, \mu_1, \theta_1, \mu_2, \theta_2)$  可得到一组输出  $\sigma = [\sigma_1, \sigma_2, \sigma_3]^T$ ， $\sigma$  的值可作为概率，取概率最大的值作为判定的事件，即：

$$L(\mu_1, \theta_1, \mu_2, \theta_2) = (\sigma = \max(\sigma))$$

其中

$$L = [1 \ 0 \ 0] \text{ 表示人群行走；}$$

$$L = [0 \ 1 \ 0] \text{ 表示人群奔跑；}$$

$$L = [0 \ 0 \ 1] \text{ 表示人群聚集。}$$

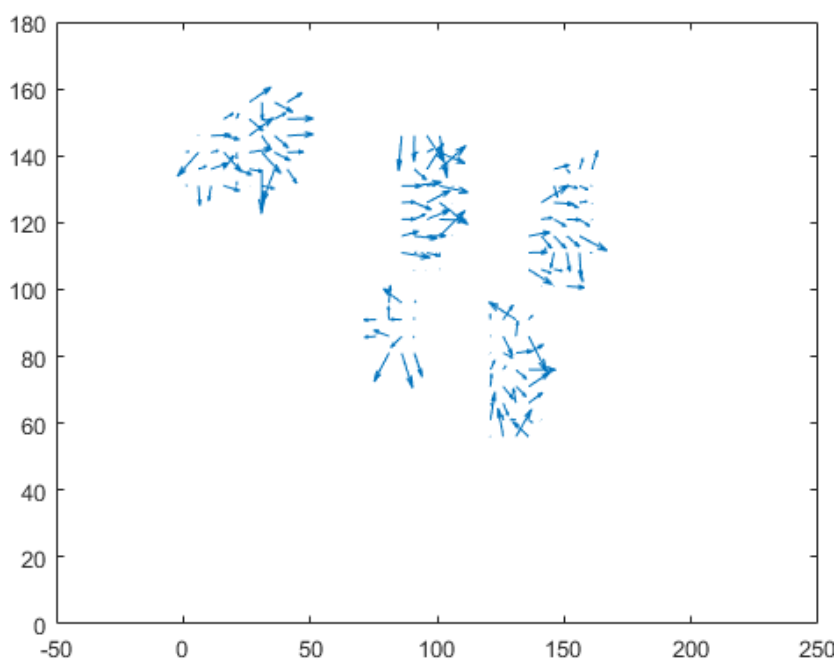
### 9.3 模型的求解与分析

选取具有人群行为特征的典型视频，以人群行走为例，进行前景检测，去除背景图像后，如图所示。

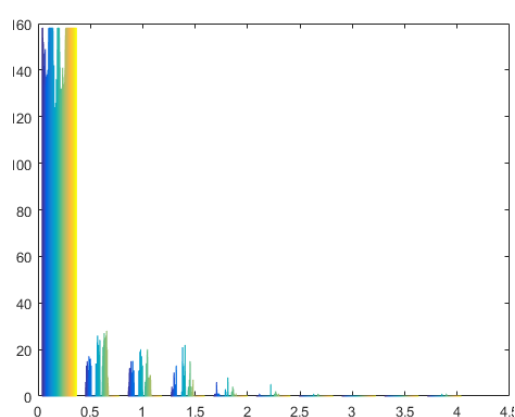


(去除背景后的连续两帧前景目标)

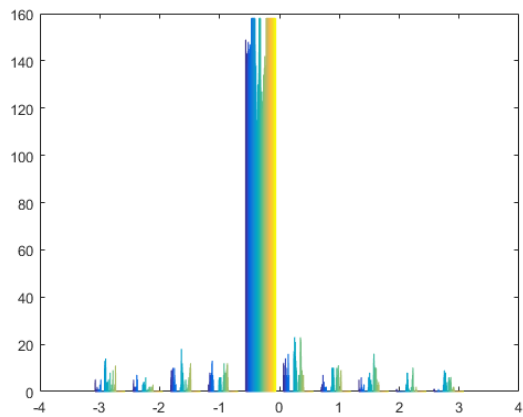
计算前景目标的光流场，得到目标运动的速度矢量，对速度大小和方向进行统计。得到频率分布如下。



