**摘 要**

本文主要考虑智能交通系统中基于单目摄像头的车辆测速问题，对图像去噪、运动目标检测和运动目标跟踪等进行了多方面的研究，并提出了一种利用国家标准规定的车道分界线尺寸建立图像平面到道路平面的映射关系的方法，从而省去了传统摄像机标定法的繁琐步骤，提高了算法的可操作性。

在算法设计方面，本文综合运用中值滤波去噪，使用帧间差分法和形态学处理检测车辆，然后利用基于像素距离的运动目标追踪方法和建立的图像-道路映射关系进行测速；在算法验证方面，本文使用Python编程语言和OpenCV开源视觉库进行程序的编写，对BrnoCompSpeed数据集中的一个视频进行测试，测速误差率的平均值为2.7%。

**关键词**：智能交通系统，车辆测速，目标检测，目标跟踪

**ABSTRACT**

In this paper, I mainly focus on the vehicle velocity measurement from a single monocular camera, which is an important part of Intelligent Traffic System. Addressing this problem, I have made a lot of researches on image denoising, moving object detecting and tracking. Using the standard dimensions of the lane dividing marking, this paper proposed a new method on creating mapping from image plane to road plane, which reduced the noisy steps of the classic camera calibration method and improved the operability of the algorithm.

For algorithm designing, this paper comprehensively used median filter to reduce image noises, frame subtraction and morphological operations to detect moving objects, and finally, the object tracking based on pixel distance and image-road mapping to measure vehicle velocity. For algorithm verification, I used Python and OpenCV for programming, and then tested on one video of the BrnoCompSpeed dataset. Finally, the mean of errors is 2.7%.

**Key Words**：Intelligent Traffic System, Vehicle velocity measurement, Object detecting, Object tracking

**1.绪论**

* 1. **研究背景及意义**

近年来，随着我国道路交通快速发展，城市机动车数量的迅速增加，越来越多的交通问题和安全隐患凸显了出来。截至2018年末，中国高速公路的总里程已经超过13万公里[[1]](#footnote-1)，远超其他国家，每年还在以超过5000公里的速度在增加，而中国[汽车保有量约2.4亿辆](http://www.baidu.com/link?url=iejgkyRzfv1guv-T7oWixsgYDYzR0kA6FLMXI3pay-R9KxZllJpo726DeBTVqjTJxMP2emsQ7S6mspzIuXw48d2wwuS7wQ9XsNp0VD8z5Ku)[[2]](#footnote-2)，每年交通事故死亡人数超25万[[3]](#footnote-3)。由此可见，我国的道路交通状况日益复杂，尽管国家投入了巨大的人力物力进行城市道路及高速公路网的建设，道路的发展也还是跟不上车辆的增加，交通严重拥堵的事件频频发生，因车辆不按道路规定速度行驶而造成的交通事故日益增多，而且车辆引起的交通事故一般会产生较为严重的后果，这些问题给交通管理部门的工作带来了沉重的负担。

针对交通管理问题以及顺应新信息技术的发展，许多国家都在大力构建智能交通系统（ITS，Intelligent Transport System），这是一种综合地将信息技术、数据通讯传输技术以及计算机处理技术等运用于整个交通运输管理体系，而建立起的一种在大范围内、全方位发挥作用的，实时、准确、高效的综合运输和管理系统。目前美国、欧洲和日本处于该领域发展的前沿，新加坡和澳大利亚等国家对于ITS的建设也取得一定的成效。紧跟发达国家的步伐，我国交通管理部门自90年代开始推动ITS的建设发展，在研发的加大投入和资本市场的作用下，国内ITS已经具备一定的自主研发能力和广阔的市场前景，其发展水平不再是跟随国外的状态了。如今在汽车产业的智能化、物联网发展趋势日益加速的情况下，无人驾驶、车牌识别和人脸识别等技术逐渐进入人们的视野，成为日常出行密不可分的一部分。

交通管理的难点之一是判断车辆是否按规定速度行驶，而目前道路主流的测速形式有三种：

（1）地感线圈测速。将两组间隔一定距离的相互平行的矩形状地感线圈埋在距离抓拍摄像头大约10米开外的路面下，分别记录某辆车通过两组线圈的时刻，再通过时间差和间隔距离来测定车辆经过的平均速度。这种测速方法的优点是技术稳定，因没有更多精密高智能化的设备却能获得比较高的捕获率，因此性价比较高。缺点是会大范围破坏路面，路面一旦变更则需重埋线圈，施工和维护难度较高；

（2）雷达测速。当车辆进入雷达探测范围后，雷达会发出一定频率的无线电波，在接收到反射回来的光波后，根据接收到的反射波频移量，应用多普勒原理计算得出车辆的行驶速度。雷达测速的优点是精准，价格较低。缺点是对角度的要求较高，且只适合在车流量较小的公路上对单车道进行监测，测速效率比较低；

（3）激光测速。激光测速利用红外线半导体激光二极管发射出一定频率极窄的光束精确地瞄准目标，通过测量红外线光波在设备与目标之间的传送时间来决定速度，优点是测量速度快、监测目标准确、测速准确度高，能够在较高车流量的路况上准确地工作。缺点是价格昂贵。

本文所要研究的是通过监控视频对行驶车辆进行测速，即所谓的视频测速方法。传统的视频测速系统是虚拟线圈视频测速系统[3]，即在视频图像中的车道上，相距（30-50）m处设两个虚拟线圈，由于视频的帧率是一定的（即摄像机采集图像的速度），通过计算图片的帧数可以得到经过的时间，利用车辆通过两个虚拟线圈的时间差，就可得出车辆的运行速度。对比主流的三种测速方法，视频测速具有测速成本低，可同时对多车道进行检测，安装和维护方便等优点，其未能成为高速公路上的常规测速方法的原因在于计算机对视频图像的分析需要消耗巨大的运算能力，实时性和测速精确度还达不到交通部门的要求。

然而，随着计算机硬件的运算能力提升和其价格的下降，利用GPU运算单元参与图像处理，及使用人工智能处理图像的技术异军突起，大大降低了视频测速方法的难度，提高了测速精确度。另外，我国已建成世界上最为庞大的视频监控系统——“天网”系统，大多数监控摄像头集成了光学去抖动、[自动光圈](https://www.baidu.com/s?wd=%E8%87%AA%E5%8A%A8%E5%85%89%E5%9C%88&tn=SE_PcZhidaonwhc_ngpagmjz&rsv_dl=gh_pc_zhidao)等功能，具有灵敏度高、抗强光、畸变小等优点，图像分辨率也有了很大的提升。而严重的交通事故大多数发生在雷达、激光等专业测速仪器缺席而仅有高清摄像头的城市道路区域，这给交通案件的事后取证造成困难，利用视频分析技术来提取车辆牌照、事主人脸和车辆行驶速度等将有利于减轻交通管理人员及执法人员的负担，提高对心存侥幸者的警示作用。

基于上述原因，基于视频分析的车辆测速研究有着极大的现实意义及广泛的应用前景。

**1.2 国内外研究状况**

基于视频的车辆测速研究的主要课题是运动目标检测与跟踪，这是计算机视觉领域中的一个热门方向，国内外的企业和研究机构都投入了大量的人力物力进行深入研究。目前常用的运动目标检测与跟踪方法有帧间差分法、卡尔曼滤波、混合高斯模型、光流法等。

早在二十世纪九十年代，美国国防高级研究项目署就已经设立了VSAM（Video Surveillance And Monitoring）项目[[4]](#footnote-4)，目的在于发展视频的自动理解技术以节省对城市区域和战场进行监控所耗费的大量人力资源。

