

# Earthquake Detection Based on Video Analysis

Cheng Huang, Tuo Zhong Yao\*

School of Electronics and Information Engineering, Ningbo University of Technology, Ningbo Zhejiang  
Email: [thomasyao@zju.edu.cn](mailto:thomasyao@zju.edu.cn)

Received: Jul. 28<sup>th</sup>, 2019; accepted: Aug. 15<sup>th</sup>, 2019; published: Aug. 22<sup>nd</sup>, 2019

---

## Abstract

Traditional seismic intensity analysis is based on traditional seismograph and optical seismograph. Although this method is widely used and relatively mature in technology, its response speed is slow and equipment cost is high. In contrast, seismic detection based on video analysis is a complement to traditional methods, which has the advantages of fast response and low cost. In this paper, computer vision technology is applied to seismic analysis. Motion vector sequence in video is obtained by motion estimation method. From this sequence analysis, whether earthquake occurs in the video is obtained, and the function of intercepting seismic video, seismic intensity and drawing acceleration and jitter maps is realized. Then, the gray value analysis, target segmentation, feature point matching and other methods are used to filter out the misjudgments of some special scenes. For the validity of this method, this method is applied to a large number of *in-situ* seismic videos and some seismic videos on the network. Experiments show that the proposed method can effectively accomplish these tasks.

## Keywords

Earthquake Intensity Analysis, Moving Object Segmentation, Feature Matching, Video Analysis

---

# 基于视频分析的地震检测

黄 诚, 姚拓中\*

宁波工程学院电子与信息工程学院, 浙江 宁波  
Email: [thomasyao@zju.edu.cn](mailto:thomasyao@zju.edu.cn)

收稿日期: 2019年7月28日; 录用日期: 2019年8月15日; 发布日期: 2019年8月22日

\*通讯作者。

## 摘要

传统的地震烈度分析是基于传统地震测量仪器以及光学式地震仪进行的。虽然这种分析的方法应用广泛, 技术相对成熟, 但是其响应速度较慢、设备成本高。相比之下, 基于视频分析的地震检测是对传统方法的一种补充, 有着响应速度快, 投入成本较小的优势。因此本文将计算机视觉技术应用到地震分析中, 通过运动估计的方法获取视频中的运动矢量序列, 由该序列分析得出该视频中是否发生地震, 并实现截取地震部分视频、地震的烈度以及绘制其抖动图的功能。再通过灰度值分析、目标分割、特征点匹配等方法滤除一些特殊场景的误判。本文将该算法应用于地震中的现场地震视频和网络上的一些地震视频, 并通过实验验证了该算法的有效性。

## 关键词

地震烈度分析, 运动目标分割, 特征匹配, 视频分析

Copyright © 2019 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 引言

当前基于地震仪测量的传统地震烈度分析方法还是主流的方法, 但是其响应时间长, 部署成本高等问题越来越受到人们的重视。又随着计算机视觉技术的发展、监控硬件的成本的降低和视频监控网络发展迅速, 监控摄像头遍布各个地区, 甚至村庄。人们把目光转向了通过视频分析的方法进行地震烈度的估计。当地震发生时, 这些摄像头完整的记录了地震发生的过程。

获取地震的烈度, 计算其加速度是地震分析的关键。通过传统地震仪器进行估算的方法无法准确且快速的估计震中周围的地区的地震烈度。而且其成本昂贵, 无法大规模的部署, 导致有些地区的地震无法快速估计。此外, 现有传统的地震分析方法聚焦于通过仪器采集各种地底的反馈信号, 对地面上的肉眼可见的震动研究则非常少, 而地面上的震动可以非常直观的观察出当地的地震烈度。通过地震视频分析的方法恰恰能把地面上这些震动的信息利用起来, 结合地底的反馈信息进行快速的地震烈度估计。且基于视频分析的方法只需要进行一次开发就能重复的部署在各个地区, 投入成本低。因此, 本文针对这些现状提出了一种基于视频分析的地震检测, 能够针对一些常见场景的地震视频进行分析, 并估算出运动加速度, 地震烈度, 绘制其加速度图等功能。

## 2. 相关研究

地震最早是由地震仪器来测量的, 比如世界上最早的地震仪张衡地动仪, 之后的机械地震仪、电磁地震仪, 再到现代的宽波段地震仪。近百年来, 地震仪器技术经历了很大的发展, 其功能和参数指标也发生了巨大的变化。但是传统的地震仪器检测技术存在如下几个问题: 1) 对常规物理性能无法进行量化测试; 2) 对多级排列等先进功能无法进行量化测试; 3) 对软件和固件性能无法进行量化测试; 4) 对特定功能或作用的局部硬件电路性能无法进行量化测试[1]。因此随着科技的发展, 新的检测技术的探索和创新是十分重要和必要的。

近年来, 逐渐有人将运动估计算法应用到地震检测领域中, 即基于视频分析的地震分析。随着硬件

成本的降低和视频分析技术的发展,视频监控网络发展迅速,视频监控可以真实还原地震发生时的状况,还可以直观反应灾害分布情况。监控的内容有着很多能反映出地震级数,地震的破坏程度等关键信息。但是视频的数量是巨大的,使用人工观察的方式是不现实的。因此 2017 年郭志宇提出基于监控视频的震害快速评估的方法[2],2017 年李东平提出了基于海量视频监控的中小地震烈度快速判断方法[3],为地震烈度快速估计开启了一个新的方向。但是前者只是定性的给出了地震的危害程度,并没有很精确的估计出该地区的地震烈度,而后者能给出一定精确度的地震烈度,却未考虑到一些特殊场景,如:夜间强闪光或者行人或者车辆等规律运动物体所造成的影响。