(前景目标运动矢量)



(速度大小直方图)



(速度方向直方图)

计算特征值作为输入，使用人群行走视频中所有的图像进行上述步骤即可得到一组带有人群行走标签的训练集。其他行为的训练集也可用同样的方法得到。训练之后得到逻辑回归模型。逻辑回归程序见附件。

本次训练得到的特征输入见附件。提取特征样例：

0.054569	-0.01033	0.250667	0.333961	1
0.056577	-0.02799	0.264661	0.313905	1
0.072239	-0.01641	0.341167	0.40986	1
0.048131	-0.01768	0.22455	0.271429	1
...	...	...	...	...

## 结果分析

使用视频对模型进行测试。测试结果测准率和召回率较低，均不足 50%。原因有两点：一是从光流场中选取的特征较少，无法准确表达光流场的变化；二是逻辑回归中使用的是一次线性回归，可能会欠拟合。



## 十 模型的评价与改进

### 优点

- 1.问题一和问题二中使用高斯分布对背景建模，经过改进后的混合高斯分布，背景学习速率快，而且能快速从光线变化等全局影响中恢复。
- 2.问题三中采用角点检测寻找变化敏感点，而且用光流场筛选掉变化过快的点，特征点的选取非常合适，能够反映出相机运动，进而恢复图像与前景提取。
- 3.问题五中不仅考虑了空间坐标系，还考虑了物体的相对关系特征。高效利用了信息，能够对多摄像机处理提供依据。
- 4.问题六中选取目标的速度信息作为行为判别特征，能够有效反映目标的行为。

### 不足

- 1.问题三中计算单应性矩阵鲁棒性不够强，单应性矩阵的误差会对背景恢复产生较大影响。
- 2.问题六中使用的逻辑回归是一次线性表达式，可能会欠拟合。

### 改进

在对摄像机运动进行补偿时，可以使用鲁棒性更强的仿射变换。行为识别中识别方法使用支持向量机或者神经网络等，识别精度可能会大大提高。

## 十一 参考文献

- [1]张娇. 基于重叠视域多摄像机协同的目标定位与跟踪技术研究[D].南京邮电大学,2016.
- [2]郭达洁. 监控视频中的前景提取和目标检测跟踪算法研究[D].浙江大学,2016.
- [3]高翔,朱婷婷,刘洋. 多摄像头系统的目标检测与跟踪方法研究[J/OL].,2015,25(07):221-225. (2015-06-23)[2017-09-20].
- [4]尚进. 监控视频中行人异常行为检测系统的设计与实现[D].东南大学,2015.
- [5]黄大卫,胡文翔,吴小培,张超,周蚌艳. 改进单高斯模型的视频前景提取与破碎目标合并算法[J]. 信号处理,2015,31(03):299-307. [2017-09-20].
- [6]韩琦. 多视几何学在多摄像机协同目标跟踪系统中的应用和实现[D].北京邮电大学,2015.
- [7]凡如亚. 基于鲁棒主成分分析的低秩矩阵恢复算法及其应用研究[D].国防科学技术大学,2014.
- [8]朱力旻. 视频中前景提取技术的研究[D].上海交通大学,2014.
- [9]衡林. 多摄像机视域中的运动目标检测与跟踪研究[D].南京邮电大学,2013.
- [10]邓小宇. 面向智能视频监控的前景提取和动作识别技术研究[D].浙江大学,2012.
- [11]闵华清,陈聪,罗荣华,朱金辉. 基于时空分析的视频前景提取[J]. 模式识别与人工智能,2011,24(04):582-590. [2017-09-20]. DOI: 10.16451/j.cnki.issn1003-6059.2011.04.016
- [12]颜杰. 多摄像头目标检测与跟踪方法研究[D].华中科技大学,2011.
- [13]Andrews Sobral, Antoine Vacavant. A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos, Computer Vision and Image Understanding, Volume 122, 2014, Pages 4-21, ISSN 1077-3142.
- [14]Daniya Zamalieva, Alper Yilmaz, Background subtraction for the moving camera: A geometric approach, In Computer Vision and Image Understanding, Volume 127, 2014, Pages 73-85, ISSN 1077-3142.
- [15]Sohailah Alyammahi, Harish Bhaskar, Dymitr Ruta, Mohammed Al-Mualla, People detection and articulated pose estimation framework for crowded scenes, In Knowledge-Based Systems, Volume 131, 2017, Pages 83-104, ISSN 0950-7051
- [16]Amitha Viswanath, Reena Kumari Behera, Vinuchackravathy Senthamilarasu, Krishnan Kutty,