2013年，蔡寿祥[4]设计并实现了双目立体视觉测速系统，提出使用两个间隔一定距离的摄像头进行拍摄，基于视差原理计算出物体的实际位置，从而给车辆测速，并发现采用的镜头焦距越大，两个摄像头距离越远，离测速目标越近，得到的结果精度就越高。优点是设备可以灵活安装，不需要对场景进行诸多限制。缺点是一次只能对一个车道的一辆车进行测速，如果有多辆车同时进入检测区域，就会引起误报，且摄像机标定繁琐，视差匹配和目标分割算法的计算量大。

2014年，孙宁等人[5]基于对早期虚拟线圈测速法的原理和精度的分析，提出在车道上定义一个矩形检测区，采用摄像头定标法将像素点从图像坐标较准确地映射到检测区内对应的地面坐标，选定车尾线下端中心点作为车速参考点，得到了较为准确的结果。Yuji Goda等人[15]为了提高夜间车辆测速的准确度，采用摄像头捕捉夜间车辆前照灯的光线，通过计算光线长度从而估算车辆行驶速度。

2015年，隋宗宾等人[6]通过结合Lab颜色空间和形态学处理的方法识别车牌来实现机动车混合车道复杂场景下的车辆测速，并利用背景模板的不断更新来防止目标车辆的丢失，缺点是容易受外界复杂物体和环境光线的影响，不够稳定。同年朱周研究了在光照变化和车辆间相互遮挡等复杂条件下的车辆跟踪算法，首先改进了ViBe背景模型以适应光照变化，然后使用非网格分块下的Markov随机场遮挡处理算法利用颜色信息的差异处理车辆遮挡现象。

2017年，陈珂[7]利用自然场景中两组正交平行线在视频图像中形成的消失点之间的内在关系对摄像机的焦距和俯仰角实施精确标定；在此基础上利用视频中目标车辆群体的平均宽度对摄像机与地面之间距离进行自动标定。该方法克服了前人研究中摄像头标定法必须手动标定的缺点[8-9],。而遆晓光等人基于双目立体视觉设计了一种具有光照适应性的多车辆自动测速方法，解决了前人双目摄像头标定繁琐的问题[10]。

**1.3 本研究的主要假设**

本文主要针对单目监控摄像头的同向二车道实时测速进行研究，本研究基于以下几点假设：

1. 目标道路平直，没有大幅度的凸起或凹陷；
2. 采集图像的摄像头是固定在同向二车道的单目摄像头，相对于摄像头视角，运动车辆迎面行驶而来；
3. 监控视频中至少有两帧无车辆的图像；
4. 车辆间无遮挡现象；
5. 除车辆外无其他运动物体，而车辆也一直在行驶，不停靠路边；
6. 车道线（包括车道边缘线和车道分界线，其中车道边缘线是白色实线，车道分界线是白色虚线）比较完整、清晰、准确；
7. 车道分界线的尺寸已知，假设虚线块的长度为，相邻虚线块之间的间隔为。

由假设可见，本文研究的主要应用场景是高速公路上的同向二车道。

**2.视频预处理**

**2.1 图像预处理**

在视频分析中，图像的质量极大程度上影响算法的效率和结果的精度，因此在将算法应用到图像上之前，常常要对图像进行一系列的预处理操作。图像预处理是视频分析中密不可分的一部分，其作用在于消除无效信息，增强有效信息，其主要包括图像灰度化、图像去噪、二值化、形态学处理和几何变换等。

在计算机对数字图像的处理过程中，坐标系的建立一般以图像左上角的像素为坐标原点，横向为x轴，纵向为y轴，从左到右x轴正方向，从上到下为y轴正方向，如下图所示：



图1图像平面坐标系xoy

**2.1.1 图像灰度化**

目前大部分摄像头采集到的彩色图像都是RGB图像，其中每个像素的色彩由R、G、B（即红、绿、蓝）三个分量共同决定。以最常见的24位深度的RGB图像为例，每个分量各占据8位，取值范围均为，这样每个像素可能的取值有种，因此当计算机处理较高分辨率的彩色图像时运算量十分庞大，再加上一般帧率为25的十分钟的视频也有帧图像，如果直接对RGB图像的视频进行处理，对计算机的运算负担相当大。

灰度图本质上是将像素的RGB三个分量用一个取值范围为的灰度值来代替，而且包含了RGB图像中的大部分有效信息，使用计算机处理图像的时候，预先对图像进行灰度变换有利于降低计算机的内存占用和后续车辆检测的运算量。图像灰度化是按照一定的规则对每个像素逐个替代来进行的，根据不同的映射规则，图像灰度化方法可分为：最大值法、分量法、平均值法和加权平均法。

用表示彩色图像上某像素点的坐标，该像素位于图像的第i列第j行，，，分别表示该像素的三个分量，表示转换后该像素的灰度值，下面叙述各种灰度化方法的主要原理：

（1）最大值法：用R、G、B三个分量中的最大值作为灰度值，公式为

（2）分量法：用R、G、B三个分量中的某一个作为灰度值，公式为

（3）平均值法：用R、G、B三个分量的平均值作为灰度值，公式为

（4）加权平均法：将R、G、B三个分量按照各自预先给定的权值进行加权求和后得到的值作为灰度值，公式为

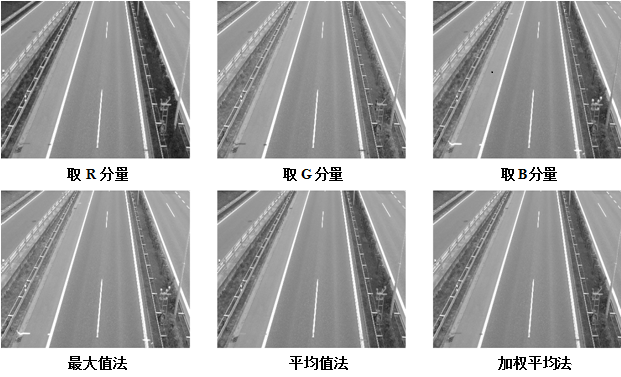


图2同一彩色图像采用不同灰度方法的效果对比

事实上，人的眼睛对不同颜色的敏感度不一，所以RGB图像每个分量的重要性并不相同，因此对RGB三个分量施以不同的权值进行加权平均是比较合理的，人眼对这三种颜色的敏感程度按从高到低排序的结果是绿、红、蓝，从而三个分量的权值也应该遵循此大小关系。为了得到最佳的视觉效果，、和最常见的取值组合是分别取0.299、0.578和0.114，则公式如下：

本文的图像灰度化就是采用该权值进行处理，后续的操作都是在灰度图的基础上进行。

**2.1.2 图像滤波去除噪声**

视频分析领域中，噪声在图像上常表现为一些与周围相比，像素值比较突兀的孤立像素点或像素块，在视觉上使得图像变得不清晰，对于有着许多噪声的图像我们可以说该图像是退化的。在摄像头采集图像和数字图像传输的过程中或多或少都产生噪声，前者主要是成像传感器的性能受光照强度和环境温度的影响；后者主要是传输信道会被干扰，例如**Wifi**传输的图像可能会因为光照或其他大气条件而被污染[1]。图像噪声在统计学上可以看作一种随机信号，从而服从某种概率分布，其按概率密度函数来分大致有高斯噪声、脉冲噪声、伽马噪声、瑞利噪声、指数分布噪声和均匀分布噪声等，其中最常见的是高斯噪声和双极脉冲噪声：

1. 高斯噪声(Gaussian noise)是指其概率密度函数服从高斯分布的一种噪声，极容易在照明度比较低或者高温条件下产生，另一个主要来源是电子电路噪声；
2. 双极脉冲噪声又称作椒盐噪声(salt-and-pepper noise)，是指两种噪声的叠加，一种是盐噪声（salt noise），表现为孤立的亮点，在灰度图中的值十分接近0（白色）；另一种是胡椒噪声（pepper noise），表现为孤立的暗点，在灰度图中的值十分接近255（黑色）。该名称十分形象地表现出其图像特征，视觉上类似于图像上随机分布的胡椒和盐粉颗粒，形成原因是成像过程中的快速过渡，例如错误的开关动作[1]。