运动估计就是在序列图像中提取有关物体运动的信息,其表达方式是运动矢量[4]。运动估计算法现阶段被广泛的应用在各种领域,如在 2003 年 h.264 标准的提出中,运动估计算法被用来节省数据传输的带宽[5],令狐永芳在 2007 年提出基于运动估计的运动目标分割[13],2012 年夏东将运动估计应用在行人检测中[6]。

运动估计的方法有很多,如:基于光流场,基于运动恢复结构(SFM),基于宏块匹配等。

光流场:在上个世纪 Bruce D. Lucas 和 Takeo Kanade 发明了一种光流估计差分的方法(Lucas-Kanade Method)降低了光流估计对图像噪声的敏感度。在 2001 年杨勇提出了一种基于改进的光流场的运动估计算法[7],加快了光流估计的运算速度。在 2015 年李艳萍提出了 SIFT 特征光流的方法[8]。由于 SIFT 特征具有尺度不变性[9],因此这个方法进一步提高了光流估计算法的稳定性。但是基于光流场的方法运算量过大,实时性相对较差。

运动恢复结构(SFM):在 1991 年 Koenderink J 提出了运动恢复结构[10],通过获取本质矩阵和基本矩阵,再由它们分解得到旋转矩阵和矩阵的平移向量(类似于运动矢量)以达到运动估计的目的。在 2014 年余洪山团队将类似 SFM 的算法应用到三维空间中的运动估计[11],提高了三维运动估计精度,且具有时间复杂度相对较小的优势。基于 SFM 的方法虽然估计的准确度提高了不少,但是需要进行大量的矩阵计算,耗时较大。

宏块匹配:在 2003 年 Wenger S 介绍了 h.264 这一编解码标准[5] (由国际电信联盟(ITU)与国际标准化组织(ISO)共同提出),其运动估计算法由点匹配上升到了由宏块进行匹配,计算量大幅下降,适用于编解码这种实时性要求比较高的场景。在 2009 年于飞的 h.264 运动估计算法分析中加速和提高了运动矢量获取的速度和精度[4]。2012 年 h.265 标准的通过,标志着网络视频进入高清时代。它的运动估计算法将宏块大小在 h.264 的基础上增加了 4 倍以上,进一步加快了算法运行速度,以适应更高的实时性要求。2018 年秦晓波等人将宏块匹配和光流法相结合提出了基于宏块的 Hs 光流算法运动估计方法[12],改进了光流场质量的同时也加快了运动估计算法的运行速度。基于宏块匹配的方法有着较强的实时性,符合本文的项目场景的要求,对本文的运动估计方法的选择有着较大的借鉴意义。

### 3. 算法概述

如何快速的精确的估算出地震烈度是地震分析的中最为关键的技术难题。为此,本文将 H.264 标准中的运动估计算法应用在地震分析中,并在此基础上添加了目标分割和特征匹配算法,增强其鲁棒性。算法结合常规情况和特殊情况提出如下的流程,如图 1 所示:

- 1) 通过运动估计算法计算每一帧每个宏块的运动矢量之和;
- 2) 对每一帧的运动矢量进行平滑和去噪声;
- 3) 根据运动矢量的最大值以及较高矢量值持续的帧数判读是否发生地震;
- 4) 根据矢量最大值所在的帧号,对其局部帧进行图像像素值的统计,以其灰度值是否发生剧烈变化来判断是否是闪光高亮异常所造成的误检;

5) 根据矢量最大值所在的帧号, 对其局部帧进行运动物体分割, 相邻帧之间进行特征点匹配, 计算其运动方向, 判断是否是物体(行人或者车辆)规律运动场景所造成的误判;

6) 最后根据所获得的每一帧的矢量值, 画出地震视频的震动图和加速度图, 截出较为精准的地震视频, 即地震前后各延迟数秒钟。并根据经验函数计算出视频中物体的最大加速度, 得出该地震视频的烈度值。