Background Modelling from a Moving Camera, In Procedia Computer Science, Volume 58, 2015, Pages 289-296, ISSN 1877-0509

[17]Stauffer C., Grimson W.E.L. Adaptive background mixture models for real-time tracking. In: Computer Vision and Pattern Recognition, IEEE Computer Society Conference on, vol. 2, IEEE, 1999.

[18]Harris C., Stephens M. A combined corner and edge detector. In: Alvey vision conference, vol. 15, 1988, pp. 50.

[19] <http://www.cnblogs.com/ronny/p/4009425.html>

[20] <http://www.cnblogs.com/youzx/p/6385513.html>

## 十二 附件

问题六中提取的速度特征

0.054569	-0.01033	0.250667	0.333961	1	行走
0.056577	-0.02799	0.264661	0.313905	1	
0.072239	-0.01641	0.341167	0.40986	1	
0.048131	-0.01768	0.22455	0.271429	1	
0.087415	-0.00663	0.368461	0.435368	1	
0.085987	-0.00444	0.33435	0.461074	1	
0.085566	-0.03811	0.349521	0.392976	1	
0.082178	-0.02293	0.341261	0.370439	1	
0.070435	-0.02858	0.277411	0.420733	1	
0.089513	-0.03333	0.345409	0.474849	1	
0.089321	-0.02566	0.330348	0.48149	1	
0.077318	-0.02129	0.28351	0.460642	1	
0.079673	-0.0233	0.29924	0.477299	1	
0.072805	-0.01686	0.255392	0.454909	1	
0.041646	0.004212	0.268971	0.367826	2	奔跑
0.035148	0.002341	0.262448	0.322115	2	
0.0256	0.00505	0.225679	0.222288	2	
0.012707	0.012638	0.108327	0.154807	2	
0.032177	0.007488	0.23168	0.292911	2	
0.02134	-0.00723	0.215909	0.215933	2	
0.009318	0.005398	0.097139	0.232211	2	
0.046773	-0.00673	0.478123	0.301883	2	
0.031299	-0.00576	0.223116	0.34156	2	
0.024488	0.022024	0.159428	0.260289	2	
0.023111	0.015057	0.147041	0.325797	2	
0.09477	-0.01516	0.527912	0.490183	2	
0.042113	-0.01624	0.204076	0.462313	2	
0.026429	0.006121	0.182474	0.471939	2	
0.026518	0.015454	0.17399	0.363384	2	
0.049269	-0.03914	0.297105	0.476739	2	
0.043424	-0.02231	0.239435	0.503234	2	
0.063636	-0.01956	0.331897	0.429108	2	
0.070994	-0.01527	0.353015	0.430366	2	
0.051931	0.017838	0.310633	0.415389	2	
0.029759	-0.01085	0.185364	0.392688	2	
0.052046	0.009315	0.29227	0.375134	2	
0.053122	-0.00117	0.273954	0.348149	2	