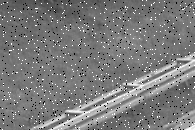
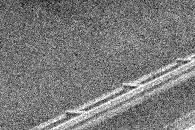


图3原图(左) 、给原图添加了高斯噪声（中）和给原图添加了椒盐噪声（右）

图像噪声会严重影响检测算法的精度和研究人员的直观判断，因此常常使用图像滤波来去除噪声。图像滤波操作又称为图像的平滑化，主要目标是在保留图像细节的同时尽可能地抑制目标图像的噪声。

下面介绍中值滤波：中值滤波的主要思想是对灰度图中的每一个像素，用一个包含该目标像素的滑动模板在图像中移动，将目标像素灰度值替换为模板中的中值。下图所描述的就是用一个形状的模板对图像进行中值滤波的过程：

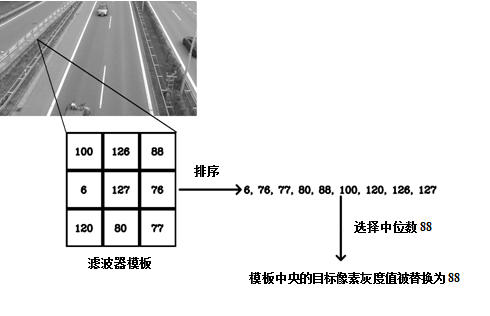
****

图4中值滤波的原理

将中值滤波分别应用于图3中添加了高斯噪声和添加了椒盐噪声的图像，得到以下结果：

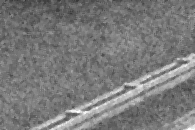


图5原图（左），高斯噪声被中值滤波处理（中）和椒盐噪声被中值滤波处理（右）

由此可见，中值滤波对椒盐噪声比较有效，能够较好的还原图像，对高斯噪声不太有效。后文中如不特别说明，所有图像都已进行灰度化和中值滤波处理。

**2.2 边缘提取**

边缘是图像最基本的特征之一。边缘像素是图像中灰度变化较为剧烈的那些像素，而边缘是连接的边缘像素的集合。在视觉上，边缘通常表现为一些曲线，分隔开两个像素值相近的连通区域。边缘提取的目的是简化图像信息，使用边缘线代表图像所携带信息，排除干扰，获得感兴趣区域，从而简化后续处理。从某种意义上说，噪声点也算是边缘像素，但由于噪声是随机产生的，所以大多数噪声点并不相连，从而无法聚成边缘。

**2.2.1 主要车道的提取**

道路监控摄像头所拍摄到的画面通常包含多个车道，本文所关注的只有主要的同向二车道上行驶的车辆，一般占据视频画面的大部分区域。为了排除无关车道的影响和提高后续车辆检测的效率，应该提取出主要车道。

根据第一章的假设（3）、（5）和（6），本文首先利用相邻帧间差分法（后文3. 1小节中有介绍）从视频中获取一帧没有任何车辆的图像：



图6无车的视频帧

其次，一般高速公路道路两侧的车道边缘线都是白色实线，而道路主体因铺满沥青而呈现深灰色，基于这个事实，可通过设置阈值然后进行二值化来提取车道边缘线：

****

图7二值化提取车道线

实践中，由于车道边缘线受到磨损，可能会像上图中主车道右下方那样的缺口，可通过形态学膨胀（后文3. 2小节中有介绍）将缺口补上：

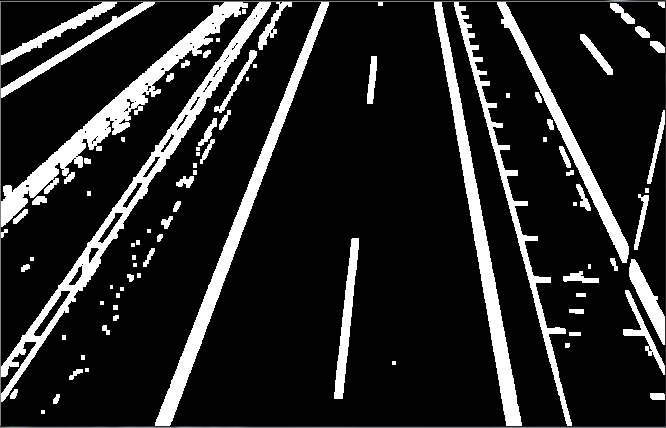


图8经过3次形态学膨胀填补缺口

然后，由于主要车道占据图像中的大部分区域，所以可以通过寻找最大的黑色连通区域的方法[[5]](#footnote-5)分割出主要车道：

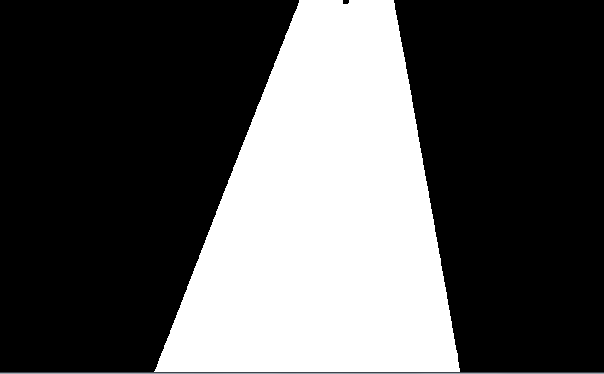


图9寻找最大连通区域

上图中，图像上方中央存在黑点，这是由于车道分界线的虚线块与图像上边界相交所导致的，可以通过取白色区域的凸包来去除：

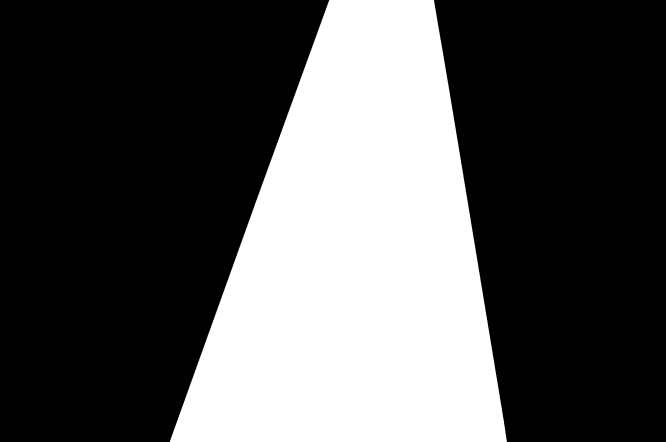
****

图10取凸包得到的主要车道区域

上图是用于提取主要车道的图像掩码，只要将视频的每一帧与该图像掩码作与运算，就可以屏蔽其他车道，后续处理集中关注主要车道，比如将图6与该图像掩码作与运算，得到：

****

图11主要车道

**2.2.2 车道分界线的提取**

由于车道分界线是白色虚线，所以可以直接对图11进行阈值二值化，得到

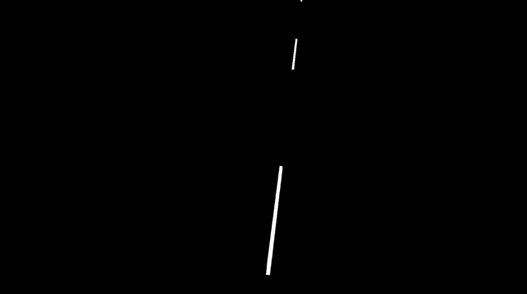


图12提取出的车道分界线

**2.3 图像平面到道路平面映射关系的建立**

为了由视频测量出真实世界的车辆速度，必须建立一个由图像平面到道路平面的映射，使得计算机能够从图像上的像素距离（单位为个）得到路面上对应的实际距离（单位为米）。解决这个问题的传统方法是摄像机标定法，其主要思想是通过求解内参矩阵和外参矩阵而得到图像平面与道路平面的对应关系，但求解矩阵的过程中还要用到实际路面的一些数据，在进行标定前一般需要预先实地测量道路尺寸，且计算过程比较繁琐。实际上在如今互联网时代，为了研究相关算法而从网络下载到监控视频，却不一定能够到实地去测量道路数据，因此摄像头标定法的可操作性太差。于是本文是利用车道分界线的尺寸来建立图像平面到道路平面映射关系，而对于车道分界线的虚线块长度和虚线块之间间隔的距离，许多国家都制定了相关标准文件[[6]](#footnote-6)。本小节基于假设（1）和（2），提出一种利用车道分界线的映射关系建立方法。