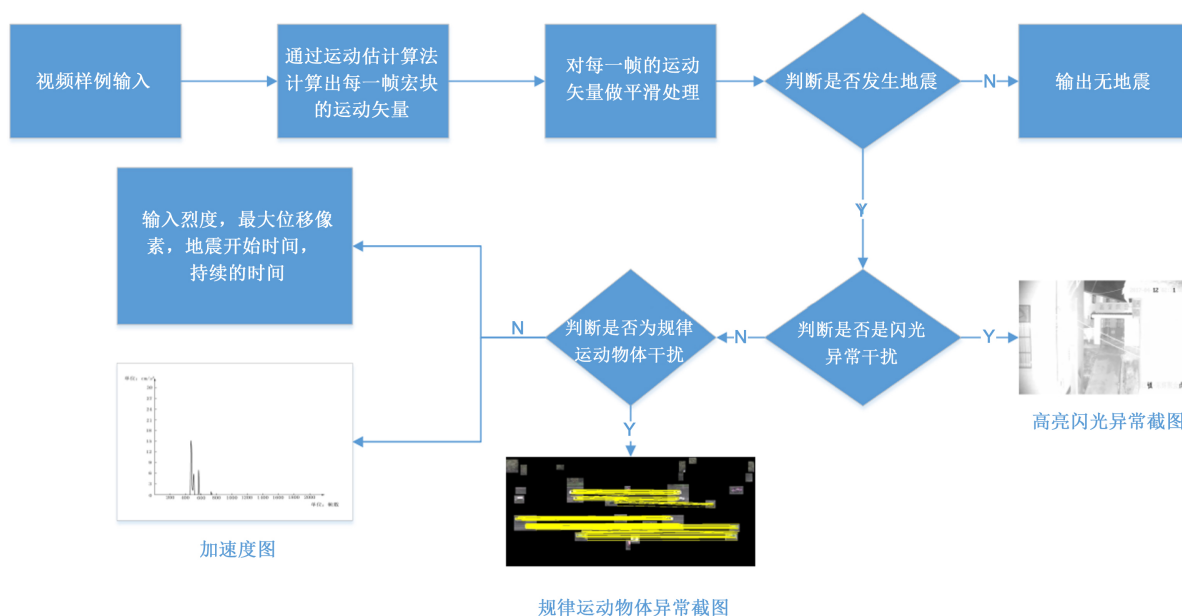


Figure 1. The seismic detection implementation framework

图 1. 本地震检测实现框架

## 4. 地震检测算法

### 4.1. 基于运动估计的地震分析

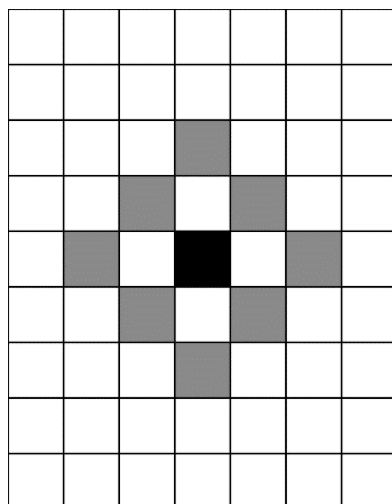
本文的地震分析的方法主要运用了运动估计, 灰度值分析、目标分割、特征点匹配等计算机视觉技术。在本章我将要介绍本文的核心算法——运动估计算法和运动矢量平滑后处理。我将从宏块的划分到运动矢量的计算来详细的阐述该运动估计算法。最后再阐述运动矢量平滑后处理操作, 使得到的运动矢量能够更好的应用于地震分析。

#### 运动估计算法概述

运动估计的基本思想如下: 将一副图像划分成一个个不重叠且大小相同的宏块, 一个宏块就是一个整体、相对独立的单位, 并认为宏块里面每个像素运动矢量都是一样的。在后一帧和前一帧图像内的相对确定范围内进行搜索, 采用求和绝对误差(SAD)距离函数(如式 1)进行匹配, 最相似的块就是匹配块。然后由前一帧的宏块与后一帧的匹配块的相对位置, 计算出运动位移。该运动位移就是当前宏块的运动矢量。

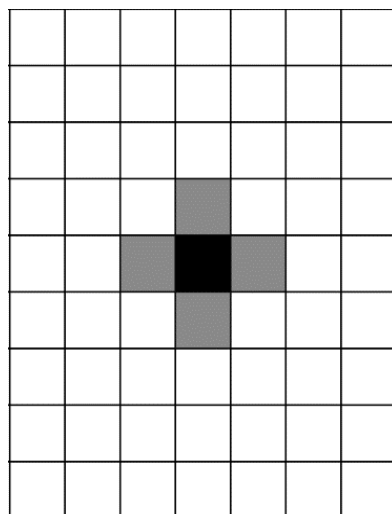
$$\text{SAD}(i, j) = \sum_{m=1}^M \sum_{n=1}^N |f_k(m, n) - f_{k-1}(m+i, n+j)| \quad (1)$$

搜索策略是运动估计算法的核心, 本文采用菱形搜索算法, 使用双模板搜索的方法, 分别是大菱形搜索模板(LDSP)和小菱形搜索模板(SDSP), 如图 2、图 3 所示:



**Figure 2.** Large diamond template

**图 2.** 大菱形模板



**Figure 3.** Small diamond template

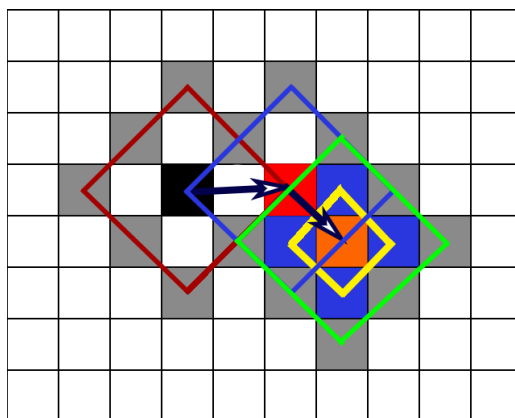
**图 3.** 小菱形模板

大菱形搜索策略采用中心生长策略, 从最佳的起始点开始搜索, 先由大菱形模板开始搜索, 计算模板中每个宏块的 SAD 值, 根据局部最优的宏块所在的位置(顶点、边和中心)不断生长, 将该局部最优块变为下一次大菱形模板的中心继续搜索, 直到其局部最优的宏块在大菱形模板的中心。停止生长, 换成小菱形模板进行中心周围的 4 个宏块的搜索, 最终确定最佳的匹配块, 如图 4。

匹配完成后就能获取该视频序列的运动矢量, 但是由于图像质量, 噪声等问题, 会出现一些运动矢量不正常的问题, 如图 5 和图 6 所示:

这些噪声信息对地震视频判读的准确性影响很大。如图 5, 会导致算法将其识别为发生地震的视频。图 6 虽然发生了地震, 但是也会影响算法对改视频中地震部分截取的结果不准确, 甚至会将整个视频截取下来, 这样做就没有意义了。因此对运动矢量进行后处理是十分必要的。

针对一些全程噪声影响的运动矢量序列, 先采用每一帧图像减去整个图像序列的运动矢量的平均值如式(2)。这样做可以将整体的运动矢量的向下降。但是如果是正常的运动矢量序列, 其大部分帧的矢量