0.055265	0.004522	0.282901	0.336578	2	
0.029159	0.009655	0.158087	0.364208	2	
0.049564	0.028279	0.249009	0.364899	2	
0.057559	-0.001	0.270753	0.375312	2	
0.056912	0.018905	0.278002	0.357613	2	
0.052035	0.003804	0.268668	0.347246	2	
0.054012	0.018147	0.282401	0.363049	2	
0.045564	0.016692	0.244332	0.370146	2	
0.046658	-0.0104	0.274199	0.330101	2	
0.061699	0.008355	0.34605	0.384024	2	
0.063665	0.021687	0.284689	0.406198	2	
0.055871	0.025531	0.271952	0.396353	2	
0.038342	-0.00132	0.198255	0.395126	2	
0.059694	0.033722	0.304902	0.43921	2	
0.07089	0.017496	0.348108	0.400802	2	
0.069094	0.012836	0.301046	0.401288	2	
0.2255	0.011824	0.599291	0.734291	3	聚集
0.215199	0.011482	0.622881	0.648441	3	
0.178236	0.008571	0.515459	0.575867	3	
0.130596	0.013873	0.36569	0.41508	3	
0.228374	-0.01009	0.737694	0.672554	3	
0.292058	-0.03084	0.816943	0.787346	3	
0.2334	-0.02236	0.626094	0.726516	3	
0.185463	0.003518	0.555453	0.655565	3	
0.178278	-0.00429	0.53067	0.658975	3	
0.150844	-0.01997	0.492102	0.576811	3	
0.139152	-0.048	0.45202	0.628468	3	
0.100162	-0.01401	0.314289	0.583881	3	
0.13625	-0.00285	0.4862	0.577532	3	
0.184743	-0.02524	0.612715	0.626987	3	

问题一中的单通道高斯模型

```
function gsingle(dir,name)
    video = load(strcat(dir,name,'.mat'));
    vmat = video.obj.XX;
    vlen = size(vmat,2);
    ishape = video.obj.siz;
    miu = mean(vmat, 2);
```

```

sigma = std(double(vmat), 0, 2);
for i = 1 : vlen
    bw = (double(vmat(:,i)) - double(miu)) > 2*sigma;
    img = reshape(bw,ishape);
    imgname = strcat('G:\matvideo\out\',name,'\fg\'', num2str(i), '.jpg');
    imwrite(img,imgname)
end
imshow(img)
img2avi(strcat('G:\matvideo\out\',name,'\fg\''),strcat('G:\matvideo\out\',name,'\'));
end

```

问题二中改进的背景提取方法

```

function check(dir,name)
    video = load(strcat(dir,name,'.mat'));
    vmat = video.Ori_H;
    vlen = size(vmat,3); %输入为三维的矩阵
    Thresh = 1/2;
    fdetector = vision.ForegroundDetector('NumTrainingFrames', 5,'InitialVariance',
30*30,'AdaptLearningRate',false);
    a = 1;
    row = size(vmat,1);
    col = size(vmat,2);
    totalp = row*col;
    for i = 1 : vlen
        frame = vmat(:,:,i);
        learningrate = 1/(a^2);
        fg = step(fdetector, frame, learningrate);
        bw = bwmorph(fg,'open',100); %开操作,先腐蚀后膨胀
        bw = bwmorph(bw,'majority',100); %联通
        SE=strel('rectangle',[10 10]);
        bw=imdilate(bw,SE);
        % stats = regionprops(bw, 'ConvexImage'); %划分区域,对找特征有用
        img = bw;
        a = a + 1;

        fore = sum(sum(img)); %判断是否有大面积变化
        if fore/totalp > Thresh
            a = 1;
        end

        % if size(stats,1) ~=0

```

```

%         img = stats.ConvexImage;
%     end
%     imgname = strcat('G:\matvideo\out\',name,'\fg\' , num2str(i), '.jpg');
%     imwrite(img,imgname)
end
imshow(img)
img2avi(strcat('G:\matvideo\out\',name,'\fg\' ),strcat('G:\matvideo\out\',name,'\'));
end