设图像中包含的车道分界线完整虚线块个数为n，令图像平面坐标系中这n个虚线块的所有端点从上到下依次为、、、…、，其对应的道路平面坐标系中的虚线块端点分别为、、、…、。

回忆图2.1-1，记图像平面上的原始坐标系为xoy，它以图像左上角的像素为原点，从左向右为x轴正方向，从上到下为y轴正方向。在图像平面上建立坐标系uOv，其中u轴穿过以图像上方第一个完整的虚线块的上端点（即），v轴与y轴重合：

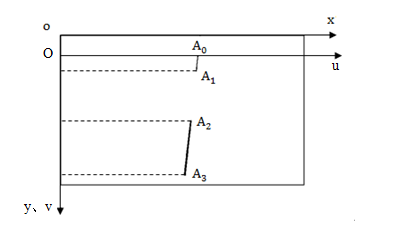


图13图像平面坐标系xoy和uOv

在道路平面上建立坐标系，该坐标系的轴穿过点，且虚线块垂直于轴：

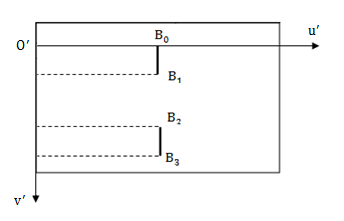


图14 道路平面坐标系

现在的任务是分别去求点、、、…、在坐标系中的纵坐标和点、、、…、在坐标系中的纵坐标，不妨设为坐标系中由点到该点纵坐标的映射。

由假设（7）可以得到

为了从图像分析得到点、、、…、在坐标系中的纵坐标，首先从图12中查找出所有白色连通区域（也就是虚线块），记录每个连通区域纵坐标最小值和最大值，从而得到图13中、、、…、在坐标系中的纵坐标，将其全部减去的纵坐标，那么就能得到、、、…、在坐标系中的纵坐标，分别设为

、、、…、。又令，那么就能得到一系列点，应用最小二乘法拟合这些点，最终得到由图像平面映射到道路平面的映射。

**3.车辆检测**

车辆检测是指从图像中将运动的车辆所在区域完整分割出来，是基于视频分析的车辆测速研究中的关键步骤，后续的车辆追踪效果很大程度上依赖于车辆检测，该步骤中是否有漏检，错检，同一车辆是否检测出不同部分（即所谓的目标分裂现象）将决定整个测速算法的成败。

运动目标检测方法根据摄像头是否固定分为静态背景下的检测和动态背景下的检测两大类。由于高速公路上的摄像头通常都是固定在龙门架上的，因此本文将讨论静态背景下的车辆检测，下文将一幅图像中的运动车辆叙述为前景，则图像中其他像素点都属于静态背景。

**3.1 背景更新与前景提取**

静态背景下的车辆检测的传统方法有帧间差分法，背景减除法和光流法，近年来还出现了基于深度学习的车辆检测方法。

本文采用帧间差分法来更新背景。由于视频中相邻两帧之间，静态背景的灰度变化较小，而前景的变化较大，所以通过将当前帧与前一帧作差并取绝对值，如果存在变化较大的像素（事先取定一个阈值来判定变化是否较大），就说明当前帧或前一帧中有运动物体；如果像素灰度变化都小于某个给定阈值，则说明当前帧和前一帧都没有运动物体，从而可以将当前帧作为新的背景。具体背景更新与前景提取的流程如下：

首先，前文在2.2.1小节中提取出了一帧不含运动车辆的图像（也就是图6），因此可以将其作为初始背景。此后对于视频中的每一帧，将其与当前背景作差，来得到前景；而对于视频中除了第一帧的每一帧，将其与前一帧作差，若所有的像素灰度都小于某个给定阈值，则将当前帧作为新的背景，否则还是使用原来的背景。

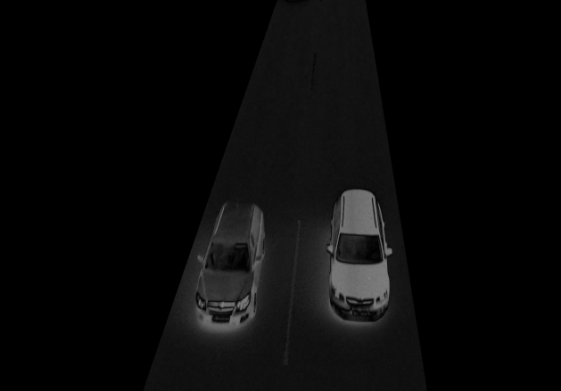
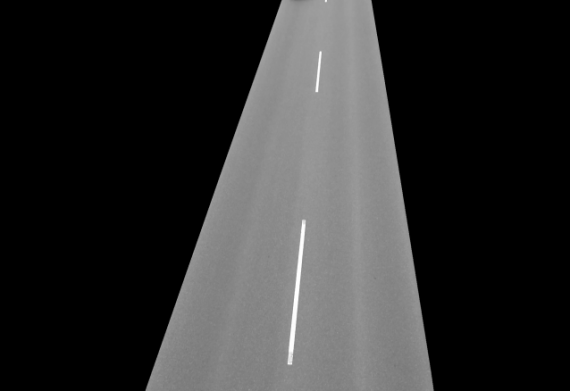
****

图15 使用帧间差分法过程中某个时刻得到的背景（左图）和前景（右图）

**3.2 形态学处理**

数字图像处理中的形态学，是指数学形态学，其主要思想是利用一个预先定义好形状的结构元素来探测图像的局部结构特征。形态学处理最常用的基本操作是腐蚀和膨胀，其他的高级方法诸如开运算、闭运算、顶帽和黑帽都是基于这两种基本操作来进行的。通常形态学操作使用的结构元素的形状有十字架形、椭圆形和矩形等，选择不同的尺寸和形状可以提取图像总的不同特征，实际应用中应该根据处理效果来选择何种尺寸和形状的结构元素。为了方便讲解，本小节的所有形态学处理都采用十字架形状的结构元素。每个结构元素都有一个锚点，十字架形状的结构元素的锚点在其中央，当结构元素在目标图像中滑动到某个像素时，整个结构元素与图像重叠部分的关系将决定锚点所对应图像像素的值。

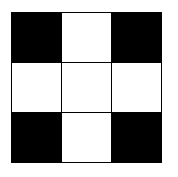


图16 形状为十字架的结构元素

一般而言，形态学操作对于彩色图像和灰度图像都是有效的，二值图像则是灰度图像的特例。基于本文的研究需要，下面只讨论关于二值图像的形态学操作，这些操作主要针对图像中的白色区域（与白色区域相比，同一个形态学操作对黑色区域会导致相反的结果，比如膨胀操作拓展了图像中的白色区域，也就同时缩小了黑色区域）。因此本小节中，在不特别说明的情况下，一旦说到区域都是指二值图像中的白色区域。