注: 红色框的菱形是第一次搜索, 蓝色的是第二次, 绿色是第三次, 黄色的换成了小菱形, 是最后一次搜索, 获得最终匹配结果。

Figure 4. The final match result

图 4. 最终匹配结果

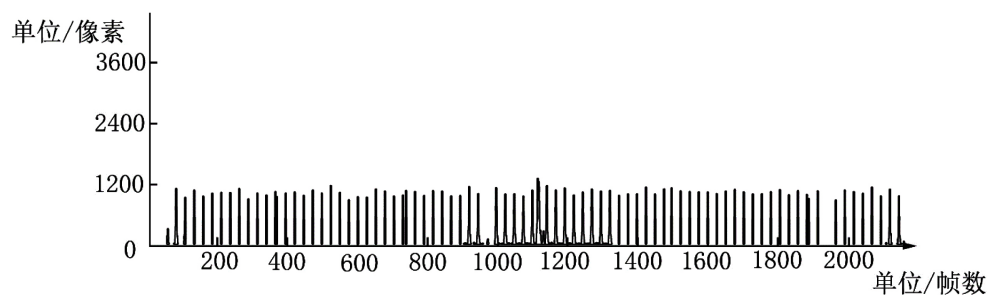


Figure 5. Motion vector without vibration image sequence

图 5. 无震动图像序列的运动矢量

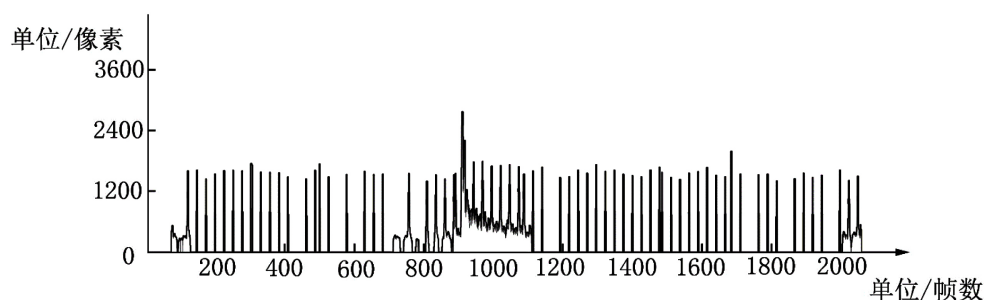


Figure 6. Motion vector with sequence of vibrating images

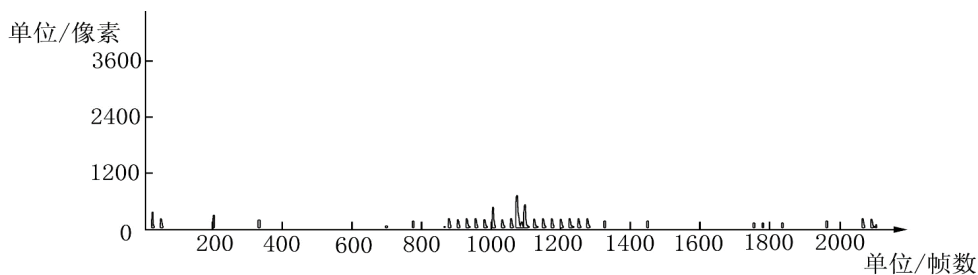
图 6. 有震动图像序列的运动矢量

都为 0, 整个序列的平均值就比较小, 对这个操作的就不敏感。

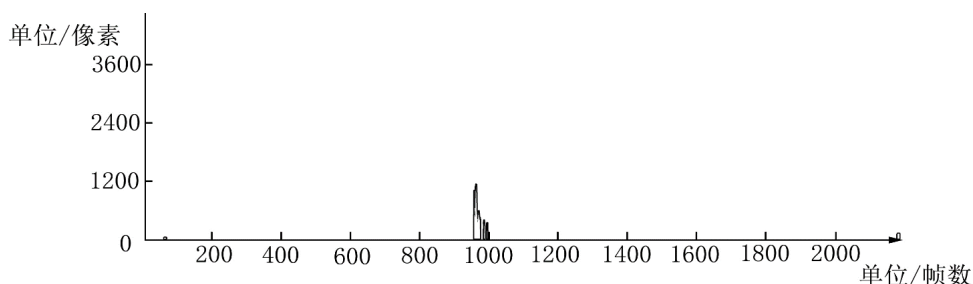
$$MV_i = MV_i - \frac{1}{n} \sum_{k=1}^n MV_k \quad (2)$$

经过该操作处理后如图 7 和图 8 所示:

效果显而易见, 孤立的噪声向量都被过滤掉了, 只剩下发生震动的帧的运动矢量保留了下来。虽然这样的过滤可能会损失掉一些信息, 但是在截取和判断该视频是不是地震视频的过程中, 这些信息并不这样。后处理操作也有利于获取最大震动的帧的帧号, 以便之后的过滤闪光和规律物体运动异常的操作。



**Figure 7.** After no vibration after treatment  
**图 7.** 无震动后处理



**Figure 8.** Results after filtering out noise  
**图 8.** 过滤孤立噪声后的结果

## 4.2. 基于灰度值分析与特征点匹配的地震误报滤除

上一小节我们已经获取图像序列的运动矢量以及最大的矢量。但是计算机会将夜间灯光的闪烁，车辆或者行人等物体进行有规律的运动场景识别成地震。因为这些视频场景复杂，内容中全程有物体在运动，其图像序列的运动矢量形式千奇百怪，有些与正常震动的运动矢量千差万别，有些却与其十分相似。所以基于这些因素，我们并不能通过简单的设置个阈值来区分这类异常和正常的震动情况。

### 4.2.1. 亮度剧烈变化场景

在监控视频中，闪光现象是经常出现的。亮度剧烈变化也存在两种情况，一种是局部亮光的移动(涉及到目标分割，放到下一节中详细讨论)，一种是整个画面的亮度发生剧烈改变。我们将图像统一转化成灰度图，每个像素的灰度值来表示图像的亮度。方法如下：