```

问题三中的单应矩阵计算方法

```

function v = Homography(pin, pout)
    if ~isequal(size(pin), size(pout))
        error('Points matrices different sizes');
    end
    if size(pin, 1) ~= 2
        error('Points matrices must have two rows');
    end
    n = size(pin, 2);
    if n < 4
        error('Need at least 4 matching points');
    end
    % Solve equations using SVD
    x = pout(1, :); y = pout(2, :); X = pin(1, :); Y = pin(2, :);
    rows0 = zeros(3, n);
    rowsXY = -[X; Y; ones(1,n)];
    hx = [rowsXY; rows0; x.*X; x.*Y; x];
    hy = [rows0; rowsXY; y.*X; y.*Y; y];
    h = [hx hy];
    if n == 4
        [U, ~, ~] = svd(h);
    else
        [U, ~, ~] = svd(h, 'econ');
    end
    v = (reshape(U(:,9), 3, 3)).';
end

```

问题三中的角点检测方法

```

function corner = Cornerdetect(img,thresh)
    img =double(img);
    [m n]=size(img);

    tmp=zeros(m+2,n+2);
    tmp(2:m+1,2:n+1)=img;

```

```

Ix=zeros(m+2,n+2);
Iy=zeros(m+2,n+2);

E=zeros(m+2,n+2);

Ix(:,2:n)=tmp(:,3:n+1)-tmp(:,1:n-1);
Iy(2:m,:)=tmp(3:m+1,:)-tmp(1:m-1,:);

Ix2=Ix(2:m+1,2:n+1).^2;
Iy2=Iy(2:m+1,2:n+1).^2;
Ixy=Ix(2:m+1,2:n+1).*Iy(2:m+1,2:n+1);

h=fspecial('gaussian',[7 7],2);
Ix2=filter2(h,Ix2);
Iy2=filter2(h,Iy2);
Ixy=filter2(h,Ixy);

Rmax=0;
R=zeros(m,n);
for i=1:m
    for j=1:n
        M=[Ix2(i,j) Ixy(i,j);Ixy(i,j) Iy2(i,j)];
        R(i,j)=det(M)-0.06*(trace(M))^2;

        if R(i,j)>Rmax
            Rmax=R(i,j);
        end
    end
end

tmp(2:m+1,2:n+1)=R;
img_re=zeros(m+2,n+2);
img_re(2:m+1,2:n+1)=img;

c = zeros(m+2,n+2); %返回角点
for i=2:m+1
    for j=2:n+1

        if tmp(i,j)>thresh*Rmax &&...
            tmp(i,j)>tmp(i-1,j-1) && tmp(i,j)>tmp(i-1,j) &&
tmp(i,j)>tmp(i-1,j+1) &&...
            tmp(i,j)>tmp(i,j-1) && tmp(i,j)>tmp(i,j+1) &&...
            tmp(i,j)>tmp(i+1,j-1) && tmp(i,j)>tmp(i+1,j) &&
tmp(i,j)>tmp(i+1,j+1)

```



```

            img_re(i,j)=255;
            c(i,j) = 1;
        end
    end
end
corner = c(2:m+1,2:n+1);
end

```

问题三中的前景检测模型

```

clear
A = load('car6.mat');
car = A.obj.XX;
len = size(car,2);
ishape = A.obj.siz;
row = ishape(1);
col = ishape(2);
vthresh = 0.001;
othresh = 0.001;
cthresh = 0.02;
alph = 0.8;
opticFlow = opticalFlowLK('NoiseThreshold',othresh);
backimg = reshape(car(:,1), ishape);

miu = backimg;
co = zeros(ishape);
% co = double(backimg)*double(backimg);
for i = 2:len
    img = reshape(car(:,i), ishape);
    %imshow(img)
    %pre_corners = Cornerdetect(back,cthresh);
    basecor = Cornerdetect(miu,cthresh);

    flow = estimateFlow(opticFlow,basecor); %光流法追踪

    corners = Cornerdetect(img,cthresh);
    flow = estimateFlow(opticFlow,corners);
    vx = flow.Vx;
    vy = flow.Vy;
    ca = vx.^2+vy.^2;
    p_features = basecor;

    [p_r,p_c,p_v] = find(p_features ~= 0); %去除变化太快的特征点
    [c_r,c_c,c_v] = find(corners~=0);
    p_features = [p_r p_c];
    c_features = [c_r c_c];
end