图17 对前景进行阈值二值化后的图像

**3.2.1 形态学腐蚀**

当形态学腐蚀作用于一幅二值图像后，对于比较大的区域所得到的结果就像是区域的边界被腐蚀掉了一样，而一些细小零散的区域就会被完全去除，所以也可以用来去除二值图像中的噪声，或者对含有微小连接部分的两个区域进行分割。腐蚀的主要过程是用结构元素在图像中滑动，若在图像中与结构元素中的白色点重叠的部分也全都是白色，那么锚点所对应的图像像素予以保留，否则将其变为黑色。下图仔细地描述了对二值图像进行腐蚀的过程，灰色部分表示在腐蚀中被变为黑色点的像素，可以看出左下角的小区域在腐蚀后消失了。

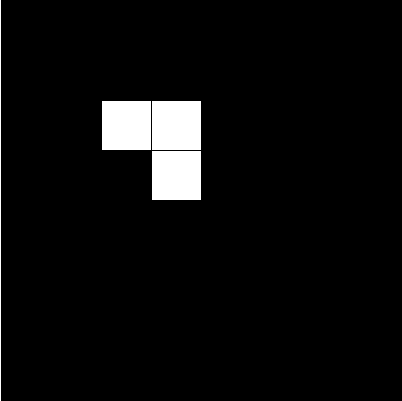
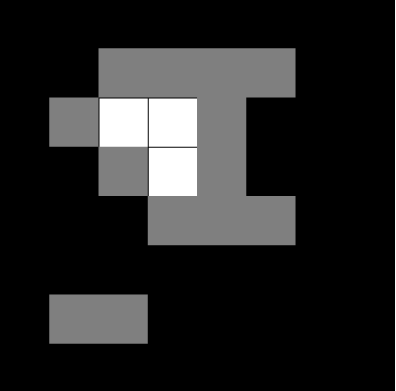
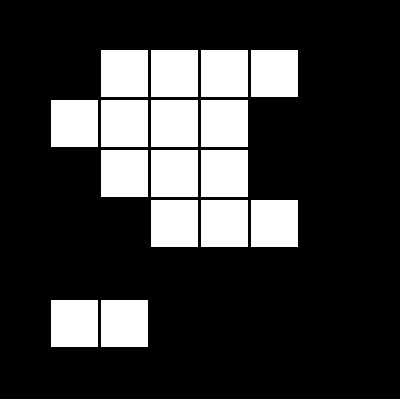


图18 腐蚀前（左）、腐蚀中（中）、腐蚀后（右）

下图以十字架形状的结构元素对二值化后的前景进行3次腐蚀，从中可以直观地看出，形态学腐蚀对于去除细小的区域很有帮助（如原图中的细长的车道分界线），但要注意的是它还有可能扩大运动目标区域内的空洞。

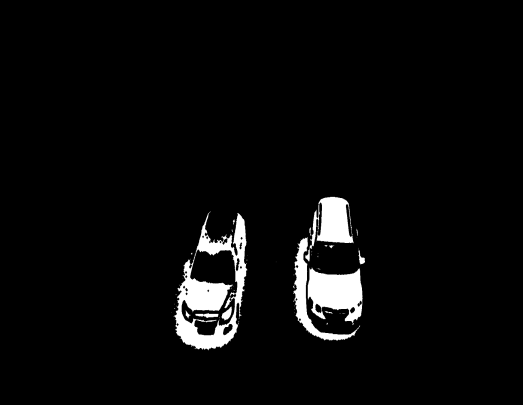


图19原图（左）、进行3次腐蚀之后得到的图像（右）

**3.2.2 形态学膨胀**

形态学膨胀的作用是拓展区域的边界，填补一些较小的区域，同时它还可以连接那些距离比较近的区域，因此膨胀操作经常用于相邻区域之间的合并。对图像进行膨胀操作的过程中，结构元素在图像中滑动，若在图像中与结构元素中的白色点重叠的部分也至少有一个点是白色，那么锚点所对应的图像像素就设置为白色，否则什么都不改变。如下图，灰色部分是新扩张的边界，经过膨胀操作后，原图左下角的小区域与附近的大区域连接起来了。

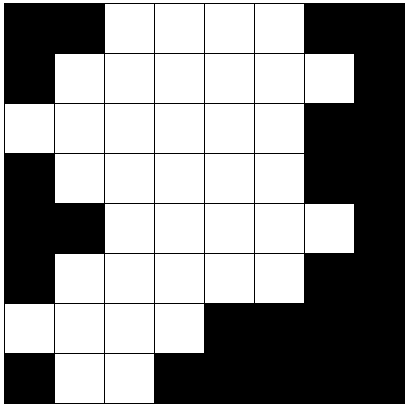
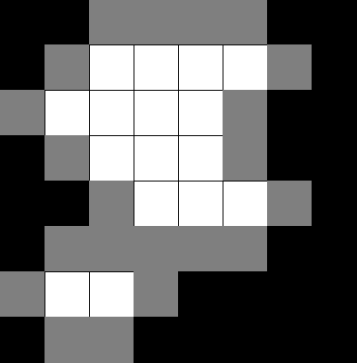
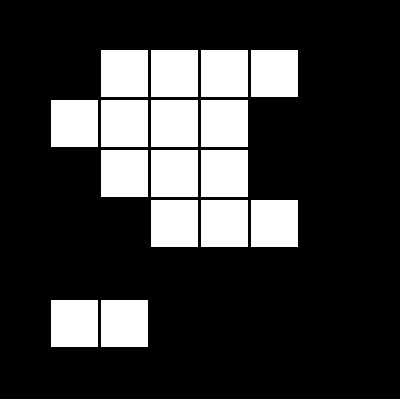


图20 膨胀前（左）、膨胀中（中）、膨胀后（右）

下图以十字架形状的结构元素对二值化后的前景进行3次膨胀，从中可以看出目标区域的外围扩展了，里面的一些空洞也缩小了，但是原本就不需要的车道分界线等噪声被膨胀操作给增强了。