1) 在获取每一帧的运动矢量的同时，统计该帧的平均灰度值如式(3)。获取该图像序列的灰度值序列 List\_AVG 并求出平均值 AVGs。

$$AVG = \frac{1}{nm} \sum_{x=1}^n \sum_{y=1}^m p_{ix(x,y)} \quad (3)$$

$p_{ix(x,y)}$  为坐标为  $(x,y)$  的像素点的灰度值， $n, m$  分别是图像的长宽。AVG 为该帧的灰度平均值。

2) 获取灰度值最大值 MAX 以及它对应的序号  $L$ 。

3) 在最大值对应的序号的前后  $\lambda$  帧图像进行像素值的统计，判断这  $2\lambda$  帧中灰度值与平均灰度值的差的绝对值大于  $\alpha$  的个数所占的比例是否大于  $\beta$  以上，是就判断该地震视频很有可能是高亮闪光或者灯光突然暗掉所造成的误判。否则就进行之后的操作。

这样我们就可以过滤掉大部分由于图像亮度突然发生剧烈变换所造成的异常。

### 4.2.2. 物体规律运动场景

在监控中，物体的规律运动是时常出现的情况，车辆和行人是这一类物体的代表。规律运动指的是

物体向着某个特定的目的地运动的。这类物体通常从监控视频中的一端出现然后消失在监控视频的另一端, 与正常的震动视频的运动矢量序列分布情况相似。本文从图像的内容出发, 观察物体的运动情况。结果发现其运动轨迹大致都是朝着一个方向前进, 具有一定的规律。而地震时, 画面的震动是上上下下, 无规律。以这个为出发点就可以将正常的地震视频和该类异常区分开来。为了计算出物体运动具有一定的规律性, 就必须先将该运动目标分割出来, 再与相邻帧进行匹配。结合以上的情况, 该算法的流程如下:

- 1) 获取最大运动矢量所对应的序号(即帧序号), 再对其前后  $\lambda$  帧进行遍历, 采用背景差分的方法, 进行高斯背景建模。
- 2) 通过建立好的背景模型进行移动目标的分割, 再进行 SURF 特征的提取, 匹配。
- 3) 通过匹配的结果来确认相邻匹配帧之间目标的运动关系, 保存并与下一次匹配结果进行联系。最终确定目标的运动方向以及其规律性。

运动目标检测有很多: 块匹配法, 光流法, 帧差法, 背景差分法。本文采用背景差分的方法, 因为该方法使用于背景基本不变的场景, 符合本项目的应用背景。1) 监控摄像头基本是静止的。2) 检测的视频一般较短, 不会发生太阳光照亮度的变化。基于以上条件采用背景差分的方法具有快速准确的优势。

背景差分法的核心就是差分, 即通过当前帧(先转换为灰度图 SOURCE)与建立好的背景(BG)进行相减[13], 就可以获得一副灰度值差值图。差值图的灰度值大部分应该是 0 左右的, 还有一部分是绝对值较大的正值和负值。这些正值和负值对应的像素点所在的区域就是运动的目标。由于既有正值又有负值无法设置一个阈值  $\lambda$  来把背景和前景区分开来, 就将正负灰度值进行取绝对值操作。再将该差值图进行二值化操作, 这样我们就获得了一幅灰度值只有 0~1 的掩膜图(MASK)。0 区域代表背景, 1 区域代表前景。通过将这张掩膜图与原图相乘, 就得到运动目标的图像(SEG)。本方法公式可以表示为式(4), (5):

$$\text{MASK}(x, y) = 1 \text{ if } \text{abs}(\text{SOURCE}(x, y) - \text{BG}(x, y)) < \lambda \text{ else } 0 \quad (4)$$

$$\text{SEG}(x, y) = \text{MASK}(x, y) * \text{SOURCE}(x, y) \quad (5)$$

但是通过观察可以发现, 该运动目标不完整, 有一小部分缺失。为了保存更多的运动目标的特征, 我们要让目标分割出来的图像更加的完整。保存更多的特征就意味着目标匹配的稳定性就会更高。因此我们对掩膜图进行膨胀操作, 增加前景部分的面积。但是还是有些缺失。基于以上的情况, 我们对每个连通域求取外接矩形, 再对矩形进行填充, 然后与原图相乘。显然, 我们得到了完整的前景图。分割结果如图 9 所示(从左到右从上到下依次是: 背景差分法求得的掩膜图、膨胀处理后图、外接矩形图、目标分割图):

本文采取特征点匹配的方式采取目标匹配, 描述目标的特征采用的是 SURF 特征。SURF 特征是一个多维的向量, 它通过计算特征点周围的水平和垂直方向像素的哈尔小波变换, 求出特征的方向。再对特征点周围  $20 \times 20$  的区域进行划分, 形成 16 个子区域, 每个区域求取 4 个特征分量[14]。从而, 形成了一个 64 维的 SURF 特征描述子。它是 SIFT 特征的衍生版本, 不但有着 SIFT 特征的优点, 还有着较快的运算速度和较强的鲁棒性。适合监控视频中, 目标物体随着运动的进行发生变大或变小的场景。匹配效果如图 10 所示:

接下来就是是否是规律运动的判断, 两帧之间的运动目标中匹配的特征点有  $\beta$  个, 通过这些特征点可以求出运动目标的运动矢量。在 2D 图像平面中, 物体运动的方向一共有 4 种:  $X^+, X^-, Y^+, Y^-$ 。

相邻两张匹配图像的运动目标的相对位置可以用这四个方向中包含  $X, Y$  的任意两个方向表示。当前帧物体的运动方向可以由式(6)、(7)表示:



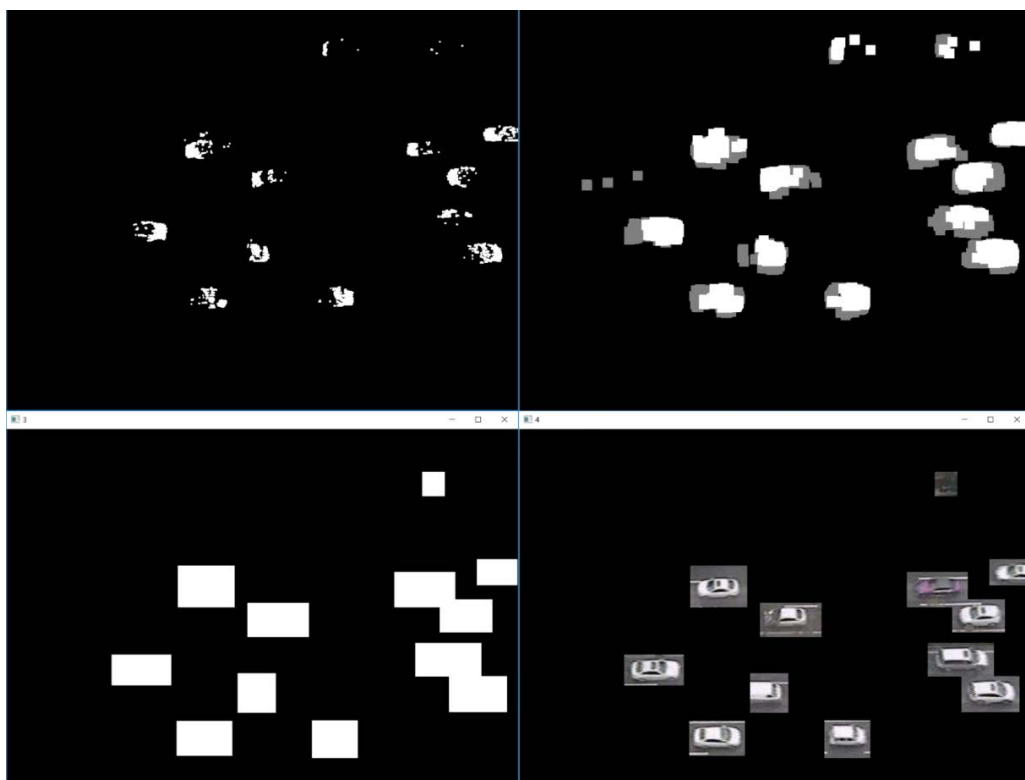


Figure 9. Target segmentation process diagram  
图 9. 目标分割过程图



Figure 10. Feature point matching diagram  
图 10. 特征点匹配图

$$DX = X^+ \text{ if } \frac{\text{count}(dx_{obj2} - dx_{obj1} > 0)}{SUM} < \gamma \text{ else } X^- \quad (6)$$

$$DY = Y^+ \text{ if } \frac{\text{count}(dy_{obj2} - dy_{obj1} > 0)}{SUM} < \gamma \text{ else } Y^- \quad (7)$$

$DX$  和  $DY$  是运动目标的二维平面上的运动方向,  $dx_{obj1,2}$  表示当前帧和前一帧的其中一个特征点的  $X$  坐标(以图像左上角为坐标原点),  $dy_{obj1,2}$  表示当前帧和前一帧的其中一个特征点的  $Y$  坐标,

$\frac{\text{count}(dx_{obj2} - dx_{obj1} > 0)}{SUM}$  这个表达式表示当前帧的目标中特征点的  $X$  或者  $Y$  坐标大于前一帧的个数占总个数的比例。如果比例小于  $\gamma$  则为运动方向为  $X^+$  或者  $Y^+$ , 否则为  $X^-$  或者  $Y^-$ 。从而可以确定目标的运动方向为  $(X^?, Y^?)$ 。接着获取之后的匹配结果进行分析, 当方向序列中有着连续的相同的方向, 则将该地震视频判断为物体规律运动场景异常。

### 4.2.3. 烈度的估计

两种场景都能用以上的方法进行过滤了, 接着就是通过已获得的最大的运动矢量来估计加速度和烈度。目前已给出的地震视频中并没有给出对应监控摄像头的相关硬件参数和内容的长度的一些标定。所以单单通过运动矢量无法完全准确的计算出地震的加速度。因此我们通过参数的拟合来快速估计地震的烈度。利用 10 个地震视频拟合出了一个经验函数, 如式(8)所示:

$$a = \frac{\text{MAX}(MVS) * 50}{30} * \text{Parameter} \quad (8)$$

$\text{MAX}(MVS)$  为当前地震视频的最大运动矢量,  $\text{Parameter}$  为烈度系数 3。通过这个经验函数来对剩下的地震视频进行预测。

## 5. 实验结果与分析

### 5.1. 一般场景有效性分析

我们采用浙江省地震局提供的磐安地震中监控摄像头所传回的视频以及在网络上的一些视频素材作为实验样本, 一共 126 个可用的视频素材, 因为网络上的地震视频大多数都是以新闻的形式, 画面不只有地震的内容还有一些广告等干扰物。所以测试的数据集也不是很多, 网络上也没有相关的数据库。但是这些数据至少能够证明我们的方法对中小地震烈度的快速估计还是有效果的。以下是一些实验结果, 如图 11, 图 12 所示:

图 11 显示的是人工标注的各烈度等级的地震视频的个数和算法估计的地震视频的个数, 实验结果我们可以发现, 烈度为 3~5 的估计的比实际多出了 1~3 个, 而 6~8 级的估计比实际少了 1~3 个。但是在从图 12 中的结果可以看出预测的烈度与实际的烈度的平均值相差很少完全符合地震烈度快速估计的要求。而 7~8 级地震视频的素材太少, 无法有很好的评价价值。而低烈度的会比真实的多一些原因是进行参数