```

```

p_features = p_features((1:50),:);
c_features = c_features((1:50),:);

H = Homography(p_features',c_features'); %计算单应矩阵

%标准化
himg = ones(3,row*col);
for p = 1:row
    for q = 1:col
        himg(:,(p-1)*col+q) = [p;q;1];
    end
end
waped = inv(H) * himg;

bimg = img; % 不重叠部分用新图

for num = 1 : row*col %tp为totalpixel
    wapedrow=int16(waped(1,num));
    wapedcol=int16(waped(1,num));
    if wapedrow<row && wapedcol<col && wapedrow>0 &&
wapedcol>0
        bimg(wapedrow,wapedcol) =
img(himg(1,num),himg(1,num));
    end
end
%imshow(bimg);
%pause;

%得到背景图后检测

%    coback = co; %求xb的协方差

%    Dc = zeros(ishape); %二值矩阵

%    for num = 1 : row*col %tp为totalpixel
%        wapedrow=int16(waped(1,num));
%        wapedcol=int16(waped(1,num));
%        if wapedrow<row && wapedcol<col && wapedrow>0 &&
wapedcol>0
%            miub = miu(wapedrow,wapedcol);
%            cob = co(wapedrow,wapedcol);
%        else

```

```

%         miub = miu(himg(1,num),himg(1,num));
%         cob = co(himg(1,num),himg(1,num));
%     end
%     Ic = img(himg(1,num),himg(1,num));
%     Dc(himg(1,num),himg(1,num)) = (Ic-miub)^2/(cob^2); %c

```

和b的比较

```

%         coback(himg(1,num),himg(1,num)) = cob;
%     end
%     res = Dc>0;

    check = abs(double(bimg)-double(miu))-10000*double(co);
    res = check>0;

    %背景更新

%     miu = (1-alpha)*bimg + alph*img;
%     co = (1-alpha)*(im2double(coback)'*im2double(coback)) +
alph*im2double((img - bimg))'*im2double((img-bimg));
    miu = (1-alpha)*bimg + alph*img;
    co =
((1-alpha)*(co.^2)+alph*((double(img)-double(bimg)).^2)).^(1/2)
;
    imshow(res);
    pause;

end

```

问题六中模型

```

function [all_theta] = oneVsAll(X, y, num_labels, lambda)
m = size(X, 1);
n = size(X, 2);
all_theta = zeros(num_labels, n + 1);
X = [ones(m, 1) X];

for k = 1 : num_labels
    initial_theta = zeros(n + 1, 1);
    options = optimset('GradObj', 'on', 'MaxIter', 50);
    [theta] = fmincg (@(t)(lrCostFunction(t, X, (y == k),
lambda)),initial_theta, options);
    all_theta( k , : ) = theta;
end

```

```

end

function g = sigmoid(z)
g = 1.0 ./ (1.0 + exp(-z));
end

function [J, grad] = lrCostFunction(theta, X, y, lambda)

m = length(y);
J = 0;
grad = zeros(size(theta));
temp = theta;
temp(1) = 0;

z = sigmoid(X * theta);

J = (-y' * log(z) - (1 - y)' * log(1 - z)) / m ;
grad = X' * (z - y) / m;

J = J + lambda / (2*m) * temp'* temp;
grad = grad + temp * lambda/m;

grad = grad(:);

end

```