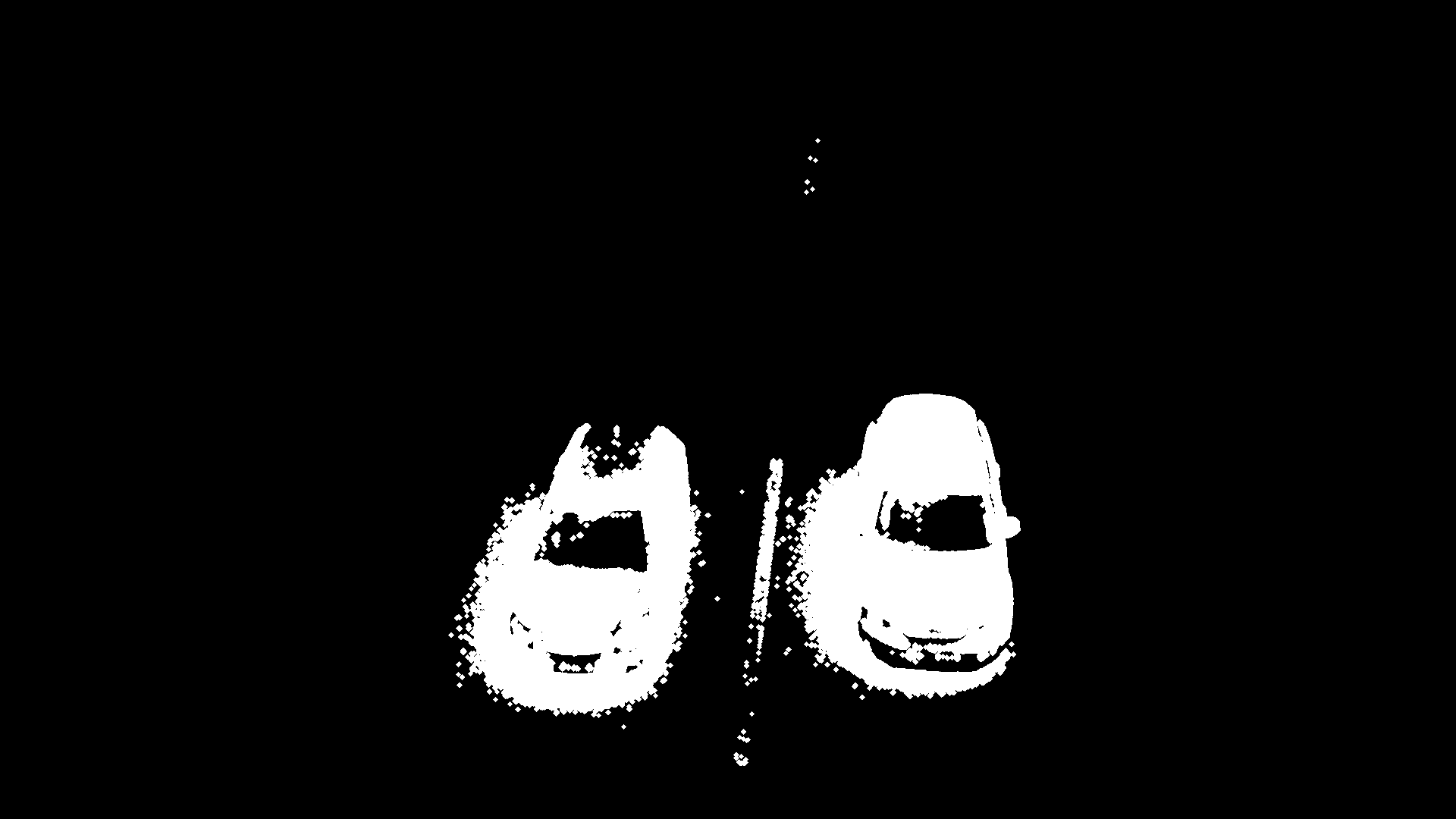


图21原图（左）、进行3次膨胀之后得到的图像（右）

**3.2.3 开运算**

开运算就是先腐蚀后膨胀的过程，主要作用是在不大改变区域面积和形状的情况下消除离散的细小区域或在纤细处分离区域，可以用于移除由图像噪声形成的斑点。下图展示了开运算去除细小区域同时保持大区域形状的能力。

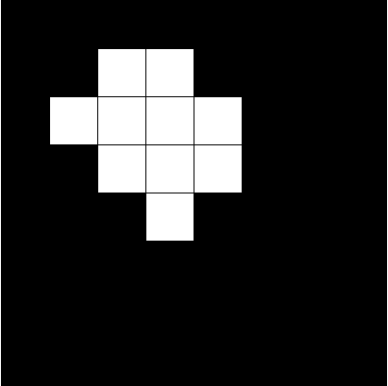
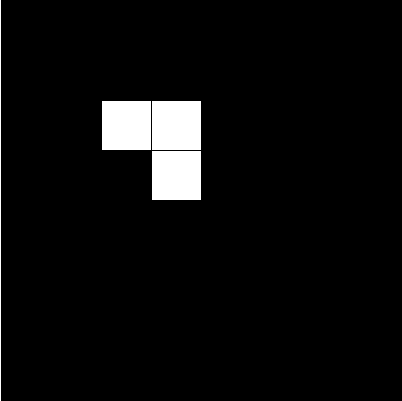
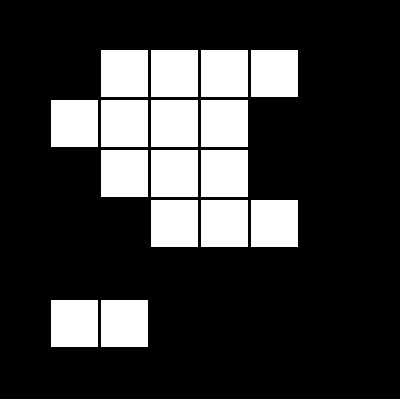


图22原图（左）、腐蚀后（中）、再膨胀后（右）

下图可以看出开运算缩小了干扰区域（车道分界线），同时没怎么破坏运动目标区域的结构，因此有许多微小区域的情况下，开运算是一种很有效的去除噪声的操作。

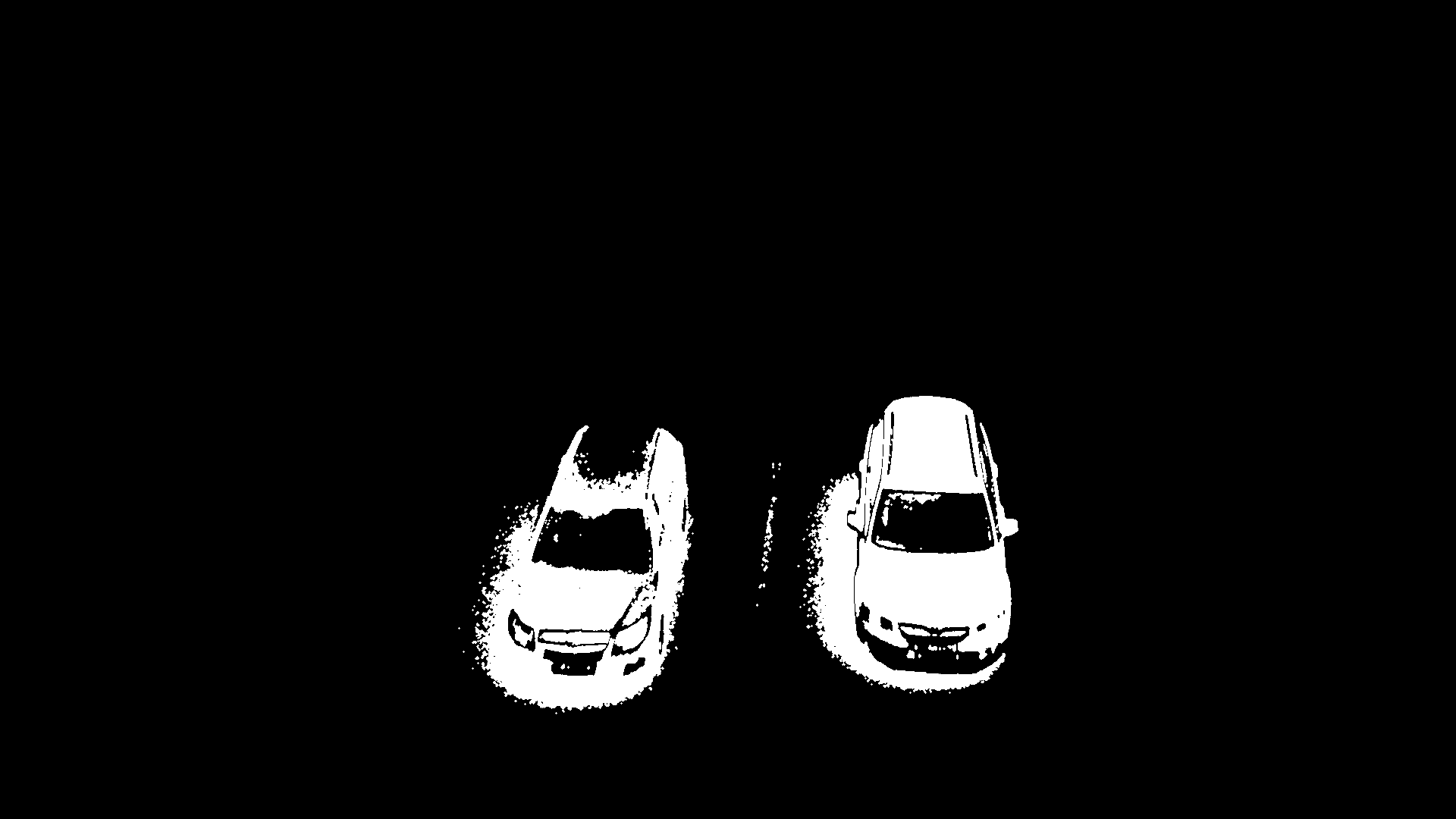


图23原图（左）、开运算（右）

**3.2.4 闭运算**

闭运算就是先膨胀后腐蚀的过程，用来再不大改变区域形状和面积的条件下连接被误分为许多小块的对象和填补小的空洞。如下图，经过闭运算操作后，图中十分接近的两个区域被连接起来，而且大小区域的形状都没怎么改变。