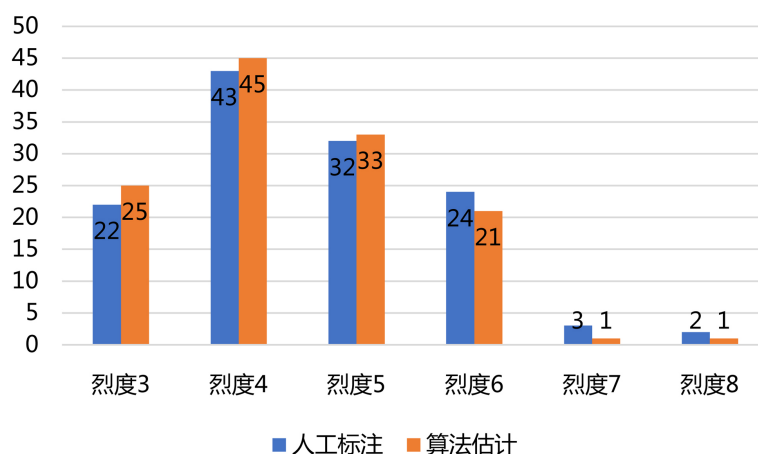


Figure 11. Each intensity and estimated distribution

图 11. 各烈度标注和估计分布图

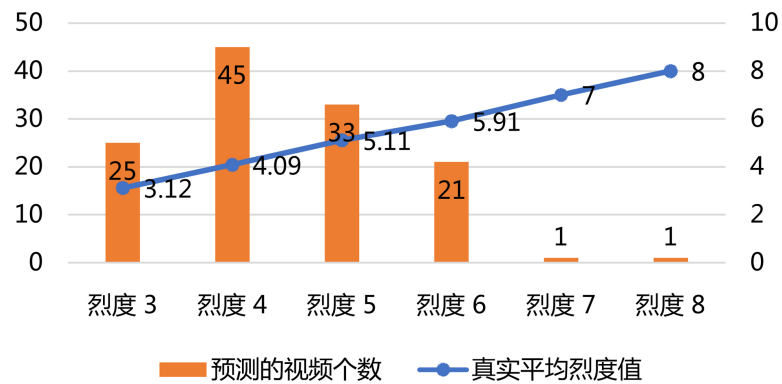


Figure 12. The true average intensity of each intensity prediction  
 图 12. 各烈度预测的真实平均烈度

拟合的时候, 低烈度的地震视频占了多数, 所以可能会有些误差。但是这个项目是针对浙江地震所进行的, 发生高烈度地震概率很小。综上, 该算法在针对中小地震时, 烈度估计的误差较小。以下是一部分绘制的抖动图, 如图 13 (图的横坐标为地震视频的帧数, 纵坐标位像素的位移矢量值, 高低代表地震的强度)所示:

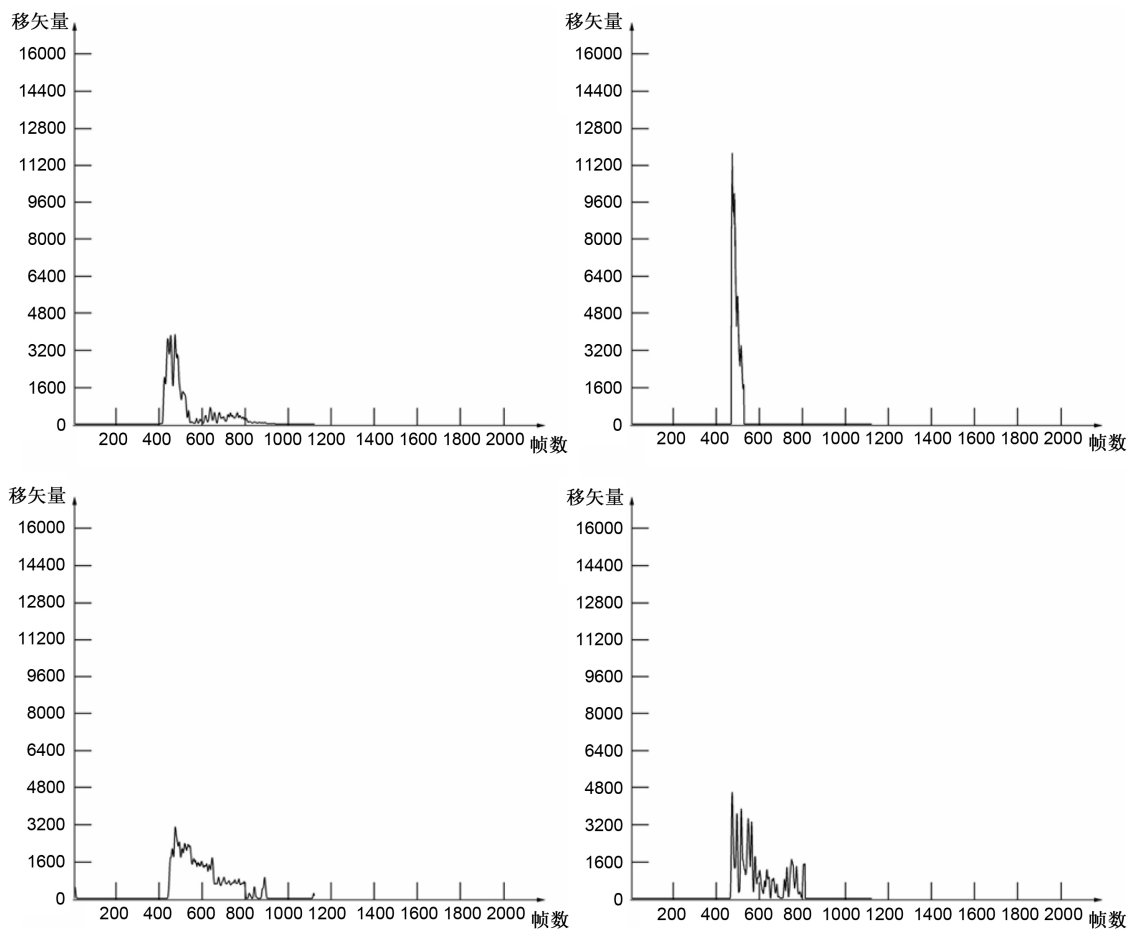


Figure 13. Jitter plot automatically drawn by the algorithm  
 图 13. 算法自动绘制的抖动图

## 5.2. 易误判场景有效性分析

易误判场景分为三类: 无震且无其他明显干扰场景(一般是视频中有些噪声)、亮度剧烈变化场景和物体规律运动场景。

我们同样采用浙江省地震局提供的磐安地震中监控摄像头所传回的视频以及在网络上的一些视频素材作为实验样本。无震视频加两种异常视频一共 28 个, 以下是一些实验结果, 如表 1 所示:

**Table 1.** Abnormal situation misdetection table

**表 1.** 异常情况误检表

类别	无震	亮度剧烈变化场景	物体规律运动场景
个数	10	7	7
误检个数	0	1	2
误检率	0%	14.3%	28.6%

从实验结果可以看出, 无异常情况的无震视频不会被检测为地震视频, 亮度剧烈变化异常也能相对较好的过滤。物体规律运动异常误检为 2 个, 原因是道路环境过于复杂。但是一般的画面质量较高, 且视频中画面才常态是静止的情况下, 该算法还是具有较高的鲁棒性。以下是一些具体的实验结果。

### 5.2.1. 亮度剧烈变化场景

亮度剧烈变化场景主要出现在晚上, 我们看下被判断为亮度剧烈变化场景的最大运动矢量对应的那一帧的自动截图, 如图 14, 图 15 所示:



**Figure 14.** Highlight flash screenshot

**图 14.** 高亮闪光截图



**Figure 15.** Highlight flash screenshot

**图 15.** 高亮闪光截图

### 5.2.2. 物体规律运动场景

物体规律运动场景相对比较复杂, 设计的算法无法考虑到方方面面的问题, 本文设计的算法对道路方向大致为一个方向的场景工作的效果比较好, 如图 16, 图 17 所示:



Figure 16. The road direction is relatively consistent

图 16. 道路方向相对一致



Figure 17. The road direction is relatively consistent

图 17. 道路方向相对一致

## 6. 结论

本文提出一种基于视频分析的地震检测算法。我们采用了 H.264 标准中的运动估计算法, 基于宏块匹配的方法来判断一个视频中是否发生地震, 且定位地震部分在视频中的位置, 进行精准截取。然后通过经验函数和视频的最大运动矢量来估计地震的加速度和烈度并绘制出地震视频的抖动图和加速度图。

接着针对一些常见的异常(亮度剧烈变化场景和物体规律运动场景)提出了解决方案。主要是采用灰度值分析与运动目标分割和匹配来解决这两个场景。

第五节的实验结果中可以发现, 本文提出的算法对无异常的中小地震的检测分析有着比较好的效果。对无震视频也可以很好的将其识别过滤。对场景稍微简单的异常地震视频也能够具有较好的鲁棒性。

## 基金项目

宁波工程学院王伟明基金项目(2017003)。

## 参考文献

- [1] 罗福龙, 蔺大禄. 地震仪器检测技术探讨[J]. 石油管材与仪器, 2018, 4(1): 1-4.
- [2] 郭志宇. 基于视频监控的震害快速评估与地震应急技术可行性研究[D]: [硕士学位论文]. 兰州: 中国地震局兰州地震研究所, 2017.
- [3] 李东平, 刘倩倩, 龚俊. 基于海量视频监控的中小地震烈度快速判断方法尝试[J]. 地震研究, 2017, 40(2): 324-332.
- [4] 于飞, 黄士坦. H.264 运动估计算法分析[J]. 计算机技术与发展, 2009, 19(4): 115-118.
- [5] Wenger, S.H. (2003) 264/AVC over IP. *IEEE Transactions on Circuits and Systems for Video Technology*, **13**, 645-656. <https://doi.org/10.1109/TCSVT.2003.814966>
- [6] 夏东. 智能视频监控中目标检测、跟踪和识别方法研究[D]: [博士学位论文]. 长沙: 国防科学技术大学, 2012.
- [7] 杨勇, 王桥, 吴乐南. 基于标号场的光流法二维运动估计[J]. 电子与信息学报, 2001, 23(12): 1321-1325.
- [8] 钟平, 于前洋, 金光. 基于特征点匹配技术的运动估计及补偿方法[J]. 光电子·激光, 2004, 15(1): 73-77.
- [9] Lowe, D.G. (2004) Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, **60**, 91-110. <https://doi.org/10.1023/B:VISI.0000029664.99615.94>
- [10] Koenderink, J. (1991) Affine Structure from Motion. *Journal of the Optical Society of America A*, **8**, 377-385. <https://doi.org/10.1364/JOSAA.8.000377>
- [11] 余洪山. 基于三维视觉系统图像特征和三维信息的运动估计方法[P]. 中国专利, CN201310544349.7, 2014-02-05.
- [12] 秦晓波, 霍立文, 柴志成. 基于宏块的 Hs 光流算法运动估计优化研究[J]. 电脑迷, 2018(10): 150+152.
- [13] 肖丽君. 基于背景减与帧间差分结合的视频运动目标分割[J]. 北华大学学报(自然科学版), 2010, 11(5): 467-470.
- [14] 崔振兴, 曾威, 杨明强, 等. 一种改进的 SURF 快速匹配算法[J]. 江苏师范大学学报(自然科学版), 2014, 32(3): 41-46.