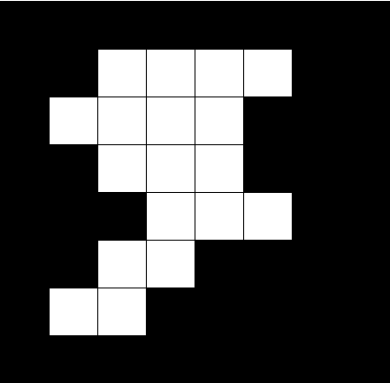
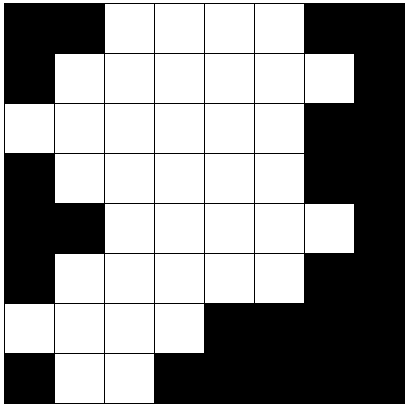
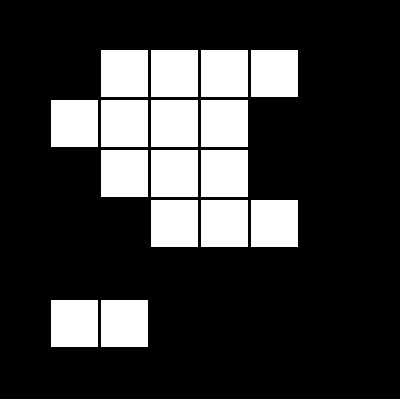


图24原图（左）、膨胀后（中）、再腐蚀后（右）

对比下图中闭运算处理前后，并没有什么大的改变，因此闭运算对这种区域之间相距较远的情况效果不大。



图25原图（左）、闭运算（右）

**4.车辆跟踪**

**4.1 提取车辆位置信息**

获得经过形态学处理的前景后，下一步的任务便是从前景中提取每一辆汽车的位置信息。由运动目标检测算法检测出来的前景各个连通区域的形状通常都很不规则，直接从前景中定位车辆是很不可靠的，为此本文使用最小外接矩形将前景中的各个连通区域分别覆盖起来。本文将该最小外接矩形的底边中点（也就是相当于车头边缘线中间点）作为车辆跟踪基准点P，选取该点的原因有两个：（1）经过试验发现，车头盖作为一个整体，其灰度值与沥青路相差较大，不易发生属于同一车辆的前景分裂现象，从而不会被形态学处理给去除；（2）车头中间点比较容易计算，直观可重复，便于反复调试。下图中用最小外接矩形（白色矩形区域）覆盖前景的连通区域，矩形底边中点（灰色圆点）代表车辆位置：

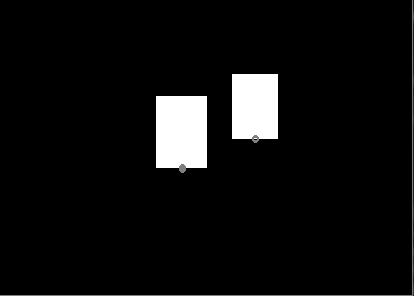


图26 车辆位置信息的提取

**4.2 记录****车辆的运动轨迹**

一般监控视频的帧率为25，相当于摄像头每0.04秒采集一次图片，在如此短的时间中，如果某辆车以55 m/s（即200km/h，远高于我国高速公路上的最高限速120m/h）的速度行驶，那么车辆在相邻帧之间运动的距离为2.2m，远小于一般的车身长度4m。因此即使考虑摄像头视角问题，在图像平面中车辆在相邻帧之间的移动距离也远小于车辆的长度，因此可以依据两帧之间车辆基准点的距离来判断其是否属于同一车辆。

某台车辆的运动轨迹实际上就是一个图像平面纵坐标的列表，每个坐标对应着车辆出现时在视频每一帧中的位置，记录运动轨迹的基本思想（实际编写的程序比这还要复杂一些）是：首先对第t帧的前景搜索所有的车辆跟踪基准点，，，对于(i=1,2,…,n)，将其与第t-1帧中所有车辆跟踪基准点进行对比，找到距离最近的那个点，如果它们的距离小于某个给定的阈值，就将并入所属的车辆轨迹；否则，将作为新出现车辆的第一个轨迹点。

**4.3 基于车道分界线的车速估算**

得到了车辆的运动轨迹之后，结合由2.3小节得到的图像平面与道路平面的映射关系，就可以估算出相邻帧间的车辆在道路上实际移动的距离，再除以视频两帧间的时间（也就是帧率的倒数），就可以得到车辆行驶的瞬间速度。而要想得到车辆在画面中的平均速度，就要用总共移动的距离除以总共的时间。

**4.4 编程实验**

本次实验利用Python编程语言和OpenCV开源视觉库实现了基于视频分析的车辆测速算法，包括视频预处理、车辆检测与车辆跟踪算法的整个流程，代码上传到Github以供查阅[[7]](#footnote-7)。

本次实验采用的数据集是由Jakub等人采集的道路监控视频集BrnoCompSpeed[[8]](#footnote-8)，采集的目的就是给学术界视频分析领域的研究提供可靠的数据集，因此是向所有人开放的，只要不是用在商业方面每个人都可以自由下载。在他们撰写的论文里面有如何利用此数据集的详细介绍[12]，本小节先简单介绍一下该数据集：

数据集主要包括道路监控视频和对应的路面真实数据（groundtruth）两部分，整个数据集大小大约200G，其中路面真实数据中的车辆速度是通过雷达测速数据和GPS轨迹数据互相对照来得到的，而关于道路监控视频，则有以下特点：

（1）视频分辨率为1980x1080，帧率为50；

（2）由位于6个不同地点，每个地点有左、中、右3个方向的监控视频组成；

（3）每个视频大约1小时，大小大约10G；

（4）视频中共有20865辆汽车。

由于数据集太大，所以本次测试只从中挑选符合本文的假设的一个监控视频，该视频是位于数据集中session1\_center目录下的video.avi，道路真实数据文件则是同目录下的gt\_data.pkl，视频中经过的车辆数目为875。另外，该监控视频的监控地点是位于捷克共和国的高速公路，通过咨询该数据集介绍论文的作者之一Roman Juranek，得到了视频中车道分界线的相关尺寸[[9]](#footnote-9)，具体来说车道分界线的虚线块长度是6m，虚线块之间间隔12米。

**4.5 结果分析**

经过编程实验后得到的结果如下表：

表1车速预测误差表

|  |  |  |  |
| --- | --- | --- | --- |
| 车辆编号 | 车速预测值（km/h） | 车速实际值（km/h） | 误差率（%） |
| 0 | 81.199 | 81.371 | 0.2 |
| 1 | 73.107 | 71.118 | 2.8 |
| 2 | 70.678 | 74.664 | 5.3 |
| 3 | 81.872 | 83.3 | 1.7 |
| 4 | 73.615 | 74.47 | 1.1 |
| … |  |  |  |
| 874 | 73.909 | 72.346 | 2.2 |

其中车辆编号是按照车辆出现的顺序由0到874进行编号，车速预测值是指通过本文所介绍的车辆测速算法估算出来的车辆速度，车速实际值是指通过读取数据集中的gt\_data.pkl文件得到的车辆的实际车速，由于数据集中提供的只有平均速度，所以本次实验所估算的车速预测值也是平均速度。用P表示车速预测值，G车速实际值，E表示误差率，则误差率的计算公式为：

最后，再用误差率平均值来衡量该算法的精确程度，本次实验此值为2.7%。由此可知，本文所描述的车辆测速算法其精度能够满足监控的基本需要，然而程序的运行速度不如人意，无法满足实时性测速的要求。

使用Python分析误差率较高的那些车辆轨迹及对应的视频帧，得知误差产生的可能原因有：

1. 视频中的车道分界线有可能由于施工人员画得不够准确，或者出现磨损后未能及时维护而导致其实际尺寸与国家标准的相关规定不符。
2. 摄像头可能因大型货车等的经过而出现抖动，或因持续的高温环境而造成图像的微小形变。
3. 因形态学腐蚀、膨胀等操作使得前景提取不完整。

**5.总结与展望**

**5.1 研究总结**

本文提出了一种基于视频分析的车辆测速算法，并完整地介绍了运动目标检测与跟踪的整个流程，使用Python编程语言与OpenCV开源视觉库实现了基于视频的车辆测速算法，最后在高速公路监控视频上进行算法测试与验证。

本文的创新点主要在于利用车道分界线来建立图像平面与道路平面之间的映射关系，化简了传统摄像机标定的繁琐步骤，提高了车辆测速算法的可操作性和可重复性。

**5.2未来展望**

本论文所介绍的视频测速方法还有不少可以改进的地方：

1. 在视频处理方面，Python处理大量图像的速度还有所不及，还达不到应用的实时性要求，因此下一步的计划是将Python代码改为C++，提高程序运行速度，实现实时测速。
2. 在对前景二值化的时候虽然可以通过多次试验来取得最佳阈值，但这就使得程序只对该测试视频有效，有“过拟合”的可能也就是固定阈值不能适应天气、光照等环境条件的改变。此问题的改进方向是使用自适应的阈值，即按照一定的规则自动修改阈值。比如通过统计每个像素的值来估算图像的亮度，当图像整体画面偏亮的时候增大阈值，在偏暗的时候减少阈值；又或是将一副图像划分成不同的区域，对颜色较深的车辆所在区域使用较小的阈值，对颜色较深的车辆所在的区域使用较大的阈值；另外还有所谓的大津法选取阈值。
3. 对目标区域中的“空洞”的处理有待改善。产生空洞的原因有很多，其中之一就是图像中车窗的颜色通常比较暗，与沥青路的颜色较为接近，因此常被检测算法判断为背景，从而使目标区域产生“空洞”或缺口。对此类问题，本文采用的方法是使用形态学操作中的膨胀和闭运算来处理，对前景进行经过多次的膨胀往往能够使属于同一车辆的各个部分连接起来，但也不能确保填满空洞。该问题的解决方向是研究高效的空洞填补算法，比如基于搜索匹配的样本修复方法、基于多视点的融合处理方法多边形有序表填充算法和种子填充算法等。

**参考文献**

[1] [冈萨雷斯](https://book.douban.com/search/%E5%86%88%E8%90%A8%E9%9B%B7%E6%96%AF)著, [阮秋琦](https://book.douban.com/search/%E9%98%AE%E7%A7%8B%E7%90%A6)译.专著.数字图像处理[M].电子工业出版社.196-213

[2] 钱其荣, 陈勇, 王敏.激光测速在道路交通执法中的应用[J].信息系统工程. 2011(11):127-128.

[3] 陈薏竹, 李一昂.视频测速综述[J].电子技术与软件工程.2014(10):43-44.

[4] 蔡寿祥.基于双目立体视觉的车辆测速系统[D].成都: 电子科技大学. 2013.

[5] 孙宁, 张重德.一种提高视频车速检测精度的算法分析和实现[A].合肥工业大学学报. 2014,37(12):1462-1467,1527.

[6] 隋宗宾, 高杨, 梁宇, 王玉全.基于车牌识别的机混车道视频测速算法[A].现代电子技术. 2015,38(18):125-127,132.

[7] 陈珂.用于视频中车速自动检测的摄像机自动标定方法[A].计算机应用.2017,37(8): 2307-2312,2333.

[8] 于艳玲, 王韬, 袁彬等.基于视频的车速检测算法研究[J].现代电子技术, 2013,36(3):158-161.

[9] 谭方勇, 于福生, 吴建平.基于消失点的坐标校准算法[J].计算机应用, 2011,31(1):58-60.

[10] 遆晓光, 张浩鑫.基于视频处理的具有光照适应性的多车辆自动测速方法[P].黑龙江:CN201710464579.0, 2017-11-03.

[11] 邬治锋.监控视频中车辆平均速度测算方法比较[J].刑事技术. 2012,37(1): 40-42.

[12] J. Sochor, R. Juránek, J. Špaňhel, etc. Comprehensive Data Set for Automatic Single Camera Visual Speed Measurement[J]. IEEE Transactions on Intelligent Transportation Systems. 2018.

[13] A. Nurhadiyatna，B. Hardjono，A. Wibisono, etc. Improved vehicle speed estimation using Gaussian mixture model and hole filling algorithm[J]. ICACSIS. 2013(9): 451–456.

[14] Sochor, Jakub. Fully Automated Real-Time Vehicles Detection and Tracking with Lanes Analysis[J]. Proceedings of The 18th Central European Seminar on Computer Graphics.2014.

[15] Yuji Goda, Lifeng Zhang, Seiichi Serikawa. Proposal a Vehicle Speed Measuring System Using Image Processing[C]. Computer, Consumer and Control (IS3C), 2014 International Symposium on Taichung: IEEE, 2014:541-543.

**致 谢**

感谢Jakub等人采集了可靠的道路监控视频数据集并开源出来，更加要感谢他们中的Roman Juranek向我提供了捷克共和国车道标志标准的相关文件和一些有用的建议。

感谢杨坦老师体谅我的诸多任性，在百忙之中抽空审阅我的论文，并在我困难的时候提供帮助。

卢学荣

2019 年 4月

1. 来自中华人民共和国交通运输部2017年交通运输行业发展统计公报 [↑](#footnote-ref-1)
2. 来自中国公安部交管局2019年发布的数据 [↑](#footnote-ref-2)
3. 来自世界卫生组织2015年的报告 [↑](#footnote-ref-3)
4. 该项目到现在还在运行，它的官网是http://www.cs.cmu.edu/~vsam/ [↑](#footnote-ref-4)
5. 查找连通区域的方法可以参考Satoshi Suzuki等人的论文Topological structural analysis of digitized binary images by border following，本文不作过多介绍。 [↑](#footnote-ref-5)
6. 例如，我国GB 5768.3-2009文件就对车道分界线尺寸做出明确的要求：[高速公路](https://baike.baidu.com/item/%E9%AB%98%E9%80%9F%E5%85%AC%E8%B7%AF)、[一级公路](https://baike.baidu.com/item/%E4%B8%80%E7%BA%A7%E5%85%AC%E8%B7%AF)和[城市快速路](https://baike.baidu.com/item/%E5%9F%8E%E5%B8%82%E5%BF%AB%E9%80%9F%E8%B7%AF)，车道分界线的尺寸为划600间隔900；划400间隔600。其他道路，划200间隔400（单位cm）。 [↑](#footnote-ref-6)
7. 实验代码的所在网址为<https://github.com/Luxuer/3vDetector> [↑](#footnote-ref-7)
8. 实验所采用数据集的网址为<https://medusa.fit.vutbr.cz/traffic/research-topics/traffic-camera-calibration/brnocompspeed/>，在该网页可以下载数据集和相关论文等补充材料 [↑](#footnote-ref-8)
9. 在捷克共和国的TP133标准中可以找到，下载网址为http://www.pjpk.cz/data/USR\_001\_2\_8\_TP/TP\_133.pdf [↑](#footnote-ref-9)